

Understanding the Development of Knowledge in Novice Database Modellers

A thesis submitted to Auckland University of Technology in partial
fulfilment of the requirements for the degree of
Doctor of Philosophy

Supervisors

Prof. Jacqueline Whalley

Dr. Mali Senapathi

2023

By

Asanthika Medawani Imbulpitiya

School of Engineering, Computer and Mathematical Sciences

Abstract

Database modelling is a skill any undergraduate student must master as part of a computer science degree program. For novices, it is a challenging task involving interconnected and abstract concepts. The aim of this research was to gain an in-depth understanding of how novices learn to model by examining their learning processes and barriers to learning.

Modelling tasks typically given to novices were classified to understand their type and difficulty. Several difficulty metrics were explored, and a novel adaptation of the cognitive dimension of the revised Bloom's taxonomy for novice database tasks was established. This work informed the design of tasks that comprised the instrument for this research. In designing the instrument, aspects of cognitive load theory were considered, including segmentation, redirected isolated element tasks, structured presentation of the case study, and sequenced sessions. Vygotsky's zone of proximal development informed a novel intervention model consisting of redirected tasks to support the learners. The tasks of this research instrument were then attempted by novice modellers through a series of think-aloud interviews. Through the constructivist analysis methods of reflexive thematic analysis and narrative analysis, insights into the learning progressions of novice data modellers of different abilities were explored.

This thesis presents an original investigation into the transfer of knowledge and its effect on learning data modelling. The research found challenges related to task-specific knowledge within the problem domain, as well as the taught modelling concepts and knowledge acquired through completing tasks in class. The thematic analysis revealed 'fragile knowledge' of concepts, terminology, and modelling syntax as the main challenges for the novices. Other difficulties included case study comprehension, lack of verification of design decisions and inability to recall and transfer on-task knowledge. Common tactics used by novices also emerged through the thematic analysis. The students who demonstrated inconsistent problem-solving processes and tactics tended to have weak or failed transfer. The narrative analysis provided an understanding of how the novices in this study constructed their knowledge. A unique visual representation was developed to explain and aid the interpretation of learning progressions.

The findings of this study emphasise how fragile knowledge can affect novices regardless of their performance level. One student showed no progress due to fragile knowledge, while other students showed evidence of progression and learning with the support of a more knowledgeable other and redirected tasks, but not alone. This highlights that in addition to cognitive development, a sociocultural approach plays a pivotal role in progressing the learning of novice database modellers. Moreover, the results show that structure segmentation of the problem and an

intervention model that isolates elements and progressively scaffolds the learner can be effective in promoting learning. The insights gained from this doctoral research strongly suggest a need for changes in the pedagogies for teaching database modelling and highlight the need for further empirical research in the area.

Table of Contents

Abstract.....	i
List of Figures.....	vii
List of Tables	xii
Attestation of Authorship.....	xiv
Co-authored Works.....	xv
Dedication.....	xvi
Acknowledgements.....	xvii
Ethics Approval	xix
Chapter 1 Introduction.....	1
1.1 Background and Scope.....	1
1.2 Motivation and Objectives.....	4
1.3 Research Questions.....	5
1.4 Structure of the Thesis	6
1.5 Contributions.....	9
Chapter 2 Related Literature.....	11
2.1 Introduction.....	11
2.2 Systematic Mapping Protocol.....	13
2.2.1 Identification of primary studies.....	14
2.2.2 Screening of papers.....	18
2.2.3 Data extraction.....	20
2.2.4 Analysis and classification.....	21
2.3 Findings.....	25
2.3.1 General findings.....	25
2.3.2 RQ1.1 – Types of research.....	27
2.3.3 RQ1.2 – Focus areas of research.....	28
2.3.4 RQ1.3 – How novice database modellers are assessed.....	38
2.4 Summary	43
Chapter 3 Research Methodology.....	50
3.1 Introduction.....	50
3.2 Research Approach	50
3.3 Ethical Considerations	52
3.4 Research Participants.....	53
3.4.1 Recruitment.....	54
3.4.2 Sampling methods.....	55
3.5 Summary	56
Chapter 4 Classification and Analysis of Database Examinations	57
4.1 Introduction.....	57
4.2 Educational Taxonomies in CSER.....	58
4.2.1 Bloom’s taxonomy and database modelling	61
4.3 Research Method	63

4.3.1	Data collection	63
4.3.2	Data analysis method	65
4.4	The Classification Scheme.....	66
4.4.1	Concept dimension.....	66
4.4.2	Question style dimension.....	68
4.4.3	Cognitive dimension	68
4.5	Results and Discussion	76
4.5.1	Topics covered.....	76
4.5.2	Question style.....	77
4.5.3	Intellectual complexity.....	78
4.6	Summary	80
Chapter 5	Measuring the Difficulty of Novice Database Modelling Tasks.....	83
5.1	Introduction.....	83
5.2	Database Metrics.....	84
5.2.1	Structural complexity metrics	90
5.3	Use of Metrics in CSER.....	90
5.3.1	Readability metrics for novice programming tasks	92
5.3.2	Measuring task difficulty in CSER	94
5.4	Protocol for Metrics Selection	94
5.5	Selection of the Metrics	96
5.6	Metrics Calculation.....	98
5.7	Data Collection Method.....	99
5.8	Data Analysis and Results.....	101
5.8.1	Correlation of metrics vs task difficulty (performance-based difficulty).....	103
5.8.2	Correlation of metrics vs task difficulty (perception-based difficulty).....	106
5.8.3	Learning hierarchy	108
5.9	Summary	109
Chapter 6	Development of Think-aloud Modelling Tasks	111
6.1	Introduction.....	111
6.2	Novice Database Design Courses at AUT	111
6.2.1	INFS601 teaching approach.....	112
6.2.2	COMP508 teaching approach	114
6.3	Task Design Process	115
6.3.1	Task overview	120
6.3.2	Task 1 and Task 2 – Analysis of the case study.....	125
6.3.3	Task 3 – Develop a conceptual model	128
6.3.4	Task 4 – Transform the conceptual model into a logical model	133
6.3.5	Normalisation problem	136
6.4	Summary	141
Chapter 7	Think-aloud: Reflexive Thematic Analysis.....	143
7.1	Introduction.....	143
7.2	Research Participants.....	144
7.3	Think-aloud Data Collection Method	145
7.3.1	In-person data collection protocol.....	147

7.3.2	The pilot study and protocol/instrument refinement.....	151
7.3.3	Transitioning to a remote protocol.....	152
7.4	Interviewer Intervention Protocol.....	154
7.5	Transcription Protocol.....	157
7.6	Researcher Subjectivity and Reflexivity.....	159
7.6.1	Researcher subjectivity.....	160
7.6.2	Personal reflexivity.....	161
7.7	Analysis Protocol.....	162
7.8	Thematic Analysis.....	177
7.8.1	Interpreting the case study.....	179
7.8.2	Problem presentation.....	187
7.8.3	Understanding the problem domain.....	188
7.8.4	Verification of design decisions.....	189
7.8.5	Fragile knowledge.....	190
7.8.6	Problem recall.....	195
7.8.7	Tools.....	196
7.9	Discussion.....	197
7.9.1	Common tactics applied by the participants.....	197
7.9.2	Difficulties and challenges faced by the participants.....	200
7.10	Summary.....	203
Chapter 8	Think-aloud: Narrative Analysis.....	204
8.1	Introduction.....	204
8.2	Research Participants.....	206
8.3	Data Analysis Method.....	206
8.3.1	Development of the learning trajectory.....	208
8.4	Morgan.....	211
8.4.1	Session 1: Analysing the case study and extracting business rules.....	211
8.4.2	Session 2: Developing a conceptual model.....	216
8.4.3	Session 3: Transforming the conceptual model into a logical model.....	221
8.5	Blake.....	226
8.5.1	Session 1: Analysing the case study and extracting business rules.....	226
8.5.2	Session 2: Developing a conceptual model.....	233
8.5.3	Session 3: Transforming the conceptual model into a logical design.....	238
8.6	Casey.....	242
8.6.1	Session 1: Analysing the case study and extracting business rules.....	242
8.6.2	Session 2: Analysing the case study and extracting business rules.....	248
8.6.3	Session 3: Developing a conceptual model.....	255
8.6.4	Session 4: Transforming the conceptual model into a logical design.....	260
8.7	Discussion of LT/P's Tasks 1-4.....	269
8.8	Narrative Analysis of the Normalisation Problem.....	276
8.8.1	Morgan's think-aloud session.....	276
8.8.2	Blake's think-aloud session.....	280
8.8.3	Casey's think-aloud session.....	284
8.9	Summary.....	289

Chapter 9 Linking Findings to Theory.....	290
9.1 Introduction.....	290
9.2 Theories of Knowledge Construction	290
9.2.1 Piaget theory	290
9.2.2 Vygotsky’s Zone of Proximal Development.....	293
9.2.3 Sfard’s theory.....	298
9.3 Cognitive Load Theory	301
9.4 Learning Transfer.....	307
9.5 Design Thinking.....	313
9.6 Novice Modelling Styles.....	315
9.7 Summary	316
Chapter 10 Conclusions and Future Work.....	318
10.1 Overview of Research.....	318
10.2 Review of Research Questions	318
10.3 Online Think-aloud Method.....	324
10.4 Implications for Teaching	325
10.4.1 Designing novice modeller tasks to promote learning.....	325
10.4.2 Self-explanation for learning	326
10.4.3 The need, or not, for CASE tools.....	327
10.4.4 Use of diagrammatic representation.....	327
10.4.5 Course design.....	327
10.5 The Impact of COVID-19	328
10.6 Limitations	329
10.7 Future Work.....	330
References.....	331
Appendix A.....	365
Appendix B.....	368
Appendix C.....	380
Appendix D.....	384
Appendix E.....	388
Appendix F	391
Appendix G.....	394
Appendix H.....	395
Appendix I	397
Appendix J.....	401
Appendix K.....	414
Appendix L	430

List of Figures

Figure 1.1 Organisation of the chapters presented in the thesis.....	7
Figure 2.1 Systematic mapping study process adapted based on the guidelines of (Petersen et al., 2015). The numbers in braces represent the sections of this chapter that detail each phase's protocol.	15
Figure 2.2 Paper screening process.....	20
Figure 2.3 Yearly distribution of CSER vs DB modelling education research. a. provides the total number of publications identified in CSER – the articles were identified using the search string ("Computer Science Education" OR CER OR CSEd) across the same six databases. b. provides the total number of publications related to database education. Note the difference in the Y-axis scale in a. and b.	26
Figure 2.4 Quinquennial distribution of the publications based on the venue type.	27
Figure 2.5 The quinquennial publications of type of research.....	28
Figure 2.6 Focus areas of research based on the percentage of publications.....	28
Figure 2.7 The quinquennial publications by focused areas of research.	29
Figure 2.8 Visualization of a systematic map between the focused area of research (x-axis) and type of research (y-axis).....	44
Figure 2.9 Focus areas of the identified articles based on the type of research for 2021 and 2022	47
Figure 4.1 Comparison between Bloom's vs revised Bloom's taxonomy.....	59
Figure 4.2 Levels of SOLO taxonomy.....	61
Figure 4.3 ERD diagram provided to extract the business rules.....	70
Figure 4.4 Tables provided for <i>Understand</i> example 2	71
Figure 4.5 Part of the EER diagram provided for <i>Apply</i> level example 1.....	72
Figure 4.6 Data table for the second <i>Apply</i> example. In the table, the prefix C = client, P = property and O = owner.....	73
Figure 4.7 Logical ERD for the example.....	75
Figure 4.8 Coverage of question styles.....	78
Figure 4.9 Coverage of revised Bloom's taxonomy's cognitive levels.....	79
Figure 4.10 Coverage of Bloom's cognitive level with question styles.....	80
Figure 5.1 Relationship between the question difficulty (y-axis) and cognitive complexity (x-axis) of the question.....	108

Figure 6.1 Extract from the full case study given in Appendix H Research Instrument – Case Study	121
Figure 6.2 The flow of the tasks in the database model development problem	124
Figure 6.3 Model answer for Task 3	129
Figure 6.4 ERD related to Task 3.1, Task 3.2 and Task 3.3	130
Figure 6.5 Model answer for Task 3.2	131
Figure 6.6 Partial ERD related to Task 3.4	133
Figure 6.7 Model answer for Task 5 in INFS601	134
Figure 6.8 Model answer for Task 5 in COMP508	134
Figure 6.9 ERD for redirected task T4.2	135
Figure 6.10 Task flow for normalisation problem	136
Figure 6.11 The data table provided for the normalisation problem	137
Figure 6.12 Dependency diagram illustrating different functional dependencies in the data table given in Figure 6.11	138
Figure 6.13 Decomposed relations in the 3NF	138
Figure 6.14 Redirected Task 2.2 where the correct answer D1= Full functional, D2 = Partial, and D3 = Transitive	139
Figure 6.15 Data table related to Task 2.3	139
Figure 6.16 Model answer for Task 3	141
Figure 7.1 Think-aloud data collection protocol	148
Figure 7.2 Overview of the ‘in-person’ think-aloud setup from the participant and interviewer perspective. A – Livescribe pen and paper used by the participant, B – A close-up of the task provided to the participant using Livescribe paper, C – Echo desktop used by the interviewer to play and view the recording of the participant’s think-aloud session.	151
Figure 7.3 Shared screens of the participant and interviewer. A – Participant’s shared screen while attempting a task, B – Interviewer’s shared screen while introducing a task	154
Figure 7.4 Transcription protocol	157
Figure 7.5 An extraction from the observational diary	165
Figure 7.6 An initial mind map developed during the coding process	167
Figure 7.7 Initial attempt at theme development	169
Figure 7.8 How the themes evolved during the review process	171
Figure 7.9 Evolving of the themes – version 2	172

Figure 7.10 Evolving of the themes – version 3	173
Figure 7.11 Evolving of the themes – version 4	174
Figure 7.12 Refined thematic map.....	174
Figure 7.13 Refined themes	176
Figure 7.14 Finalised thematic map.....	177
Figure 7.15 Incorrect use of annotations.....	181
Figure 7.16 Step-by-step doodling of Jackie	183
Figure 7.17 Jackie’s attempt to verify the conceptual design by sketching the database structure	190
Figure 7.18 Incorrect solution due to fragile knowledge. The errors are marked in red colour.	192
Figure 8.1 Sample LT/P timeline.....	210
Figure 8.2 Morgan’s annotations on the case study and presentation of the extracted information	213
Figure 8.3 Morgan’s LT/P at the end of Task 1.....	214
Figure 8.4 Morgan’s attempt at writing the business rules showing later corrections.....	215
Figure 8.5 Morgan’s LT/P at the end of Task 2.....	215
Figure 8.6 Morgan’s initial entities with attributes.....	217
Figure 8.7 Morgan’s current LT/P	218
Figure 8.8 Morgan’s initial attempt to add a relationship before (top) and after (bottom) the interviewer intervention. Red boxes indicate the errors Morgan made	219
Figure 8.9 Morgan’s conceptual model. The red box indicates the one relationship error in the model.	220
Figure 8.10 Morgan’s LT/P at the end of Task 3.....	220
Figure 8.11 Morgan’s LT/P till resolving the many-to-many relationship.....	223
Figure 8.12 Morgan’s logical model showing their resolution to the multivalued attribute problem	224
Figure 8.13 Morgan’s completed logical ERD with any remaining errors highlighted using a red box.	225
Figure 8.14 Morgan’s LT/P at the end of Task 4.....	225
Figure 8.15 Blake’s identified constructs for section 1. The text in green is the items identified during the think aloud and the edits marked in black are the changes Blake made after the retrospective interview.....	228

Figure 8.16 Blake's doodling of the identified relationships for section 2.....	229
Figure 8.17 Blake's presentation on the relationship between instructor and program. The changes in black colour indicate the changes Blake made after the retrospection.	229
Figure 8.18 Blake's LT/P at the end of Task 1	230
Figure 8.19 Blake's first attempt at writing business rules.....	231
Figure 8.20 Blake's finalised business rules.....	232
Figure 8.21 Blake's LT/P at the end of Task 2	233
Figure 8.22 Blake's initial entities and the relationship (bottom) and the referred business rule (top).....	234
Figure 8.23 Blake's diagram before (left) and after (right) the interviewer's intervention	235
Figure 8.24 Blake's conceptual model. The red boxes highlight the errors they made, which were later corrected during the retrospection. The error at the bottom indicates a syntax error: using a solid line (indicating a strong relation) instead of a dashed line (indicating a weak relation)..	237
Figure 8.25 Blake's LT/P for Task 3	237
Figure 8.26 Blake's resolution of the many-to-many relationship in the logical model.....	239
Figure 8.27 Blake's completed LT/P at the end of Task 4.....	240
Figure 8.28 Blake's logical ERD before (top) and after (bottom) the retrospection interview	241
Figure 8.29 Casey's attempt to annotate the case study.....	244
Figure 8.30 Information extracted from analysing the first section of the case study	245
Figure 8.31 Casey's presentation of the relationships	246
Figure 8.32 Casey's representation of the relationship between <i>Instructor</i> and <i>Program</i>	246
Figure 8.33 Casey's LT/P at the end of Task 1	247
Figure 8.34 Casey's first attempt at writing the business rules.....	249
Figure 8.35 Newly added business rule in the red box	250
Figure 8.36 Casey's attempt on the redirect task – T2.1.....	250
Figure 8.37 Casey's LT/P upto T2.1	251
Figure 8.38 Casey's second attempt at writing the business rules	252
Figure 8.39 Casey's LT/P at the end of Task 2.....	254
Figure 8.40 Casey's LT/P during the first half of the session.....	257
Figure 8.41 Incorrect relationships vs business rules.....	258

Figure 8.42 Casey's learning trajectory at the end of Task 3.....	259
Figure 8.43 Casey's conceptual model before (left) and after(right) the feedback. Red boxes indicate the errors in their solution.	260
Figure 8.44 Casey's LT/P during their first attempt on Task 4	262
Figure 8.45 Casey's LT/P halfway through the session.....	265
Figure 8.46 Casey's completed logical model. The red boxes highlight the remaining errors in their model.	267
Figure 8.47 Casey's LT/P at the end of Task 4.....	268
Figure 8.48 Morgan's solution to Task 2. The red box indicates the error in their solution.....	277
Figure 8.49 LT/P of Morgan during the normalisation problem	280
Figure 8.50 Blake's incomplete answer for Task 2 (identification of dependencies).....	281
Figure 8.51 Blake's dependency diagram.....	282
Figure 8.52 Blake's decomposed relations in 3NF	283
Figure 8.53 Blake's final solution.....	283
Figure 8.54 LT/P of Blake during the normalisation problem.....	284
Figure 8.55 Casey's justification for selecting the primary key composed of S_ID, C_ID, L_ID	285
Figure 8.56 Casey's partial solution for Task 2	286
Figure 8.57 Casey's second attempt on Task 2.....	287
Figure 8.58 Casey's attempt to decompose the relations	288
Figure 8.59 LT/P of Casey during the normalisation problem	289

List of Tables

Table 2.1 Search queries used in digital libraries.	17
Table 2.2 Inclusion (I) and Exclusion (E) criteria.....	18
Table 2.3 Data extraction form	21
Table 2.4 Type of research classification scheme (Adopted from (Taipalus & Seppänen, 2020)).	22
Table 2.5 Focus areas of research in database modelling education.....	23
Table 2.6 Articles identified through snowballing.....	24
Table 2.7 Assessment of novice modellers.....	25
Table 2.8 Approaches and practices proposed to assess novice database modellers.	38
Table 3.1 Overview of the utilised research methods.	51
Table 4.1 The revised Bloom’s taxonomy’s cognitive levels in domain modelling.....	63
Table 4.2 Examination Demographics.....	64
Table 4.3 Inclusion (I) and Exclusion (E) criteria developed for question screening.....	65
Table 4.4 Type of concepts in database modelling questions.....	67
Table 4.5 Topics coverage and percentage marks by topic.	77
Table 5.1 Comparative analysis of metrics in to measure the quality of database designs.....	86
Table 5.2: Selected and eliminated metrics.	97
Table 5.3 Interpretation of Flesch-Kincaid Grade Level.	99
Table 5.4 Questions selected for analysis.	100
Table 5.5 Complexity metrics for instructors’ model and difficulty of questions.	102
Table 5.6 The correlations between performance-based difficulty and metrics.	103
Table 5.7 The correlations between performance-based difficulty and metrics for different types of tasks.	106
Table 5.8 The correlations between perception-based difficulty and metrics for similar types of tasks.	107
Table 6.1 Main topics and key concepts covered in the INFS601. The abbreviations are as follows: MA – Midterm Assessment; N&LDD – Normalisation and LDD Assessment; DBP – Database Project.	113
Table 6.2 Overview of the tasks.....	122

Table 6.3 Steps required to solve the tasks.	125
Table 6.4 Model answer for Task 1.1.	127
Table 7.1 Background information of the think-aloud participants.	145
Table 7.2 Interviewer intervention protocol.....	156
Table 7.3 Adopted transcription guidelines.	159
Table 7.4 Overview of themes.	178
Table 7.5 Annotation systems developed by the participants.	182
Table 7.6 Identified difficulties and challenges mapped to the existing literature.	203
Table 8.1 Reasons for providing the assistance.	211
Table 8.2 Summary of interviewer interventions, time on task and errors made by the participants across the sessions.....	273
Table 9.1 Transfer types.	309

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature

Date

Co-authored Works

- Imbulpitiya, A., Whalley, J., & Senapathi, M. (2024). A Qualitative Study of Novice Experiences with Database Normalisation. Proceedings of the 26th Australasian Computing Education Conference (ACE '24). Association for Computing Machinery, New York, NY, USA, 94–103. <https://doi.org/10.1145/3636243.3636254>
- Imbulpitiya, A., Whalley, J., & Senapathi, M. (2023). Reflections on Conducting Online Think-Alouds. *ACM Inroads*, 14(2), 26–34. <https://doi.org/10.1145/3596917>
- Imbulpitiya, A., Whalley, J., & Senapathi, M. (2021). Examining the Exams: Bloom and Database Modelling and Design. *Proceedings of the Twenty-Third Australasian Conference on Computing Education*, 21–29. <https://doi.org/10.1145/3441636.3442301>

Dedication

අසමත් වීම යනු පරාජය නොවන බවත්...

එය ඉදිරි ගමනට ලද සවිමත් පන්තරයක් බවත්...

පසුබැස්ම යනු ගමනක අවසානය නොවන බවත්...

මා හට පසක් කල සදාදර අම්මා සහ අප්පච්චිට

තැනි සිනින් පුදමි !

To my parents,

Who taught me that failure is not defeat,

But an opportunity for the growth to meet

Who taught me that set backs are not the end,

But stepping stones on which I ascend!

~ අසන්තිකා / Asanthika ~

Acknowledgements

The journey to complete this thesis could have been the hardest manoeuvre I have ever done in my life. It was not a walk through a bed of flowers by any means, not only to me but to everyone around me. So, I would like to take this opportunity to convey my heartfelt gratitude to everyone who was a part of this journey and was there with me through thick and thin. I wouldn't be writing this if it wasn't for the amazing support I've received from all of you.

To Professor Jacqueline Whalley, thank you for inspiring me every day. From the first day you replied to my email to our first meeting, and to date, you have become an amazing mentor, more than any student could ask for. Thank you for your patience, motivation, the abundance of knowledge and, above all, believing in me. I shall always treasure our discussions, the impromptu meetings we did whenever I struggled and did not know which door to take and the guidance you provided. Thank you for always being there when it was needed the most and teaching me every nook and cranny of being a researcher with integrity.

To Dr Mali Senapathi, thank you for your constant encouragement, support, and guidance. I always appreciated the domain expertise you brought into our discussions, bringing new perspectives. Your gentle smile, kind demeanour, and the simple *"You got this!"* carried me through some tough times during this journey. I am grateful to both of you for walking with me because there wouldn't be a thesis if it weren't for you both.

I also would like to thank AUT's School of Engineering, Computer and Mathematical Sciences for awarding me a PhD and a stipend scholarship, which immensely lessened the financial burden and helped me focus more on my work. I have met amazing researchers and academics while being a part of the Software Engineering Research Centre. Thank you for welcoming me with open arms all these years ago. I would like to particularly thank Associate Professor Tony Clear for always extending a helping hand, whether it was a request for feedback or simply a question out of curiosity. Thank you to all the colleagues I've met during my studies. You made this life a little bit easier, especially Waruni, Priyanka, and Barry, for the endless coffee and constant encouragement. Furthermore, thank you to all the staff at CMS PhD Team, AUT library, GRS, and ICT services team for your valuable support.

A special gratitude goes out to all the participants who volunteered to participate in my study. As an academic, you inspired me with your resilience and devotion to continue amidst the pandemic and lockdowns. A special thank you to my friends Sachinthana and Ajith for helping me out with the statistical analysis of the data.

Last but not least, thank you to all my friends and family for your support and constant encouragement. A special thank you to both my parents, Saliya and Kumari, for showing me that nothing is impossible once you set out to achieve a target. I still resonate with what they said when I left Sri Lanka, *“It will be hard, but we will be there every step of the way and cheering you on once you reach the end of the tunnel”*. Thank you for being my cheering squad across the seas, Taridu, Kavidu, Rangi and Thaksi.

Asanthika Imbulpitiya, August 2023.

Ethics Approval

Ethical approval to conduct the research reported in this thesis was granted by the Auckland University of Technology Ethics Committee (AUTEC) on the following dates. The ethics approval letters are included in Appendix A.

Reference number 20/167 – Granted on 31st July 2020.

Reference number 20/167 - Granted on 15th December 2020.

Reference number 21/24 - Granted on 11th March 2021.

Chapter 1 Introduction

1.1 Background and Scope

Modelling is an essential skill that any undergraduate should acquire following a Computer Science (CS) or related undergraduate degree. Models that are created during different phases of software development can play a prominent role in transferring requirements to real working software. In Software Engineering, there are two main paradigms of modelling: one related to the object-oriented modelling of the software and the other related to data modelling and creating databases that manipulate the data of the software. Database modelling is considered to be a mandatory course for many CS and related undergraduate degree programs. Database modelling is an integrated set of concepts to describe and manipulate data, relationships and constraints and is an integral part of the database design phase of the Database Life Cycle (DBLC) (Coronel & Morris, 2016). Wagner (2005) highlighted why data modelling is important for students as a curriculum area, claiming that it lays the foundation for other database-related topics such as querying, creating database systems from a model, and database administration. Consequently, many introductory database courses are taught in the first year of an undergraduate degree. Further, the importance of introductory-level database courses is reflected in their inclusion as a core module in the IEEE/ACM curricula, not only in CS degrees but in Information Technology, Information Systems, and Data Science (DS) degree programs (ACM Data Science Task Force, 2021; CC2020 Task Force, 2020; Topi et al., 2010).

These curriculum guidelines, regardless of the degree program, recommend similar foundation courses that should cover, among other topics, conceptual data models (including Entity Relationship Diagrams [ERD] and modelling grammars) and relational data models (including design and normalisation). The development of such models involves working with unstructured problems that require students to deal with concepts of a high level of abstraction. In a recent article, Goldweber et al. (2022) discuss what needs to be taught in undergraduate database courses. They delve into several issues within the curricula, mainly the less emphasis placed on data management courses in general, which led to the overload of topics that need to be taught, exceeding the bandwidth of a single introductory-level course. Another complicating factor is the curricular viewpoint of the institute where they placed CS on the spectrum between pure science and a profession, which again brings the question of what to teach. Throughout their discussion, they focused on four case studies across different degrees, including BA and BSc in CS, DS and Computer Engineering. In all the case studies, they explained the consistent use of the relational model, which further emphasised the importance of database modelling as a competence requirement for novices. Further, in their fifth case study, industry personnel from Microsoft mentioned data modelling (i.e., relational data models, JSON extension) as a key concept to

be taught in undergraduate database courses. The limited space in the curricula against the number of concepts and topics that need to be covered to meet the required learning outcomes and the abstract nature of these concepts results in a significant challenge for educators teaching the subject area and the students learning database modelling.

At the time this research started, there was very little work in the academic literature on the teaching and learning of database modelling despite its importance. Interestingly, no systematic reviews were conducted at the time to gain an overall understanding of the existing work, which led to the Systematic Mapping Study (SMS) presented in Chapter 2. There was literature that pointed to the fact that learning to model is challenging, but there was little understanding as to why novices find it difficult or how they learn to design. These articles point to several barriers faced by novices, like outdated teaching methods (Wang & Chen, 2014), unrealistic case studies that are not relatable to the real world (Abid et al., 2015), the uncertainty of students as to why a model is required (Chilton, 2006), less practical assessments (Wang & Chen, 2014), and learning number of highly interrelated concepts (Katz & Shmallo, 2016).

Recent research by Abid et al. (2015) discusses the impact of using simple outdated examples, such as a library management system, on student engagement. Because a traditional library system with librarians checking out books is not how current libraries typically operate, such case studies are not relatable in the current context of the students' world. This lack of relevance makes the students passive and less than thrilled about learning the content.

Understanding the high number of interrelated abstract concepts in database design presents one of the main challenges for novice database modellers (Amin et al., 2019; Katz & Shmallo, 2016). In a typical introductory level database course, students are required to learn concepts such as conceptual model, relational database model, SQL, normalisation rules, etc. Database modelling is a process with a high level of concept interactivity. Therefore, understanding one concept in isolation is not sufficient since each topic is understood and learnt with reference to another. The need to process these concepts simultaneously is the major reason why database modelling is known to be a complex cognitive process for novices. Due to these difficulties, novices often find it difficult to think like expert database modellers (Watson, 2006).

There has been little research on the teaching and learning of database modelling. For the most part, this research related to teaching and learning database modelling has focused on associated pedagogy (Abid et al., 2015; Fang et al., 2017; Mason et al., 2016; Podeschi, 2016; Wang & Ma, 2017) and related tools (Bock & Yager, 2005; Goelman & Dietrich, 2018; Milik et al., 2006a; Mitrovic et al., 2004; Mitrovic & Suraweera, 2016). This is further confirmed by a recent 2019 paper which reviewed the

literature on the teaching and learning of databases in computer science, where the authors discovered only 35 articles that were relevant and within those identified three main themes: student difficulties, methods to improve teaching and learning, and tools (Hamzah et al., 2019).

Several researchers have discussed the struggles of novice database modellers and associated this with the cognitive complexity of tasks associated with the database design phase (Dinesh Batra, 2007; Katz & Shmallo, 2015). Due to this, it is a critical consideration of any pedagogy applied in teaching and learning database modelling to reduce this cognitive load. A number of pedagogic approaches to teaching novice database modellers have been proposed that were, in essence, intended to reduce the complexity of the topic area for learners. Watson (2006) proposed a “spiral model” in which teaching relational database design and SQL is essentially interwoven. Bite-sized design concepts are taught and closely followed with the associated SQL and database implementation steps. He argued that when the topics were taught in blocks and design was taught first, and then SQL students failed to understand core concepts like foreign keys because their knowledge was not consolidated. By teaching design and SQL in a spiral mode, learning abstract design concepts is supported with the more concrete SQL implementation and querying of a database.

Katz and Shmallo (2016) introduced a “learning from errors” (LFE) approach to help novices make the necessary transitions between abstraction levels when designing a relational database model. They took the novice design errors observed in their prior work (Katz & Shmallo, 2015) and used those errors to inform the design of formative exercises and infuse examples with these errors as teaching points. These errors are intended to trigger the cognitive dissonance (Festinger, 1957) that is necessary for learning and building appropriate mental models or schema. The authors go on to discuss the importance of expert schema and their role in the cognitive processes involved in database design, noting that novices lack these long-term mental schemas. In a later study, Katz (2018b) adopted the four-level TSSL (Task, Semantics, Syntax and Lexicon) user activity model to guide students through conceptual data modelling. The aim was to lower the extraneous cognitive load associated with learning database schema modelling. Their article provides a useful worked example of how to apply the TSSL lens to database modelling learning activities. Anecdotally, the teachers using this approach noted an improvement in student-generated designs. Katz (2020) then furthered this work by evaluating the TSSL approach in a controlled experiment where the experiment group was exposed to their TSSL visually guided approach, and the control group was not. They found that the TSSL group, made fewer errors, provided higher quality solutions and had more positive perceptions of the subject area.

Database design appears to be particularly problematic for students from Non-English Speaking Backgrounds (NESB). This difficulty has been linked to learning materials and assessments and their presentation. The argument is that the way such materials are typically presented increases the load on

the working memory of NESB students (Mason & Seton, 2021). Mason and Seton present a set of guidelines for presenting assessments that address aspects of cognitive load and show through an empirical evaluation that this type of treatment levels the playing field and benefits all students. However, the authors acknowledge that further investigation is needed to confirm these findings.

Self-regulation is also believed to have a positive effect on learning. One study looked at the role of different types of prompts (e.g., self-explanation, knowledge integration prompts, procedural prompts, etc.) on the ability to complete database modelling tasks (Lin et al., 2015). They reported that the self-explanation group scored lower than other treatment groups on a post-intervention test designed to test database conceptual knowledge. The notion that learning database design is complex, with multiple interrelated and abstract concepts, and imposes a high cognitive load on learners resonates throughout the literature in the area. However, little research has been published addressing why novices find database modelling difficult and how these difficulties map to issues in their knowledge structures. Research is needed that empirically investigates when students encounter barriers to learning and how they progress their learning in the context of database modelling. Hence, this thesis proposes a study that follows novice database modellers through their introductory database courses with the aim of understanding how students integrate and accommodate new elements and concepts to build their knowledge as novices of creating database models that they encounter when traversing through the DBLC.

1.2 Motivation and Objectives

As discussed in section 1.1, the main focus of teaching and learning database modelling is related to a few focused areas. Hence, the **first** objective of this thesis is to develop a broader systematic review to understand the breadth of the existing literature.

Understanding the nature of the tasks given to novices is essential before observing them to comprehend how they learn. Several studies have used educational taxonomies like Bloom's taxonomy of educational objectives (Bloom et al., 1956), the revised Bloom's taxonomy (Anderson et al., 2001) and SOLO (Biggs & Collis, 1982) to frame their research and explain their findings. In particular, these taxonomies have been used to describe the nature of programming tasks (Sheard et al., 2013; Thompson et al., 2008; Whalley & Lister, 2009). Mohtashami and Scher (2000) outlined the importance of using Bloom's taxonomy to develop suitable formative and summative assessments for their database design course. They do not discuss how they interpreted the Bloom taxonomy or how they classified the assessment tasks. More recently, Bogdanova and Snoeck (2019) used the revised Bloom's taxonomy to classify learning outcomes which they derived from educational resources related to Unified Modelling Language (UML) in object-oriented analysis and modelling. Although UML and object-oriented

analysis is not the same as developing database models, both have similar teaching and learning requirements and challenges. Following the work of Bogdanova and Snoeck (2019), the **second** objective of this research is to develop a question classification framework to understand the nature of database modelling tasks currently given to novices.

Further, as its **third** objective, this doctoral study aims to identify an objective measurement to assess the difficulty of novice tasks using the structural complexity metrics related to database models. Although the existing studies have begun the discussion related to modelling assessments and skills, no research has looked at the possibility of a hierarchical structure of mastery of skills in modelling. Earlier work has established that there is a hierarchy of skills in learning to program (Lopez et al., 2008; Pelchen et al., 2020; Philpott et al., 2007; Whalley & Lister, 2009). Intuitively, it makes sense that there is some dependency in building models that describe the technical requirements of the software, including databases. Therefore, the **fourth** objective of this research is to investigate whether a hierarchy of skills also exists for novices learning database modelling using a causative statistical approach (similar to a manner in work conducted in programming by Lopez et al. (2008)).

This research also investigates areas in which students struggle in terms of creating database models and how they overcome these issues as novices. Further, the individual Learning Trajectory/Progression (LT/P) of students provides an understanding of situations where students managed to progress, how they were unable to progress and what barriers hindered this process where they couldn't integrate this new knowledge into that development. Hence, the aim is to map the learning trajectories for each of the students to relate them with the hierarchical structure, if it exists, using the think-aloud data. The **fifth** and final objective of this research is to triangulate between quantitative analysis of a large group of assessments from a cohort of novices using naturally occurring data with that of a qualitative think-aloud study that follows a small group of novice students through their introductory course of learning database modelling. Naturally occurring data can be defined as data that “already exists by virtue of the examination process and simply requires collection and analysis” (Clear et al., 2009, p.157).

Ultimately, the main overarching objective of this study is to better understand the learning processes and the developmental stages of novice modellers and how they build their knowledge as novices. This understanding is intended to allow educators to teach and assess database modelling more effectively.

1.3 Research Questions

Based on the objectives described in the previous section, the following research questions have been developed to gain a deeper understanding of how novices learn to model, emphasising their cognitive development process.

RQ1: What are the existing focus areas of research in database modelling education?

RQ1.1: What types¹ of research are presented in the identified publications?

RQ1.2: What areas of introductory database design and modelling are the focus of the literature?

RQ1.3: How are novice database modellers assessed?

RQ2: Can a question classification framework be established that describes the difficulty of novice database modelling tasks?

RQ2.1: What is the nature of database modelling tasks used to assess novice database modellers?

RQ2.2: What type of measurements can predict the difficulty of database modelling tasks?

RQ2.3: Does the performance of students in database modelling tasks indicate a hierarchical structure in their mastery of different types of novice modelling tasks?²

RQ3: How do novice database modellers incorporate new modelling concepts and knowledge into their existing understanding of database modelling?

RQ3.1: What difficulties do students face when attempting novice database modelling tasks?

RQ3.2: What common tactics do students apply when attempting novice database modelling tasks?

RQ3.3: How does the learning of novice database modellers of different capabilities develop over time?

1.4 Structure of the Thesis

This thesis is organised into ten chapters, and this chapter provides an overview of the thesis, including an introduction outlining the background literature, the research aims and the contribution of the thesis. The rest of the thesis is structured as follows, and the organisation of the chapters are illustrated in Figure 1.1.

¹ Types of research, as defined by Petersen et al. (2008) reflects the research approach used in the research in general and independent from a specific focus area.

² Even though a hierarchical structure of mastery of different types of tasks was hypothesised, it was not observed within the analysed data. Hence these findings are presented in Appendix K.

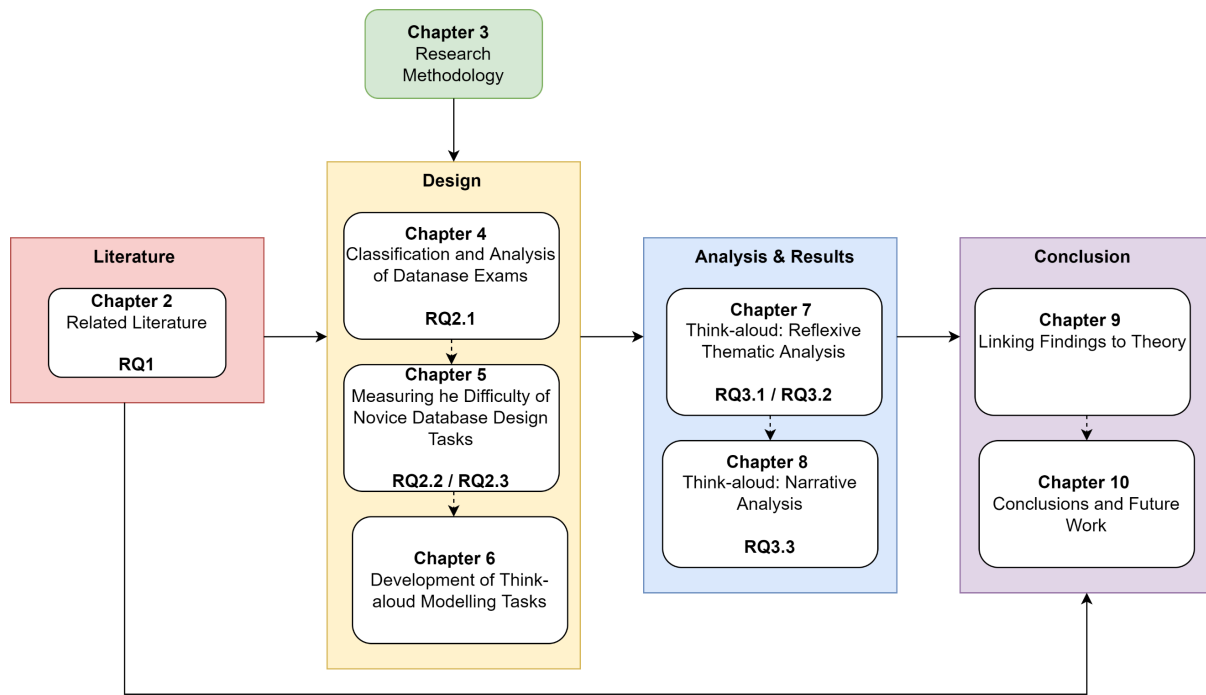


Figure 1.1 Organisation of the chapters presented in the thesis.

Chapter 2 presents the literature review conducted at the commencement of this doctoral research as a systematic literature mapping study. The study was conducted to explore the breadth of published work in the area of learning and teaching novice database modelling. To the researcher's knowledge, no such reviews had been published at the time. The aim of this SMS was to identify the focus areas of literature in novice database modelling over the last 30 (from 1990 to 2020) years. Because this thesis required a research instrument for novice modelling tasks, it was important to get a clear understanding of the existing approaches of assessments to develop the research instrument that's utilised for the think-aloud interviews. The articles related to the broad area of assessment were then investigated in-depth in order to identify any existing approaches and practices used to assess novice database modellers. For this research, the chapter concludes with an addendum that presents the subsequently extended literature review from 2021 to 2022.

Chapter 3 discusses the overarching methodology adopted in this research study, including an overview of the various research methods, utilised in different phases of the study to collect and analyse data. The intent here is to provide an overview and guide. The details/protocol of each method used is provided in the chapter that represents the body of work that utilises that method.

Chapter 4 presents the question classification framework developed to analyse the nature of the introductory-level database examination questions. It was essential in this study to understand the nature of the novice database tasks before diving in and designing the tasks to be utilised in the study. This

chapter introduces the existing mechanisms used to analyse the nature of tasks, followed by data collection using naturally occurring data, the utilised ethics process, and the open coding data analysis method. Then, it describes the classification scheme used to analyse the questions based on the concept(s) covered by a question, the question style (e.g. multiple choice, true/false, etc.), and the intellectual complexity of the question along with the research method utilised to collect the data. The chapter concludes by summarising the shortcomings of the existing questions used in assessing novice database modellers.

Chapter 5 presents the quantitative analysis conducted in this thesis with the intention of developing a set of metrics that can assess the difficulty of the database modelling tasks. It starts by analysing the existing database and difficulty metrics used in Computer Science Education Research (CSER), followed by the protocol used for metric selection and the selection of metrics. Then, the chapter discusses the method used to evaluate the metrics, including the research participants, the ethics process, the data collection method using naturally occurring data and the statistical data analysis method, where statistical analysis like correlation tests, multiple regression tests and principal component analysis was used. Following that, the chapter discusses a potential set of structural and readability metrics to identify the difficulty of database modelling tasks. A statistical correlative evaluation of the metrics in terms of their suitability to predict task difficulty as measured by student performance on a task is presented.

Chapter 6 focuses on the design of the modelling tasks utilised in the think-aloud studies. The design is informed by the findings described in Chapter 4 and Chapter 5. This chapter starts by addressing the context of the think-aloud study in terms of the introductory-level database courses INFS601 (Logical Database Design) and COMP508 (Database System Design), from which the participants were recruited. Then, it discusses the task design process which was utilised during the think-alouds.

Chapter 7 presents the reflexive thematic analysis conducted to identify the common patterns of challenges and difficulties faced by novice modellers, along with the tactics applied in attempting the tasks. The chapter starts with a detailed description of the ethics process, recruitment process, and the participants, followed by the protocol for the think-aloud data collection and the reflexive thematic analysis. Then, it details the researcher's positioning, transcription and analysis protocols. The chapter then presents the narrative analysis by theme and finally ends with a discussion of the tactics adopted by the participants and the challenges that the participants encountered.

Chapter 8 presents the analysis of the same think-aloud data used in Chapter 7 from a different perspective. This chapter details the narrative analysis of three participants (best, middle and lowest-level performing participants) and maps this to a visual representation of each participant's LT/P to comprehend their learning progression throughout the sessions.

Chapter 9 relates the findings discussed in Chapter 7 and Chapter 8 to existing educational theories and cognitive theories of learning. The chapter then goes on to explore how novice modelling tasks can be designed, based on the identified theoretical aspects, to enhance approaches to teaching and learning.

Chapter 10 concludes this thesis by discussing the limitations of the study, the conclusions of the findings, and the future directions of the research into teaching and learning of novice database modellers.

1.5 Contributions

Based on the work reported in the literature at the start of this study, there is a limited understanding of how novices learn to model and the cognitive development processes that trigger this learning in novice modellers. The doctoral research proposed herein aims to investigate novice database modellers to gain a deeper understanding of their approach to learning database modelling, with a focus on the growth of their cognitive processes. This research conducts an in-depth investigation into understanding how novice database modellers learn to model. In order to conduct this work, it was required to understand how to build an instrument and the relative difficulty of modelling tasks to structure the study in a meaningful manner. The contributions of the study are summarised as follows:

- Conduct a comprehensive systematic mapping study of the literature in database modelling education to lead to an understanding of the existing work.
- Develop a question classification framework to understand the nature of novice modelling tasks based on the task style, content covered and the cognitive complexity of a task.
- Develop a framework to predict the difficulty of typical database modelling tasks using database metrics, which didn't yield the expected results but provided an indication that it is possible to develop such a measure with some of the identified metrics and use more data points.
- Examine the notion of a hierarchical structure in the mastery of database modelling skills based on novices' performance. Even though the hierarchy was not established within the analysed data, this work provided an initial framework for other researchers to inform further studies to investigate the idea of a hierarchical structure in mastering different types of tasks.
- Conduct the first qualitative investigation of novice database modellers to the best of the researcher's knowledge using a reflexive thematic analysis to have a more in-depth and respondent-led study. The intention was to gain a deeper understanding of how students approach learning based on the source of student difficulties, tactics they have developed, and the type of exercises and scaffolding of exercises that support student learning. to gain more insight.

- Conduct a qualitative think-aloud study with a series of interconnected tasks across multiple sessions compared to the think-aloud often conducted in this domain using a single task at a snapshot.
- Form a visual representation of the learning progression and trajectory, providing an easier interpretation format.
- Develop a protocol to conduct think-aloud virtually in the CS education context.

Chapter 2 Related Literature

2.1 Introduction

Almost all undergraduate degree programs in computing include an introductory level course on database management systems, often followed by more advanced courses in the domain. An integral topic area identified in these introductory level courses is that of database modelling and design. The importance of such courses is reflected in its inclusion as a core module in the IEEE/ACM curricula. The jointly initiated ACM/IEEE/AIS computing curricula developed in 2005 (CC2005) listed modelling and designing a database as one of the graduate outcomes for programs in computing regardless of the discipline (Shackelford et al., 2006). The next iteration of the IEEE/ACM computing curricula developed in 2020 also includes data and information management as a competency requirement in computing, particularly in Computer Science (CS), Information Systems (IS), and Information Technology (IT) disciplines (CC2020 Task Force, 2020). This confirms that teaching database modules to undergraduates in computing is still relevant despite the passage of time. The 2013 ACM curriculum guidelines for Undergraduate Degree Programs in CS, by the Joint Task Force on Computing Curricula (2013), consider Information Management as one of the prominent knowledge areas in computer science and suggest a core module that includes conceptual data models (Entity Relationship Diagrams [ERD], modelling grammars, etc.) and relational data models (design and normalisation) among other topic areas. Similarly, the 2010 ACM Curriculum Guidelines for Undergraduate Degree Programs, IS 2010.2 recommends a Data and Information Management foundation course (Topi et al., 2010). Further, the 2020 IEEE/ACM computing curricula included DS as an emerging curricular model and was presented as a curriculum under development at the time (CC2020 Task Force, 2020). Later, in 2021, the ACM Data Science Task Force published a report highlighting the computing competencies for undergraduate DS curricula, including a course in databases and data management consisting of traditional relational databases (ACM Data Science Task Force, 2021). Moreover, they highlighted the importance of learning about various relational and other database formats across the eleven knowledge areas recommended for a data science degree. Many contemporary undergraduate degrees in CS, IS, and DS still include data modelling and database design as part of their core curriculum. The consistent inclusion of database modelling highlights its role as an important skill that any student following a computer science or related degree should master at an early stage of their studies.

While database modelling is acknowledged as an important foundation topic in most computing undergraduate degrees, students often find learning database design difficult. Educators also find it harder to teach certain concepts that are abstract in nature. Several researchers have discussed the

challenges in teaching and learning database modelling and link this to modelling being a complex topic that is abstract in nature and consists of a high number of interrelated concepts (Amin et al., 2019; Badia & Lemire, 2011; Hussain, 2017; Katz, 2018b; Watson, 2006).

Despite the accepted importance of students learning database modelling and the reported difficulties in the area, there is little published empirical research aimed at improving the teaching and learning of novice database designers, as evidenced by the systematic mapping included herein. At the time of this review in mid-2020, which helped to frame the research undertaken for this thesis, there had only been one attempt to identify and synthesise the research in database modelling education. This paper reported a comprehensive review of the area and was published in *Revista Espacios*, where the articles are published in Spanish, Portuguese and English and not a commonly known venue in computer science education. The paper by Hamzah et al. (2019) reviewed the literature on the teaching and learning of databases in computer science. The authors appear to have used a systematic approach, but the publication is a short paper where the methodology lacks the details necessary to replicate their study. The parameters for their search were not provided however, all the papers mentioned in their article were between 2007 and 2018. Over the ten years, the authors reported discovering only 35 articles that were relevant. These 35 papers were identified as belonging to two main themes in the literature: student difficulties and methods to improve teaching and learning. Under the second theme, they have mainly identified approaches such as blended learning, flipped classroom approach, cognitive load theory approach, active learning techniques and computer-based instructions presented in the literature. It was surprising to see only 35 papers over ten years related to only two themes, but the analysis therein further supports our claim of the gaps in the domain.

Considering the lack of reviews of the area, it was decided to conduct a full Systematic Mapping Study (SMS) to understand the literature on the teaching and learning of database modelling. The main objective was to build an understanding of the span and focus of the existing literature and identify any gaps in the literature related to the teaching and learning of databases. Further, to conduct this research, it was required to develop database modelling tasks that assess and measure the existing knowledge of novices and get them to think aloud while attempting the tasks. Hence, it was important to understand how database concepts were typically assessed to conduct this research. Due to that, after the SMS, the area of assessments was explored more in-depth to comprehend the existing work in this area.

The SMS was conducted to answer the following research question and associated sub-questions:

RQ1: What are the existing focused areas of research in database modelling education?

RQ1.1: What types of research are presented in the identified publications?

RQ1.2: What areas of introductory database design and modelling are the focus of the literature?

Further, an author collaboration network was developed to understand how researchers have collaborated with each other in the area for the past three decades. While this analysis was not central to the thesis, it builds a picture of what the current research landscape looks like in the area of database education. A detailed analysis of the author collaboration is provided in Appendix I.

This chapter presents the entire systematic mapping protocol used in this research study to capture the published literature between 1990 and 2020, followed by the findings of the study and a discussion. The findings are discussed in detail based on the dimensions of types of research, focus areas of research, practices, and approaches proposed to assess novice database modellers. The addendum of this chapter discusses the subsequently repeated search at the time of writing the thesis to include the literature published during 2021 and 2022.

2.2 Systematic Mapping Protocol

Systematic mapping studies are a means of getting an overall picture of a research area based on a classification scheme and counting the research contributions for each category arising from the classification scheme. Even though systematic mapping studies started to appear around 2008 in CSER, they have mainly been used in software engineering research. The protocol presented here is based on the systematic mapping study protocol introduced by Petersen et al. (2015) for conducting a systematic mapping study in software engineering.

The protocol developed for this study was a step-by-step adoption of the steps used by Petersen et al. (2015) in their study. The main deviation from their protocol is the removal of their exclusion criterion, “quality assessment”. This decision was made retrospectively after screening the papers and at the time of full-text evaluation. The reason was the low number of papers identified and the high number of these selected papers that would have been excluded based on the “quality assessment” criterion. This decision was deemed reasonable as despite conducting a quality assessment in their study, Petersen et al. (2015) stated that *“quality assessment should not pose high requirements on the primary studies as the goal of mapping is to give a broad overview of the topic area”* (p. 12). Otherwise, the steps of the protocol remained the same. Several procedural decisions, specifically when handling the search strategy, were decided based on the issues discussed in the study conducted by Brereton et al. (2007) and on the lessons they learnt when applying a systematic review process to the software engineering domain. As one of their suggestions the research questions were reviewed with the progress of the mapping study which led to include an additional research question to identify how novice database modellers are assessed considering the lack of data identified during the mapping study related to assessing novices.

RQ1.3: How are novice database modellers assessed?

Further, conducting a pilot study to evaluate the systematic mapping protocol was also added based on their suggestions. More specifically, multiple search strings were developed and tested as suggested by (Brereton et al., 2007) based on the search model of each database to identify the appropriate one to address the research questions. An overview of the phases of the SMS conducted as part of this doctoral research is depicted in Figure 2.1.

2.2.1 Identification of primary studies

Choose the search strategy

Petersen et al. (2015) discussed three different search strategies (i.e., snowballing, manual search and database search), of which database search is the most frequently used approach. They recommend selecting one as conducting multiple search strategies can be quite a time-intensive task. Considering the time and the goal of achieving an overview of the study area, a database search was used to identify the potential studies. Six digital libraries were used to search for specific keywords relating to teaching and learning novice database modelling and design. Initially, it was decided to search the ACM (Association for Computing Machinery) Digital Library, IEEE Xplore, Springer Link, Science Direct and Taylor & Francis digital libraries since these databases included high-impact CSER journals and conference proceedings like ACM Transactions on Computing Education (ToCE), Computer Science Education, the ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE), the SIGCSE Technical Symposium (SIGCSE TS), and the ACM Conference on International Computing Education Research (ICER). Later, it was decided to expand the search to include Information Systems venues that were focused on databases by including the AIS elibrary when an initial exploratory search showed that some database education articles appeared in the Information Systems literature. Another more pragmatic reason for selecting these databases was the ease of accessibility provided via our university library system.

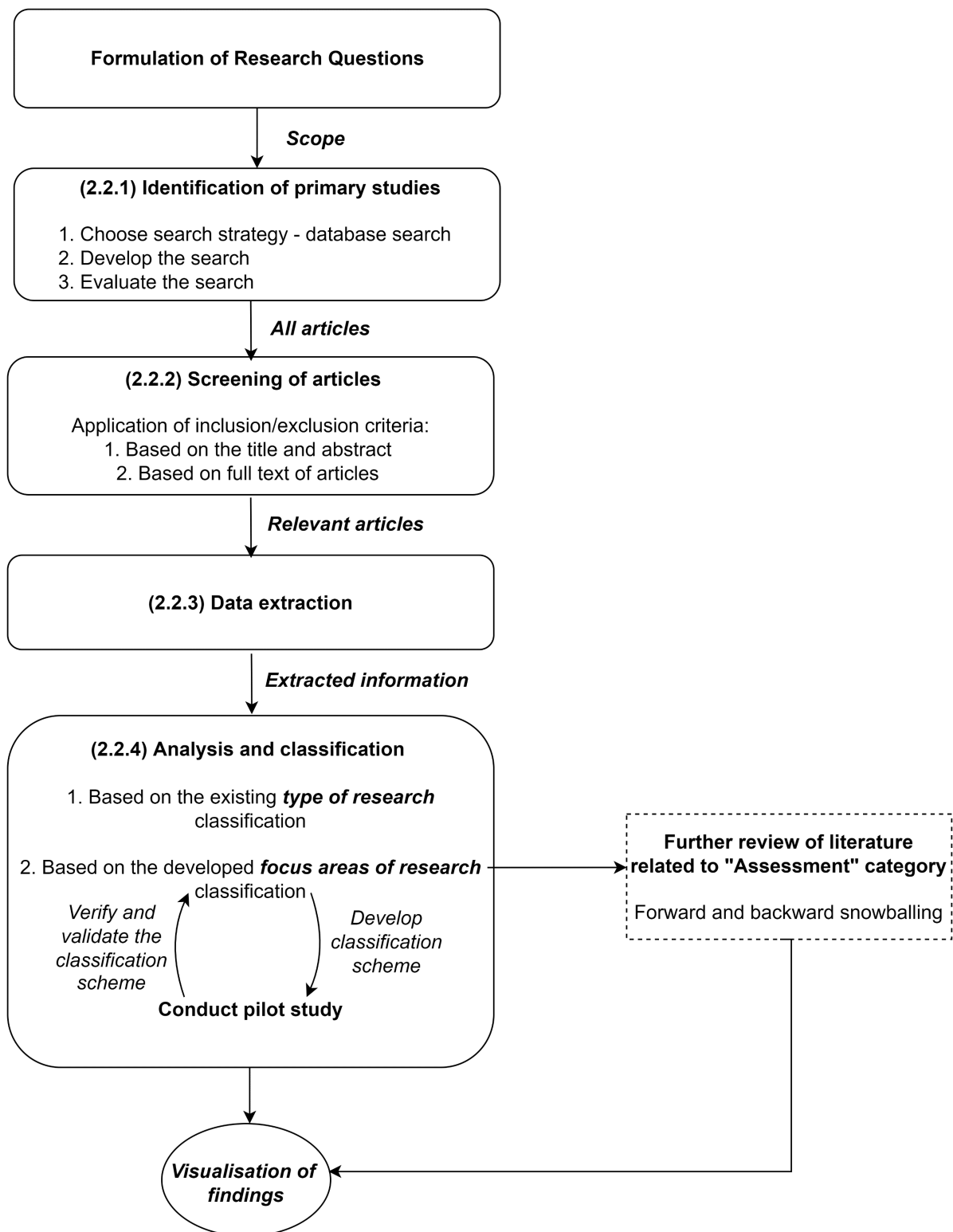


Figure 2.1 Systematic mapping study process adapted based on the guidelines of (Petersen et al., 2015). The numbers in braces represent the sections of this chapter that detail each phase's protocol.

Develop and evaluate the search

As suggested in the guidelines of Petersen et al. (2015), the research questions mentioned in section 2.1 and PICO (Population, Intervention, Comparison, Outcomes) were used to develop the search strings. Based on their recommendation, for mapping studies, only P and I from PICO were considered in building the search string. The search strings were composed of three main terms. The first term was intended to be broad and consisted of keywords related to the database modelling education areas of interest: teaching, learning and assessment. The second term was used to constrain the focused areas of the search to database modelling. This second term was intended to, by inference, exclude articles focused on database implementation and querying. The third term consisted of the keywords related to the focused population of the research: novices. Wild cards, when and where allowed by the digital library search functions, were used to capture the different variations of the terms used in the search strings. Further, alternative synonyms were selected for the search terms, and these were linked using the logical operators AND/OR, where AND was used to link the search terms and OR to link the synonyms.

During the initial stage, three different search strings were developed to retrieve primary studies from the digital databases. And then, the search strings were used on each digital database. Finally, the results of each search string were compared, as sometimes the searches resulted in a large number of irrelevant hits. The search strings developed along with the results from the ACM digital library are given below as an example to provide an idea of how the search string was finalised.

Search string 1: articles retrieved - 28

(teach* OR learn* OR asses*) AND (“database modelling” OR “database design” OR “entity-relationship modelling”) AND novice

Search string 2: articles retrieved - 311

(teach* OR learn* OR asses*) AND (“database modelling” OR “database design” OR “entity-relationship modelling”) AND (novice OR beginner OR introductory)

Search string 3: articles retrieved - 5082

(novice OR beginner OR introductory) AND (“database modelling” OR “database design” OR “entity-relationship modelling” OR “data modelling”) AND (teach* OR learn* OR asses*)

Based on the results retrieved from all the databases, search string 2 provided more consistent results compared to both search strings 1 and 3. Hence, search string 2 was finalised as the base search string

for this literature search. This was further evaluated by looking at the first 100 hundred results (where applicable) of each string to decide whether the string managed to capture the relevant studies. As an example, search string 3 captured many articles (~40%) in other modelling domains, such as domain modelling in UML, business process modelling, etc., and it was the same across the majority of the databases. Hence, it was excluded as an unsuitable search string for this study.

The base search string was then adjusted and refined accordingly as constrained by each database's available search options (Brereton et al., 2007). For example, in the ACM digital library, since the library did not allow wild cards with word phrases, the search string had to be extended using boolean operators (AND/OR) and full words to include all the required variants. In some cases, the search string was adapted based on the number of Boolean operators permitted in the string— the ScienceDirect digital library only allows a total of eight Boolean operators in a search, while IEEE Xplore allows ten Boolean operators. The search strings finalised for each digital library can be found in Table 2.1.

Table 2.1 Search queries used in digital libraries.

Database	Search
ACM	("data modelling" OR "data modeling" OR "database modelling" OR "database modeling" OR "database design" OR "database course" OR "er model" OR "entity-relationship model") AND (novice OR beginner OR student OR undergraduate OR introductory) AND (teach* OR learn* OR assess *)
SpringerLink	'(asses* OR teach* learn*) AND ("database design" OR "database model*" OR "er model*" OR "database course" OR "entity relationship model*") AND (novice OR beginner OR introductory)
IEEEExplore	(teach* OR learn* OR assess*) AND ("data modeling" OR "data modelling" OR "database modeling" OR "database modelling" OR "database design" OR "database course" OR "er model*") AND (novice OR beginner OR student OR introductory
ScienceDirect	(assessing OR teaching OR learning) AND ("database design" OR "er modelling" OR "database modelling") AND novice
Taylor & Francis	(asses* OR teach* OR learn*) AND ("database design" OR "database model" OR "database modelling" OR "database modeling" OR "entity relationship modelling" OR "entity relationship modeling") AND novice
AIS elibrary	(teach* OR learn* asses*) AND ("database design" OR "database model*" OR "er model*" OR "database course" OR "entity relationship model*") AND (novice OR beginner OR introductory)

2.2.2 Screening of papers

Once the search strings were finalised, they were used to identify the potential articles across the six digital databases. The identified articles were then subjected to an initial screening process based on the criteria developed for this SMS (See Table 2.2). The first inclusion criterion (I1) was developed to identify the articles within the intended scope of this review, which was teaching, learning, and assessing database modelling among introductory-level novices. The study range was thirty years, from 1990 to 2020, as defined by I2. Considering that there was no broader systematic review conducted in the area that could be replicated, it was decided to investigate the research as far as possible and 30 years was concluded as an achievable timeframe. The publication sources included only peer-reviewed journal and conference papers (I3). Moreover, grey literature, such as technical reports and blogs, is not common and less likely to be relevant to this thesis research area. The final inclusion criterion (I4) was defined to include only articles that were written in English. The exclusion criteria mentioned in Table 2.2 were used to scope the identified articles into database modelling. Hence, most of the exclusion criteria were designed to eliminate articles that were not relevant, for example, studies in the educational context focused on database implementation, querying, and security.

Table 2.2 Inclusion (I) and Exclusion (E) criteria.

ID	Criterion
I1	Studies specifically based on the teaching, assessment, and learning of conceptual & logical (e.g., relational) database modelling and design modules in tertiary education.
I2	Studies published between 1990 – 2020.
I3	Peer-reviewed studies published in journals and conference proceedings.
I4	Studies written in English.
E1	Studies describing the physical implementation of a database.
E2	Studies that mention databases that don't have a core focus on teaching and learning of databases (e.g., DB security, data warehouse, SQL, data Mining).
E3	Studies based on other modelling domains (e.g., Object-Oriented Analysis & Design, enterprise modelling, business process modelling).
E4	Studies where full text is not provided (where the publication is an abstract /extended abstract/poster) or not accessible.
E5	Studies that don't belong to the educational context.
E6	Duplicate studies (same study retrieved from different sources).
E7	Studies related to K-12 education.

Figure 2.2 illustrates the paper screening process in action and details the number of articles filtered by each of the exclusion criteria.

When applying the study selection criteria, the articles were first sorted by date of publication, from oldest to newest, as sometimes the selection was conducted in more than one session when large search

results were involved. Once the selection criteria were applied for each search result, the data (i.e., data source, date and time of selection, publication title, authors, year, venue, DOI or URL, include or exclude status, exclusion reason) was recorded for each article.

In this screening stage, the titles and abstracts of the articles were read and reviewed using the criteria given in Table 2.2. As a result, candidate papers were identified from the search across the six databases. In cases where the articles could not be definitively screened using the title or abstract, a full-text read was conducted against the same selection criteria. During full-text reading, it became apparent that further articles should be removed as they did not adhere to the inclusion and exclusion criteria. The remaining articles were included and went on to be analysed further, employing the systematic mapping process.

The literature search was subsequently repeated during the writing of this thesis to extend the study on 6th December 2022 to include articles from 2021 and 2022 and a further 21 articles were identified. A discussion of the findings from the extended study is included as an addendum at the end of this chapter.

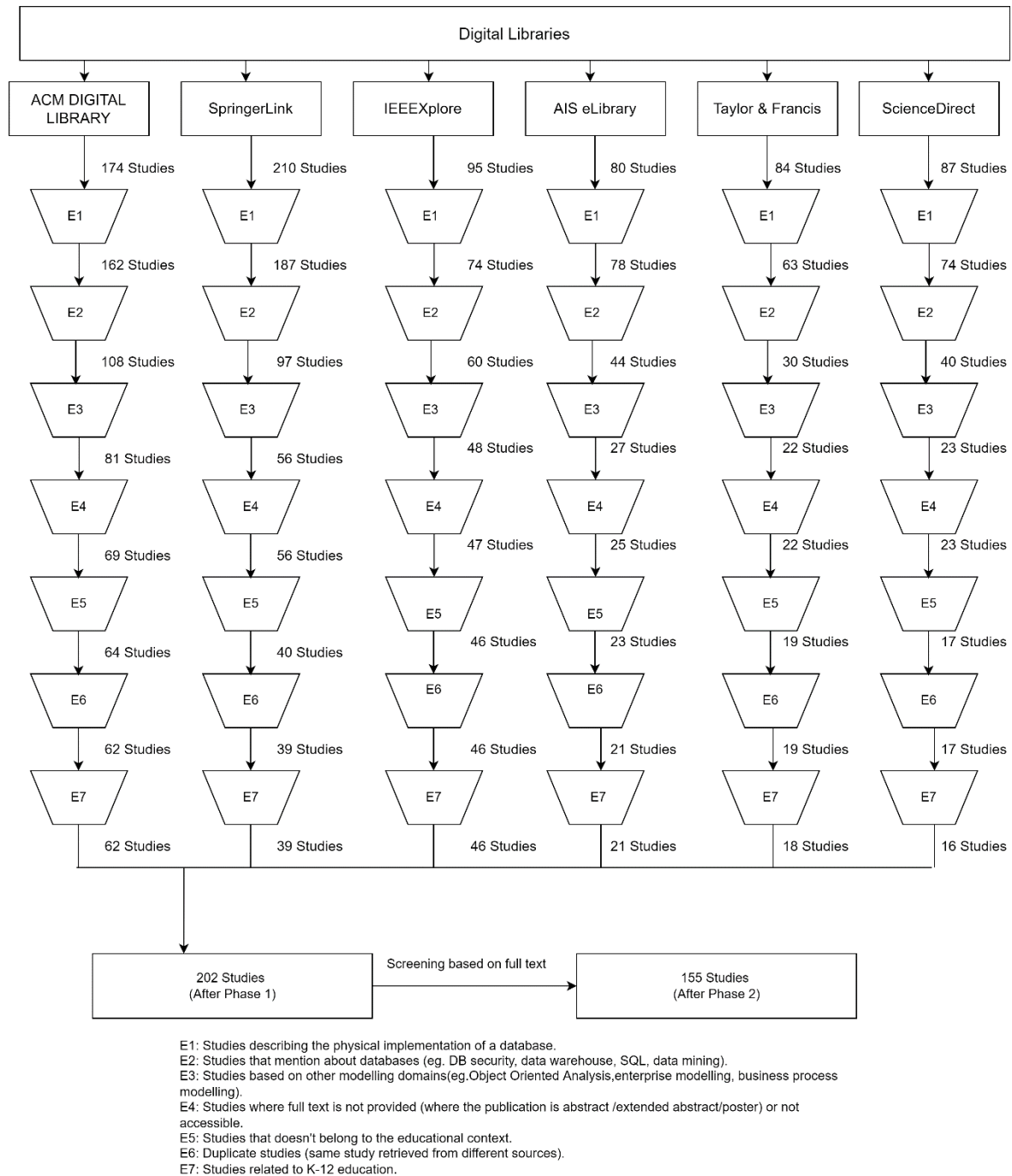


Figure 2.2 Paper screening process.

2.2.3 Data extraction

To extract data from the full set of selected articles, including those used to build the assessment focus area classification system, were read in full and classified accordingly. The template provided in Table 2.3 was used to extract and classify the selected articles.

Table 2.3 Data extraction form

	Data item	Research question
General information	Paper ID	
	Paper title	
	Author/s	
	Year of publication	
	Venue	
	Paper type (i.e., journal, conference, symposium, workshop)	
	Abstract	
	Keyword	
Classification information	Type of Research (i.e., empirical, replication, secondary, experience/opinion studies)	RQ1.1
	Focused area of research (i.e., course design, assessment, pedagogy, tools and resources, curricula, challenges)	RQ1.2
	Low-level classification of assessment category - practice and approaches proposed to assess the novice modellers (i.e., skill development and acquisition, assessment type, auto-grading of design/model, task/model assessment, feedback, academic integrity)	RQ1.3
	Reason for classification	

2.2.4 Analysis and classification

The extracted articles were then grouped under a category according to a classification scheme during the analysis. Petersen et al. (2015) distinguished two types of classification schemes that can be applied in systematic reviews and mappings. The first type is general classification schemes that can be applied regardless of the domain of the review, like venue, research type and research methods. The second type is more topic-dependent, where the classification schemes emerge from the data or are based on the existing literature. Existing classification schemes were explored to categorise the type of research presented in the selected articles. Wieringa et al. (2006) proposed six categories to classify the research type for requirements engineering papers, a common research type facet used in SMSs. However, these categories were not a natural fit for the classification of educational research publications. After further exploration, the categories related to the type of research dimension were adapted from the research type categories introduced by Taipalus and Seppänen (2020) for educational research, as provided in Table 2.4. However, the authors did not explain why they chose to adopt the research type framework they did. But, it seemed reasonable to adopt this classification in this study, considering the lack of systematic reviews in the area. It was important to understand the quality and rigour of the results of this study. That was the main reason why the research type categories were identified to form an understanding of how reliable the results presented in the articles were.

Table 2.4 Type of research classification scheme (Adopted from (Taipalus & Seppänen, 2020)).

	Category	Definition
1	Empirical research	Papers using empirical evidence to validate the knowledge claim stated in the paper can be done qualitatively, quantitatively, or using a mixed approach. e.g., Hypotheses can be tested to validate the stated research questions using several statistical measures.
2	Secondary study	Papers present a study conducted based on the existing literature in the form of a systematic literature review or systematic mapping study.
3	Opinion/Experience study	Papers present the author's or a third party's opinion on something (e.g., whether something should be taught, how it should be taught, or to whom it should be taught) or their experience on how something was done (e.g., implementation of new course materials, positive and negative points of the implementation). Typically, no scientific evidence is presented in these papers.

A pilot study was conducted to build the next classification scheme, as no existing schema looked at the focus areas in database education at the time this was conducted. So, this classification scheme had to be developed in a data-driven manner and based on the information from the existing literature. The classification scheme was developed by following a similar open coding approach discussed by Petersen et al. (2015), going through the full text of the articles to identify the categories that described the focus area of the research. To initially develop this classification scheme, a pilot study was conducted where a total of 30 studies (five studies from each database) were randomly selected across the six digital databases. The researcher and both supervisors were involved in developing the conceptual framework through discussion until a consensus was reached. The scheme was fine-tuned throughout the pilot study. As a new article came that could not be categorised into an existing category, a new category was added to extend the classification scheme. However, if it seemed to fit into an existing category, but the definition didn't capture it, then the definition was broadened as required. For example, a category named "Design skills" emerged during the pilot study, but it did not occur more than once, and hence, it was removed as a category. Additionally, "Tools" under the assessments category was removed as all the articles related to tools could be more precisely categorised as related to tools to automate marking and were categorised as "Auto-grading of model". "Academic integrity" was introduced as a new category to include articles discussing academic integrity and plagiarism detection of student-generated database models.

The detailed definitions of the categories under the focus areas of research are provided in Table 2.5.

Table 2.5 Focus areas of research in database modelling education.

	Category	Definition
1	Tools and Resources	Include interactive computer systems to support teaching and learning, evaluation of existing tools, data sets, and fully developed assessments e.g., student worksheets
2	Pedagogy	Effective teaching and learning practices including how the teaching occurs, the approach to teaching and learning, the way content is delivered, and what students learn as a result of the process. e.g., Teaching & learning approaches (application of an existing approach, introducing a new approach)
3	Curricula design	The process of adding/updating a database systems module/course into an integrated group of courses/modules aligned to a program. e.g., ACM curricula design of a program
4	Course design	Overall course structure in creating a new course, redesigning an existing course or evaluating an existing course. Design specification covers most aspects of the design of the single course or module and may include details of learning outcomes, content, pedagogy, assessments, etc.
5	Assessment	Evaluating and measuring the learning progress, skill acquisition, and learning needs of students.
6	Teaching/learning challenges	Challenges faced in teaching and learning novice database courses e.g., Issues, misconceptions, common errors faced by students in learning or teaching challenges in transferring tacit knowledge like a high number of interrelated concepts with abstract nature

Based on the developed classification scheme, each article was categorised under one single category aligned with the primary research contribution, as determined by the researcher. When each article was categorised, a rationale for the decision was recorded in the data extraction form. The supervisors reviewed the data extraction to validate the suitability of the classification. Once the data extraction was completed, the data were synthesised to answer the research questions.

As discussed in section 2.1, it was decided to explore the literature related to the “Assessment” category more in-depth. However, due to the small number of articles (18 articles) backward and forward snowballing (Wohlin, 2014) was performed. For each article, when it comes to forward snowballing, first, the title and abstract of the cited article were checked along with the place of the citation to make a decision. The same inclusion and exclusion criteria (see Table 2.2) were applied to exclude the articles, and further, some articles were removed since they were already included in the review. In backward snowballing, titles and the place of reference to the identified articles from the reference list were checked first. If there is a possibility to include the article, then the abstract of the referenced article was checked to make a decision. The full text of the articles was referred to when a decision could not be made from the title and abstract reading. See Table 2.6 for the articles identified via snowballing.

Table 2.6 Articles identified through snowballing.

Backward snowballing				Forward snowballing		
Article	Number of referenced articles	Number of included articles	Included article	Number of cited articles	Number of included articles	Included article
(Batra & Davis, 1992)	39	0	NA	71	2	(Batra & Sein, 1994) (Yang, 2003)
(Bock & Ryan, 1996)	15	0	NA	1	0	NA
(Ryan et al., 2000)	65	0	NA	17	2	(Moores et al., 2006) (Dhillon & Dasgupta, 2011)
(Batra & Wishart, 2004)	52	0	NA	21	0	NA
(Fuller et al., 2010)	9	0	NA	13	0	NA
(Pittges et al., 2009)	9	0	NA	1	0	NA
(Paterson et al., 2012)	15	0	NA	1	0	NA
(Etlinger, 2013)	9	0	NA	1	0	NA
(Granda et al., 2015)	37	0	NA	5	0	NA
(Kawash & Sailunaz, 2020)	12	0	NA	1	0	NA
(Kawash et al., 2020)	15	0	NA	3	0	NA
(Prados et al., 2006)	9	1	(Prados et al., 2005)	5	0	NA
(Thomas et al., 2007)	15	3	(Thomas et al., 2006) (Higgins & Bligh, 2006) (Waugh et al., 2004)	14	3	(Simanjuntak, 2015) (Thomas, 2012) (Thomas et al., 2015)
(Kesh, 1995)	19	0	NA	54	0	NA
(Sinha & Vessey, 1999)	16	0	NA	36	0	NA
(Lee & Chan, 2001)	16	0	NA	2	0	NA
(Becker et al., 2008)	130	0	NA	AIS eLibrary did not provide citation details		
(Kouznetsova, 2009)	14	0	NA	1	0	NA
			4 articles			
				7 articles		
				Total number of articles from snowballing = 11 articles		

All articles categorised under this theme were then further analysed to extract common themes to understand the assessment spectrum better to answer RQ1.3. The entire process that was used to develop the *focus area* categories was repeated to classify how novice database modellers are assessed. Table 2.7 provides the scheme developed to code the assessment of novice modellers.

Table 2.7 Assessment of novice modellers.

	Category	Definition
1	Skill development and acquisition	Papers that discuss different skills students need to acquire in building database designs/models, the factors which affect the skills of novices/models they design, or the papers that discuss the performance of modellers. e.g., comprehension skills, database creation skills
2	Assessment type	Papers that discuss the design and evaluation of different types of assessments given to students. e.g., Formative assessments, summative assessments
3	Auto-grading of model	Papers that discuss the automatic instruments developed to assess the design/models.
4	Task/model assessment	Papers that used learning theories/ taxonomies/ heuristics/guidelines in assessing tasks or design/model quality.
5	Feedback	Papers that discuss the feedback provided as a part of summative and formative assessments. e.g., Automated or manual feedback
6	Academic integrity	Papers that discuss the academic integrity and plagiarism of assessments.

2.3 Findings

2.3.1 General findings

Seven hundred and thirty articles were obtained using the finalised search strings across the six databases between 1990 and 2020 inclusive. These articles were then subjected to a screening process as described in section 2.2.2. The initial screening conducted on the article's abstract and title by applying the inclusion and exclusion criteria resulted in 202 articles. The articles that were not definitively screened using abstracts and titles were then decided based on a full-text reading, resulting in 47 more articles being removed. Altogether, 155 articles were selected for further analysis.

This section provides an overview of the selected studies. The number of articles identified across thirty years highlights the need for more research in the domain. Figure 2.3 provides a comparison between the number of articles published in the past thirty-two years in database modelling and computing education overall. Based on the number of CSER articles published per year, the proportion of articles published in database modelling education has increased, but not at the pace of articles in other areas of CSER. Two recent studies conducted at the time of this thesis research also highlight the lack of

research conducted in the domain (Ishaq et al., 2022; Simon & Sheard, 2020). Simon and Sheard's (2020) analysis of 24 years of the ITiCSE conference, arguably one of the highly ranked conferences in CSER, found that 38% of papers represent research on teaching and learning computer programming, while only 2% are database-related publications. The comparison presented in Figure 2.3 further confirms the findings of Simon and Sheard (2020) regarding the low number of publications in the area of database education.

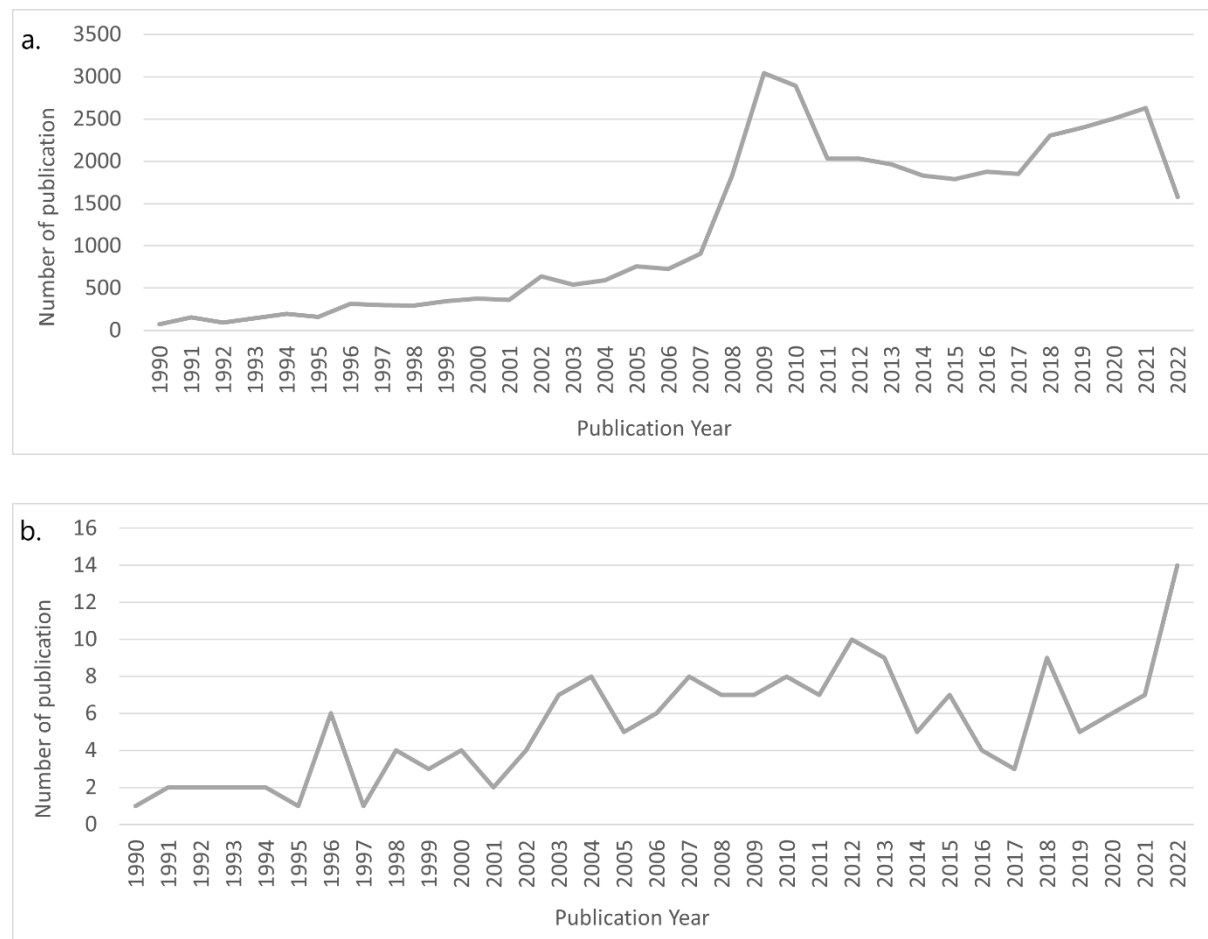


Figure 2.3 Yearly distribution of CSER vs DB modelling education research. a. provides the total number of publications identified in CSER – the articles were identified using the search string ("Computer Science Education" OR CER OR CSEd) across the same six databases. b. provides the total number of publications related to database education. Note the difference in the Y-axis scale in a. and b.

More than half of the studies identified in the SMS conducted for this thesis (55%) were published in conference proceedings, while 37% were journal articles, and 8% were from symposium and workshop reports. Figure 2.4 indicates the quinquennial distribution of the publications based on the venue type.

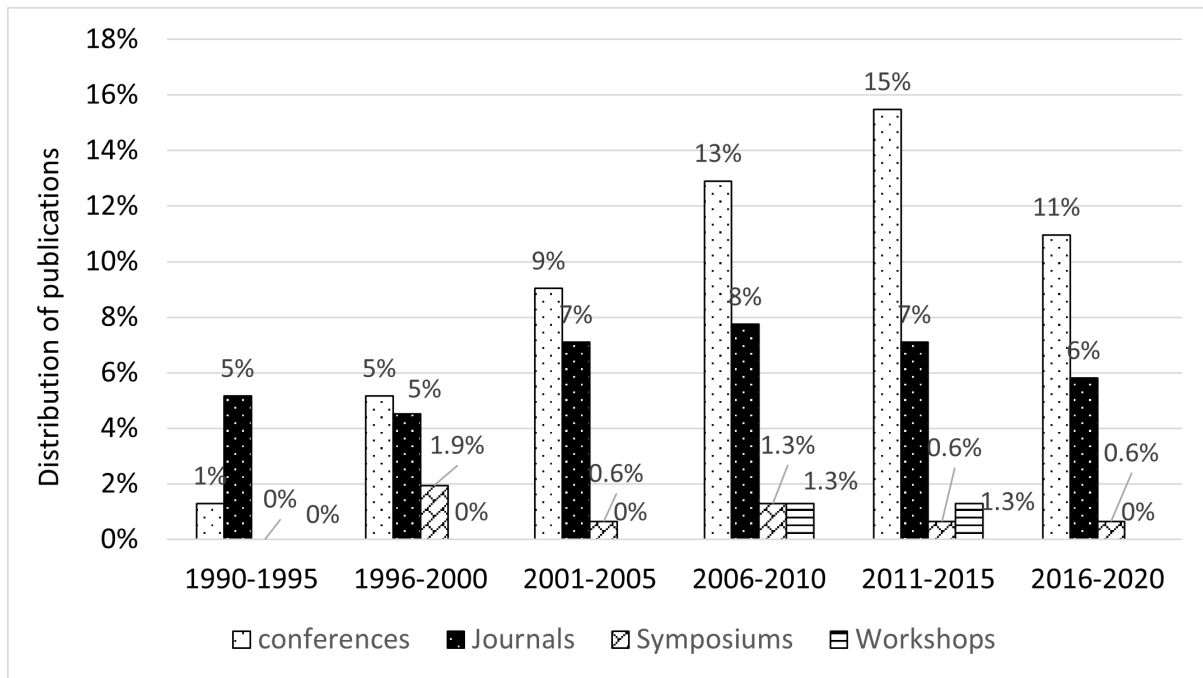


Figure 2.4 Quinquennial distribution of the publications based on the venue type.

2.3.2 RQ1.1 – Types of research

Research question 1 was utilised to understand the diverse nature and the broader landscape of the literature on database modelling education by classifying the articles based on research type. While some of the published articles were empirical, most of the claims of the efficacy of an intervention or a tool were based on conjectures formed from opinion, perceptions, or anecdotal experience. Around one-third of the publications were classified as reporting on an empirical study where empirical evidence was used to validate the findings and claims stated in the paper. A limited number of articles, only 6%, fell into the secondary studies research type.

Based on the frequency of the types of research over the period (See Figure 2.5), predictably, opinion and experience articles were the most favoured type of article. Although there was a consistent increase in the number of empirical research studies until 2015, a significant decline in empirical research can be observed from 2016 to 2020.

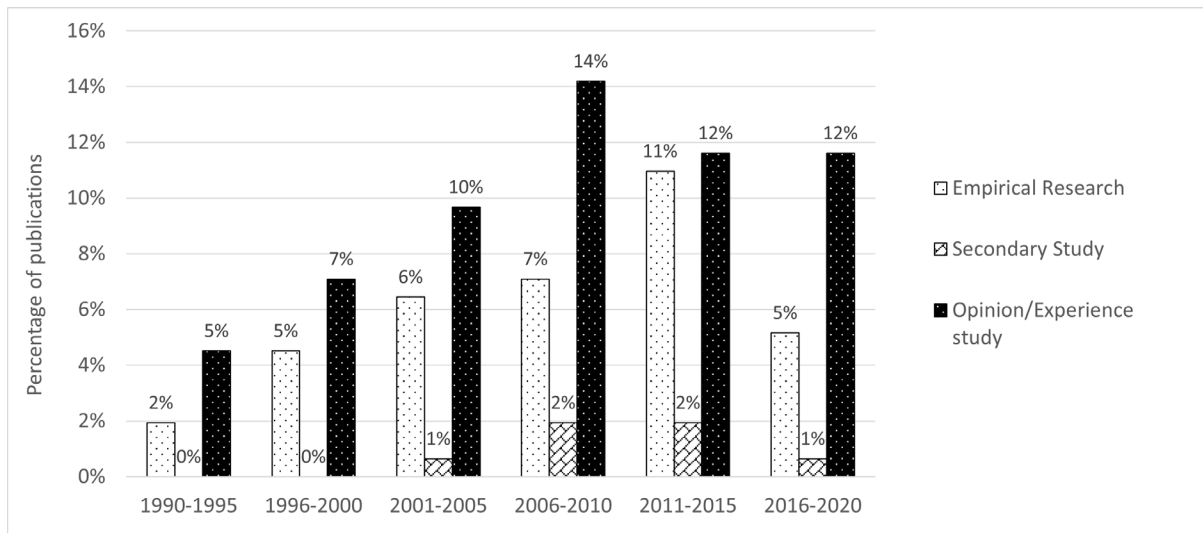


Figure 2.5 The quinquennial publications of type of research.

2.3.3 RQ1.2 – Focus areas of research

Research question 2 was used to explore the different research areas reported in the literature concerning introductory database design and modelling to get a broad overview of current research. Figure 2.6 provides the percentage of publications under each category of focus area. As per Figure 2.6, the most prevalent areas of research were related to the development of supportive tools for teaching and learning, followed by publications related to pedagogy.

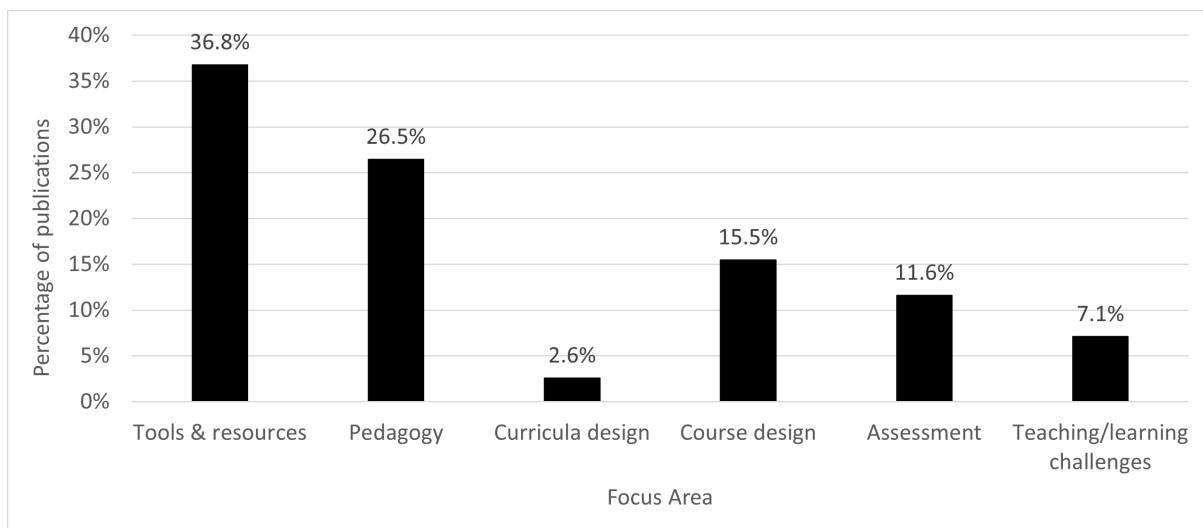


Figure 2.6 Focus areas of research based on the percentage of publications.

Tools and resources

The tools and resources category includes work related to interactive computer systems or teaching resources, such as fully developed student worksheets built to support the teaching and learning of novice database modelling. Since 2001, there has been an increase in the number of articles in the literature that focus on tools and resources (See Figure 2.7). However, almost all of the tools are not open-source or freely available, which might explain why the tools are not empirically evaluated or used in replication studies. Many of the tools appeared to have been used in a research study and then lost to obscurity. The tools identified in this study mainly handle one of three main stages of database design: either conceptual design (Nattassha & Azizah, 2016; Weerasinghe & Mitrovic, 2002; Zhou et al., 2004), logical design or relational schema (Protogeros & Tzafilkou, 2015) or normalisation (Kaur & Chopra, 2010; Kung & Kung, 2013). However, a limited number of tools were designed to facilitate the transition from one stage to another: from a textual description to a conceptual model (Omar & Baryannis, 2020; Ridjanovic, 2003) or from a conceptual model to a logical model (Lim & Hunter, 1992).

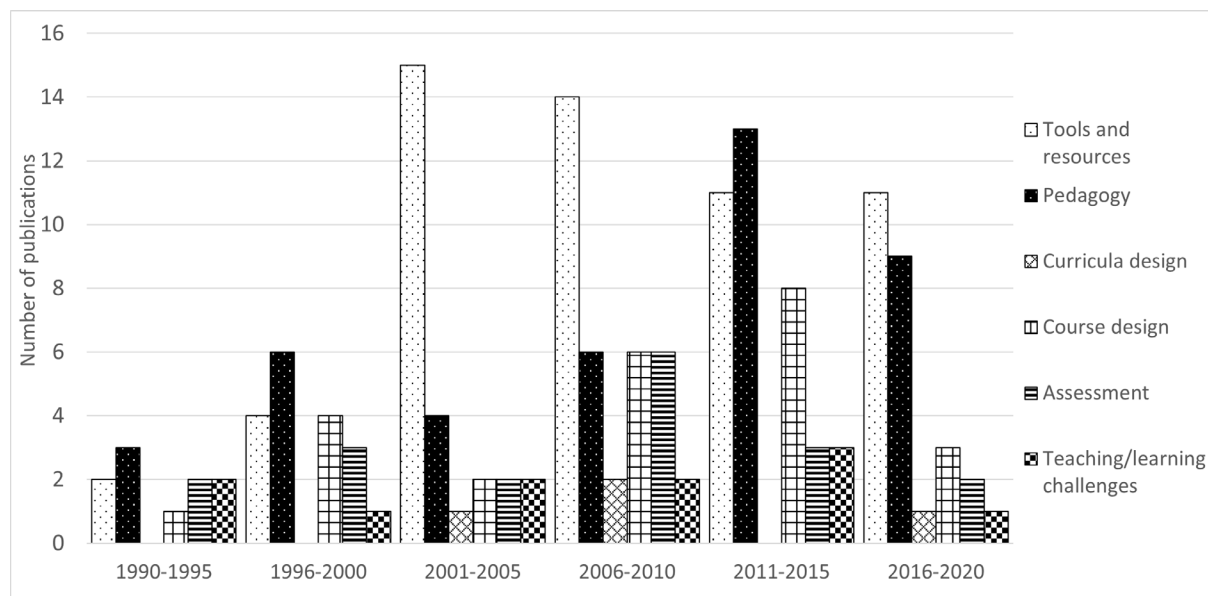


Figure 2.7 The quinquennial publications by focused areas of research.

From a very early stage in the literature, several researchers have worked on tools related to conceptual design or ER models. A virtual learning environment for developing an ER model is one such early initiative. The tool required the learner to produce textual commands in order to generate an ER model dynamically (Hall & Gordon, 1998). This tool is an example of early research looking at feedback mechanisms in a database design tool in which the user could only enter correct commands, and provided feedback was provided on incorrect command choices. Over time, educators and researchers

have published articles that introduce more interactive tools for building conceptual designs by introducing more user-friendly interfaces (Kim et al., 2007; Milik et al., 2006b; Zhou et al., 2004). Protogeros and Tzafilkou (2015) used a *simple-talking* approach to build a tool to abstract database terminology and logic, arguing that it would help untrained users progressively design their database. The core idea behind the approach is to allow users to select a verb that describes the relationship between two entities. This chosen verb is then utilised to generate relevant questions about various database constructs, such as attributes, relationship types etc., to generate a relational schema. Based on their experiment study between experienced and non-experience users, the authors have concluded that the approach appeared to be a good tool to train designers, particularly novices, in building their database schema.

While several tools have been developed to facilitate database modelling, fewer tools have been developed to facilitate the normalisation process. Of those tools, all attempt to support the learning of normalisation through visualisation (Kaur & Chopra, 2010; Kung & Kung, 2013; Suraweera & Mitrovic, 2004). For example, Kung and Kung (2013) developed a Java applet tool in which students enter the functional dependencies that they have identified. Then, the tool allows them to visually observe how the entered dependencies are normalised in a step-by-step manner.

A group of New Zealand researchers has published several articles on the development of Intelligent Tutoring Systems (ITS) for teaching and learning database design concepts (Milik et al., 2006b, 2008; Antonija Mitrovic et al., 2004; Antonija Mitrovic & Suraweera, 2016; Weerasinghe et al., 2011; Weerasinghe & Mitrovic, 2002, 2006; Zakharov et al., 2008). While the majority of the tools identified in the literature were not publicly available and were not empirically evaluated, these ITSs have been used in their courses at the University of Canterbury since the early 2000s. Furthermore, they deployed two database modelling-related tools (KERMIT and NORMIT) to a web portal aimed at university-level students taking undergraduate database courses in 2003 (Antonija Mitrovic, 2006). The researchers initially developed KERMIT, a knowledge-based ER modelling tutor that supports students learning conceptual database modelling (Suraweera & Mitrovic, 2002). The tool presents requirements for the students to develop basic ER diagrams. KERMIT later evolved to support Enhanced ER diagrams – EER Tutor (Mitrovic & Suraweera, 2016). EER tutor also supports the students in learning to design database models from textual descriptions, however unlike KERMIT, EER tutor was web-enabled and covered more advanced constructs of an ER diagram. ERM tutor was the natural progression from the earlier tools to teach the mapping of conceptual to logical database models (Milik et al., 2006b). In this problem-solving environment, the problems are presented to students in the form of Entity-Relationship (ER) diagrams. The purpose of this environment is to guide students in mapping these ER diagrams into relational schemas. This mapping process is carried out step-by-step, providing students with a structured approach to convert the conceptual ER diagrams into concrete relational

database schemas. Subsequently, they developed a tool to support the normalisation – NORMIT (Mitrovic, 2002).

While most of the articles categorised under this theme were related to interactive computer tools, approximately 15% of the articles presented different resources that may be used to teach database modelling. An early resource developed by Storey et al. (1998) was an ontology developed to be used as a surrogate for the meaning of words. This database design ontology is a reasoning mechanism to help students understand what different terms mean and how they relate semantically. Their objective was to develop an ontology (a set of constructs used to represent real-world domains) that can be later incorporated into a database design tool as a separate component to automatically compare terms based on understanding their semantics. They went on to extend this study and evaluated the ontology in prototypes and tools (Sugumaran & Storey, 2006, 2003). In their latest study, the tool developed allowed the users to create and manage new ontologies by giving an ontology name, domain name, terms, basic relationships, and domain relationships that can be utilised in ER model generation.

Many researchers have introduced learning aids or courseware to support the teaching and learning of database modelling. Several of these learning aids utilised visualisation to support novices. Examples include using visualisation to help map the relationships between entities or to understand the basic concepts in database modelling better (Boyd et al., 2005; Goelman & Dietrich, 2018; Murray & Guimaraes, 2008a, 2008b; Pahl et al., 2004)

An exemplar structured data modelling worksheet is another resource that has been made available to educators (Bock & Yager, 2005). The worksheet is arranged in two columns where the left side is used to record the design constructs like entities and attributes while the right hand side column is used to draw simple depictions of each relationship as they are identified. The authors conducted a between-groups experiment study where one group used the structured worksheet, while the other group did not use the worksheet to work on the same database design task. The authors found that the students who used the worksheet demonstrated a significant improvement in learning with regard to their ability to identify entities, attributes and particularly ternary relationships over the group that did not have the worksheet. The authors claim that their structured worksheet approach reduces the cognitive distance between a modelling task and a modelling solution, which is an interesting finding that can be particularly useful with novices. This is an interesting way for the novices to analyse a case study description, and due to that, this could inform the design of the think-aloud instrument of this study, considering the additional cognitive load that needs to be handled during the think-aloud data collection.

Further, a few researchers have developed data modelling patterns – a standardised representation of a data structure that exhibits similarities with commonly encountered solutions to data requirements

problems and provided heuristics that can be followed in applying the patterns (e.g., a transaction pattern that exchanges resources between two entities can be applied in domains like selling of goods, services, borrowing etc.) during a database design task (Antony & Mellarkod, 2004; Thonggoom et al., 2011). These patterns can be quite useful with novices to avoid the lack of experience and incomplete knowledge of the problem domain.

Pedagogy

The articles identified under this theme discuss effective teaching practices and approaches developed to overcome the challenges students and instructors face. Various teaching and learning innovations fall into this category, varying from delivery modes to mechanisms developed to reduce the cognitive strain of novice designers. Approximately 31% of the articles identified were categorised under this theme. Further, the studies related to pedagogy appeared to surge during 2021 and 2022, which is presented in detail in the addendum provided at the end of this chapter.

One of the most prominent pedagogical innovations in database modelling has been the application of project-based learning (PBL) to enhance novice database modelling skills. Throughout the literature, researchers have discussed how database projects should be integrated into a course. One such model is bringing real-world projects to the classroom through university-industry partnerships (Dietrich & Urban, 1996; Saiedian & Farhat, 1991; Scorce, 2010; Tashakkori & Andrews, 2018). These real-world projects were ones where the students had to be closely involved in the rigorous design processes required to create applicable models. Using case studies is another widely suggested approach to learning database modelling, situating the case study with little or no industry partner engagement (Boyeena & Goteti, 2010; Rae, 1990; Robbert, 2000; Shebaro, 2018; Tuttle, 2002). Robbert's (2000) work has introduced a multi-step project consisting of multiple components to stimulate real-world experience that evolved over time. In their approach, the students are initially required to develop a simple database where after a while, they are informed that the companies have merged with other similar companies (a.k.a. other students) and how they handle these scenarios have been observed during the study period providing an opportunity for the students to experience real-world problems. Other researchers have looked at students' learning in team-based environments (Marjanovic, 2013, 2016; Mitchell, 2020). Brown and Klein (2003) introduced two cooperative learning techniques – paired thinking and cooperative groups which appeared to produce positive learning attitudes among their students, though the impact of these approaches was not empirically evaluated.

Different delivery modes, like flipped classrooms (Kui et al., 2018; Lina, 2018; Martinez, 2012) and blended classrooms (Wang & Ma, 2017), have been discussed in the prior literature. The researchers working on such initiatives argue that these approaches can be more student-centric and, as a result,

may help overcome the difficulties novice database designers face, such as a lack of engagement while increasing the student's problem-solving abilities.

Work conducted by Lin et al. (2013) looked at the gap between doing and knowing in novices and explored the opportunities to reduce this gap using different types of prompts (e.g., self-explanation, knowledge integration prompts, procedural prompts, etc.) in peer-tutoring scaffolding. A follow-up study reported that a group that used elaboration-based self-explanation scored lower than the other treatment group that adopted scenario-based self-explanation (M.-H. Lin et al., 2015). Further, the students who used integral question prompts outperformed the ones who received gradual question prompts in performing the database problem-solving tasks. The prompts they have used here, in fact, inspired the researcher to introduce certain prompts while developing the in-person think-aloud data collection protocol discussed in section 7.4.

One main issue discussed in teaching and learning is the abstract and complex domain of database analysis and design. Several researchers have looked at ways to overcome this issue by adhering to techniques built around constructivist epistemology. Connolly and Begg (2006) argue that *learning design by doing design* will help the students achieve knowledge and higher-order skills to perform a task as effectively as an experienced designer. Several other educators shared the same notion that the constructivist approach provides a solid foundation for mastering the art of database design (Chen et al., 2012; Pamela Neely, 2007).

Mitrovic et al. (2012) also explored two teaching strategies using the NORMIT tool, where one strategy enforced the procedural and scaffolded nature of data normalisation, and the second strategy provided the students with more freedom to complete the task. Unsurprisingly, the scaffolded strategy was reported to result in the acquisition of better problem-solving skills, resulting in the creation of better models than the freer, less-scaffolded strategy, which could be considered in developing the research instrument as a mechanism to reduce the cognitive load related to database modelling tasks.

Curricula design

The articles categorised under this theme discuss the process of adding or updating a database systems module in line with an integrated group of courses in a program. Database design courses have been revisited multiple times for various reasons like adhering to guidelines, accreditations, or evaluations. The ACM curriculum guidelines are such initiatives where recommendations are provided about the modules in degree programs. The 2010 ACM Curriculum Guidelines for Undergraduate Degree Programs in IS recommends a Data and Information Management foundation course -IS 2010.2 (Topi et al., 2010). Similarly, Martínez-González and Duffing (2007) examined the effect of directing the curricula to be taught similarly throughout different universities across Europe. Their analysis suggested

that universities must redesign their degree programmes and teaching strategies in line with one set of guidelines. Further studies have discussed placing database systems courses in line with proceeding database courses in a program. As an example, Udoh (2006) describes how the curriculum was structured to provide two courses in sequence for CS and IS majors in an integrated project environment. In a similar attempt in a recent study, Mithun and Luo (2019) presented redesigning all database courses in their data management curriculum enhancement to incorporate cross-course knowledge integration. This is an interesting curriculum design where they have utilised a common project throughout all the database courses so that the students can integrate and apply theoretical knowledge gained in lower-level courses to the advanced ones in the same project.

Course design

The articles under this theme focus on the overall course structure in creating a new course, redesigning or evaluating an existing course. Course design specification covers most aspects of a course or a module, including details like learning outcomes, content, pedagogy, assessments, etc.

Drake (1998) discusses how they developed a database systems course integrated with the topics of software engineering, database management, human-computer interfaces, and computer ethics. They introduced each content area embedded within a database project. This approach allowed students to design and implement a database and an interface, considering aspects of interface design and ethics. This is an early example of attempting to teach introductory database topics differently, which may not be seen even now in course delivery. In 2000, Springsteel et al. (2000) discussed the direction database courses should take for the upcoming decade and what needs to be changed. However, they concluded it was hard to provide a “one-size-fits-all” recommendation and instead provided alternatives that could be adopted. Like Drake’s unified concept areas, their first recommendation was to offer two semi-independent courses with a web/database integrated course focusing on the web as a front end to a relational/object-oriented (OO) database. And a database management systems course with a relational and OO hybrid model with traditional topics. The second recommendation was to have a year long sequence of courses starting with a data management course that covered both relational and object database management systems (DBMS) followed by a course on data management that covers topics on data warehousing and data mining. From time to time, several researchers have discussed the integration of web integration into databases however, none of these studies has been empirically evaluated to study the impact of such an approach (Maiorana, 2015; Schaeffer & Olson, 2011).

One article described how a course teaching team evaluated an introductory database course in 2006 and then again in 2008. During this intervening period, they made changes mainly to the delivery of content from 2006 to address one main feedback received from the students –the struggle of connecting

the theoretical aspects with the implementation. Due to this, in 2008, they interleaved the SQL concepts and practices with database modelling instead of delivering the content in two parts: the first half on ER modelling/ normalisation and the second half on SQL and implementation, as they did in 2006, which in turn changed some of the assessment components. While there was some improvement in results in ER designing learning outcomes, the SQL-related outcomes did not change between the years. However, their assessment identified a significant decrease in performance in outcomes related to normalisation. The author argues, *"One issue is that there is a lot of material in the 11 week course. Changing the course to place more emphasis on one area may have a corresponding negative impact in other areas unless careful design is done."* (Holden, 2008, p. 137). Several researchers have discussed the amount of delivery material of a typical introductory course, agreeing with the argument and how the heavy material particularly brings forth several interrelated concepts that can be challenging for novices. Even though this is still an ongoing debate on what to teach or what to remove, many researchers have attempted to develop courses to reduce this issue. Ren and Nan (2012) developed a course around three stages, taking database design as a central trunk line. The course starts with analysing and organising business data, then creating a data model and developing, managing and maintaining the data model. Another article reports on a course design that introduced a sequence of topics in an introductory database course that was significantly different from the adopted approaches at the time (Kwan, 2013). The author explained that the course should start with an introduction to database management and relational database concepts. They argue that basic SQL should be taught next to get students engaged and learning, as SQL is comparatively easy to learn when compared to data modelling or normalisation. Further, they have suggested teaching database normalisation next, stating that it's easier to learn than data modelling. The notion that normalisation is easier than data modelling for novices is contrary to most views in the literature (Amin et al., 2019; Wang et al., 2020). Kwan's design is one of the first pieces of literature that has proposed an ordering of database topics in a course that deviates from the traditional model and then implements and, indeed, argues against such a sequence of topics.

Mason et al. (2016) employed Cognitive Load Theory (CLT) to redesign a database systems course. The main objective of the redesign was to address poor student performance. They applied the ideas from CLT to reduce the extraneous cognitive load by making changes to the instructional design, for example, resequencing the content, using worked examples, segmentation and distribution of core SQL concepts, etc. The authors reported that the new cohort's pass rate was improved by 34% and generated higher student satisfaction. While the authors highlighted the success of the course redesign using evaluations and pass rates, no controlled experiments were included.

Teaching/learning challenges

The articles identified under this theme highlight the challenges students and instructors face in teaching and learning novice database courses. Batra and Antony (1994) stated that underlying causes for the errors committed by novices may be related to the complexity of the task, misapplication of heuristics and/or incomplete knowledge of database design. They also highlighted the inability of the novices to consider alternative design solutions.

The work of Batra has considered four viewpoints (problem-solving principles, design principles, information overload and systems theory) to identify factors that cause complexity in data modelling (Dinesh Batra, 2005, 2007). Batra has mainly categorised these factors into two categories –those that are intrinsic to the problem and hard to control and those that can be addressed to minimise the complexity. Batra proposed a semantic differential scale to assess cognitive complexity. Although this classification is subjective and lacks empirical evaluation, certain factors are highlighted as having high severity, including significant interactions, mental aids, and the lack of shortcuts, which provide valuable insights into the factors that can be utilised in the development of research instruments for the think-aloud sessions. Moreover, he has mentioned that *“there is the need for a tool that provides mental aids, that is, guidance, prompts, and aids to move the designer along the modeling process”* (Batra, 2007, p.241), which further highlights the importance of promptly guiding the novices during the modelling process.

Several researchers have attempted ways to identify the errors made by novices, which is an important requirement before suggesting solutions like introducing tools (Currim et al., 2014; Hussain, 2016). Currim et al.’s work has significantly evaluated some main errors made by novices and the possible causes. They claimed that when a novice manages to differentiate between relevant and irrelevant information, they will likely not add extra relationships to the model. Further, they have stated that using attributes in place of relationships was the most frequent error made by the novices, possibly not being able to see the “big picture” between entities but instead seeing a property of one entity. They further stated: *“Another possible reason is the usage of CASE tools that represent relationships by modeling foreign key attributes within an entity class. Using attributes instead of a relationship results in a loss of distinction between entity properties (attributes) and associations (relationships)”* (Currim et al., 2014, p.22), which brings the question of whether using a CASE tool is the best route to analyse the novices. The work of Hussain (2016) has identified making assumptions as a root cause for many errors made by novices, which is an interesting factor that could be further looked into during the observations of novices in this thesis.

Assessment

The articles under this theme discuss the assessment of novices. This includes evaluating and measuring their performance, the quality of the produced designs, the required skills, and how the relevant skills are acquired. Only eighteen articles (11.5%) were found that focus on assessing novice database designers, and further, 11 articles through snowballing (see Table 2.6). This theme was explored further since the studies categorised under this theme may have several implications for the research conducted for this doctoral thesis.

As mentioned earlier, all the articles in the “Assessment” category were further analysed to extract common themes to better understand the spectrum assessments used to assess novice data modellers to answer RQ1.3. See Table 2.8 for the assessment themes along with the articles identified as falling into each category. Most articles were categorised under skill development and acquisition and auto-grading of models, while the rest were focused on either assessment types and task or model evaluation.

Table 2.8 Approaches and practices proposed to assess novice database modellers.

Category	Definition	Paper ID
1 Skill development and acquisition	Papers that discuss different skills students need to acquire in building database designs/models, the factors which affect the skills of novices/models they design, or the papers that discuss the performance of modellers. e.g., comprehension skills, database creation skills	(Batra & Davis, 1992) (Bock & Ryan, 1996) (Ryan et al., 2000) (Yang, 2003) (Batra & Wishart, 2004) (Moores et al., 2006) (Fuller et al., 2010) (Dhillon & Dasgupta, 2011)
2 Assessment type	Papers that discuss the design and evaluation of different types of assessments given to students. e.g., Formative assessments, summative assessments	(Pittges et al., 2009) (Paterson et al., 2012) (Etlinger, 2013) (Granda et al., 2015) (Kawash & Sailunaz, 2020) (Kawash et al., 2020)
3 Auto-grading of model	Papers that discuss the automatic instruments developed to assess the design/models.	(Waugh et al., 2004) (Prados et al., 2005) (Higgins & Bligh, 2006) (Prados et al., 2006) (Thomas et al., 2006) (Thomas et al., 2007) (Thomas, 2012) (Simanjuntak, 2015) (Thomas et al., 2015)
4 Task/model assessment	Papers that used learning theories/ taxonomies/ heuristics/guidelines in assessing tasks or design/model quality.	(Kesh, 1995) (Sinha & Vessey, 1999) (Lee & Chan, 2001) (Becker et al., 2008)
5 Feedback	Papers discuss the feedback provided as a part of summative and formative assessments. e.g., Automated or manual feedback	(Batra & Sein, 1994)
6 Academic integrity	Papers discuss the academic integrity and plagiarism of assessments.	(Kouznetsova, 2009)

2.3.4 RQ1.3 – How novice database modellers are assessed

Auto-grading of design/model

As indicated in Table 2.8, many of the articles were related to the auto-grading of database models. Auto-grading database designs is a difficult task, considering that there are often multiple potential solutions to a database design problem. Despite the challenges, a few articles have reported on attempts to automate the grading of student-generated database models. Higgins and Bligh (2006) marked student-generated ERD diagrams automatically by comparing the submitted diagrams' nodes, text, and

links to an existing model answer the authors called a “feature test”. However, the authors noted some shortcomings of their auto-grading approach for assessing ERD diagrams, such as the inability to mark the diagram’s appearance, capture the different possibilities of the design, and provide effective feedback. In a similar approach, Thomas et al. (2007) introduced a framework to compare student ERDs to a specimen solution diagram. Like Higgins and Bligh, their approach involved matching two diagrams, and a grade was computed based on the degree of similarity between the two models. However, unlike Higgins and Bligh (2006), Thomas et al. (2007) validated the computed grade by comparing it with the grade given by human markers. Thomas et al. (2007) noted less accuracy in the auto-grading method than the human grader when inheritance relationships were present in the model. Prior to this, the same group of authors had presented their experience using patterns in analysing the diagrams. In this earlier work, they discussed how patterns could be used to identify the sub-diagrams within an ER diagram and used this in a revised version of their auto-grading tool.

Prados et al. (2005) introduced a web-based tool to automatically correct relational schemas entered into their tool by students. In this tool, a student’s answer is compared to a “correct” instructor-supplied solution based on several metrics, such as the number of relations, primary keys, foreign keys, and corresponding attributes. An extension of this work by Prados et al. (2006), unlike the previous two approaches to auto-grading Entity-Relationship models, included the capability to compare each student’s model with several possible solutions rather than just a single solution using a two-layered comparison module. Comparing a student solution with more than one viable model solution better reflects the potential model solution space. Even though the tool’s impact was not empirically evaluated, the authors reported that the tool received positive feedback from both the teachers and the students.

While these auto-grading approaches have been implemented, most tools have not been empirically evaluated, so the claims are often purely anecdotal or based on student/staff perceptions. Several authors have noted the difficulty of auto-grading a model when the student’s answer, which may be perfectly reasonable, is in a form/layout that appears to be quite different from the possible model answers.

Skill development and acquisition

Another main area the articles were categorised was related to the skills and performance of novice designers. In one early piece of research, Batra and Davis (1992) compared novice and expert database modelling skills. This work attempted to understand the cognitive processes underlying the steps involved in conceptual data modelling. They found that experts approach the problem by first trying to gain a holistic understanding of the problem before modelling. In contrast, novices demonstrated an inability to integrate the various components of the problem scenario into appropriate knowledge structures despite having a similar understanding of modelling constructs to that of the experts. They

also observed that expert and novice behaviour was similar when modelling facets like entities, attributes, and binary relationships but were somewhat different when it came to complex facets like modelling ternary relationships and even more different when handling unary relationships. Another study conducted by Bock and Ryan (1996) around the same time delved more into the roots of what Batra and Davis had done. They reported on the experiments to identify the effect of the familiarity of the problem structure and its relationship with a novice's ability to model ternary relationships in Enhanced Entity Relationship (EER) diagrams. The authors discussed Tversky's psychological theory of familiarity to explain their findings. Tversky's (1977) theory of similarity explains a human's ability to recognize real or abstract objects by judging the similarity of the unknown object to other known objects using a feature-matching process. The authors tested this perception by asking participants to model a problem with ternary relationships where some ternary relationships had intersection data (additional data relevant to the relationship), and some did not. They conducted a controlled laboratory experiment where one group was given relationships "without intersection data" and the other "with intersection data" to evaluate the correlation between the presence of intersection data to modelling correctness and the identification of correct cardinalities. The authors reported no correlation between the presence of intersection data and the model's correctness. Still, they report uncovering a cognitive gap in the students' ability to identify the degree of a relationship that needs to be bridged. These were some of the earliest works that attempted to look at novice skill acquisition in database design.

In a later study, Batra and Wishart (2004) developed a conceptual framework for assessing novice designers based on rule-based and pattern-based training/learning approaches. Their framework allowed them to manipulate the complexity of tasks using combinatorial complexity, intricacy, and semantic mismatch of the data model. In their study, they manipulated the combinatorial complexity of a task by increasing the number of interconnected entities, leading to a combinatorial increase in the number of possible relationships. Intricacy has been added to the tasks with the claim that certain types of relationships are more complex than others by introducing ternary relationships to manipulate the task complexity. Semantic mismatch scenarios have been introduced to tasks by manipulating the novices' tendency to err in the direction of the familiar and expected. The researchers have manipulated these variables to develop tasks with three levels of complexity: low, medium, and high. Then, they conducted a laboratory study to compare the effectiveness of the two training approaches (between subjects) using three levels of task complexity (within subjects). They have used the performance of the database designers measured using the correctness of the task to experiment with the training approach and task complexity. Further, they reported no statistically significant difference between the training approaches. They suggested that novice designers' performance depends on the complexity of the task rather than the learning approach. This is an interesting fact since identifying further measures like what the authors have used to manipulate the task complexity is an important step in designing

assessments and a sequence of tasks as a research instrument for studies conducted to understand the learning of novice database modellers like in this thesis.

Other researchers have discussed the effectiveness of the data model representation method (data models organised in columnar fashion vs non-organised) on task performance based on the cognitive fit theory (Fuller et al., 2010). Vessey's (1991) cognitive fit theory suggests that the suitable fit between the task requirements and the graphical representation can improve performance. Fuller et al. (2010) utilised Vessey's theory to examine the performance of novice database designers when data models are organised (columnar fashion) vs non-organised. The results indicated that the columnar organised data model improved the performance of novices, specifically with respect to error detection and query construction tasks.

Another study looked at the effect of self-efficacy and cooperative team-based involvement on novice data modelling skills (Ryan et al., 2000). Based on a quasi-experimental study, the authors emphasised that self-efficacy is significantly related to the acquisition of data modelling skills. The cooperative learning approach raised the question of whether the cooperative team-based learning approach promotes skill acquisition among novices in short-term academic courses. Researchers have also discussed the utilisation of self-efficacy to understand the performance of database modellers (Dhillon & Dasgupta, 2011; Moores et al., 2006). These studies were related to skill acquisition and provided insights into several factors related to cognitive style, domain knowledge, metacognition, and self-efficacy that could affect the performance of database modellers. In summary, these studies all examined how novices understand database modelling tasks and the source of difficulties that typical novice database modellers face.

Assessment type

The Assessment type theme includes articles that introduced different types of formative and summative assessments. One researcher reported on their assessment strategy involving a computer-based quiz-style database modelling assessment. The type of questions utilised included: single-answer MCQs, multiple-answer MCQs, matching from a list of options, and true/false questions. They had some facility to include design diagrams in the questions, but this was limited. They first attempted to use this computer-based test formatively and later used it as a summative assessment. In conclusion, the study was only an experience report, but the author emphasised the need to carefully consider the design of questions in computer-based assessments of database modelling (Harrison, 2004).

Several other researchers have discussed various formative and summative assessment components like project-based assignments, in-class assignments, and group assignments to assess the modelling skills of novices (Etlinger, 2013; Harrison, 2004; Pittges et al., 2009). A group of authors discussed their take

on a project-based assignment where the students were asked to develop an online shopping cart application (Pittges et al., 2009). The assignment consisted of two main components: design and implementation. The students completed the assignment's design portion by writing an essay describing their design guided by a set of design questions. Interestingly, the authors argue against open-ended design, claiming they are rarely effective at an introductory level and hence discussed a design solution once the essays are submitted to ensure the students start the projects with a good design on hand. The students were asked to submit four deliverables in their projects: design essay, source code, test driver/s and release notes. This study brings forth an interesting way of assessing the novices with both front end and database aspects of an enterprise level application. However, the authors have discussed the challenges encountered in a project like this where the design skills required to complete the project exceeding the abilities of a novice, the lack of desire and skill in programming to implement the project and the dependencies among assignments that leads to students being penalised heavily due to their earlier stage mistakes. However, with the evolution of databases the authors suggest that students need to understand data persistence, scalability, transactions, and concurrency control along with good design principles in their introductory database course.

Etlinger (2013) introduced a contributing student pedagogy where students were responsible for the content and delivery of an assigned topic. In a similar study, the authors used an approach where students contributed to the learning of peers by collaboratively creating a bank of self-assessment questions (Paterson et al., 2012). In a recent study, Kawash et al. (2020) looked at the notion of peer learning by assessing their students using a method of group examinations called Graded Group Activities (GGA). In this study, the students attempted the examination individually without any feedback. Subsequently, they re-sat the examination as a group utilising an immediate feedback instrument where they were able to submit a solution to the same question multiple times until a fully correct answer was identified. The researchers did not find any correlation between the individual and group components of GGA. One weakness of this study is that the immediate feedback approach may have resulted in the students gaming the system rather than learning from it. However, the authors reported finding a strong correlation between the group component of the GGA and the performance of the students as individuals in their final examination when compared with the GGA individual component and the final examination performance. The researchers claimed that this is an indication that the students had benefited from the group GGA, though the improved performance might not have been solely due to the GGA experience.

Task/model assessment and others (feedback and academic integrity)

Very few articles discuss how novice database modelling tasks or models are designed or assessed using learning theories, taxonomies, heuristics, or guidelines. Researchers have also tried to identify the

relationship between the user's data modelling performance and the modelling constructs' complexity (Lee & Chan, 2001). Their objective was to propose a way to estimate the difficulty of a modelling task at the facet level. They derived the complexity based on three main aspects: component, coordinative and coupling complexity. Two semantic network variations were used in their study to measure the complexity, and these values were compared with the novice database designer's performance. The authors suggested that using semantic networks might be an appropriate way to estimate the modelling complexity and user performance. Although these studies have attempted to measure the database designer's performance or the complexity, it's still an area that needs further work to be conducted to validate the metrics.

Interestingly, the literature review only produced one article that discussed the freeloading issue in a database systems assignment (Kouznetsova, 2009). The author discussed an approach to team-based design that ensures that each team member is involved throughout the collaborative project. The project started as an individual assignment rather than a group project, where they were asked to submit a draft report of their design based on the application area assigned by the instructor. Once the first phase was over, the students grouped according to the application area assigned to them and started working on a single design. The new design was either based on one of their own designs or a hybrid version of it, and the students were asked to justify the reason for their selection. This mechanism allowed the instructors to assess the critiquing design skills of the students, and they argued that it also alleviated the problem of freeloading. In a very early study, Batra and Sein (1994) conducted a laboratory experiment to test the effectiveness of feedback when using the SERFER (Stimulated ER-based Feedback system for Relational databases). They have considered three theoretical perspectives in providing feedback: backtracking, which provided the concept and nature of feedback itself, the semantic and articulatory distance, which suggested that feedback should take the form of natural language as well as ER diagrams and the theory of working memory, which suggested that feedback should decompose the design task to provide feedback systematically. The experiment identified that feedback provided as natural language statements for each relation entered by the novice designers and up to date ER diagram of the relational solution was helpful for the users to detect certain errors related to modelling ternary relationships, but no such relationship was identified with unary and binary relationships.

2.4 Summary

This chapter has discussed the findings of the systematic mapping study conducted to understand the existing literature of the area, which may be used for the design and analysis of this research. The findings of the systematic mapping study based on the higher-level dimensions (type of research and focus areas of the existing research) of the classification scheme are summarised in Figure 2.8. Simon

(2020) conducted a bibliometric analysis of the first 22 years of the ACE conference, which arguably represents the type of work being conducted globally in CSER. He reported that nearly 40% of papers were related to programming even though the study didn't indicate the context of other related areas, and he further explained the phenomena as follows:

The growth of papers in the context of programming has taken place at the expense of other contexts. Capstone projects and software engineering each account for 4% of the papers, and all the remaining contexts have fewer papers still. (Simon, 2020, p. 205)

This substantiates the gap we found in the existing literature with a relatively low number of works identified in database modelling education.

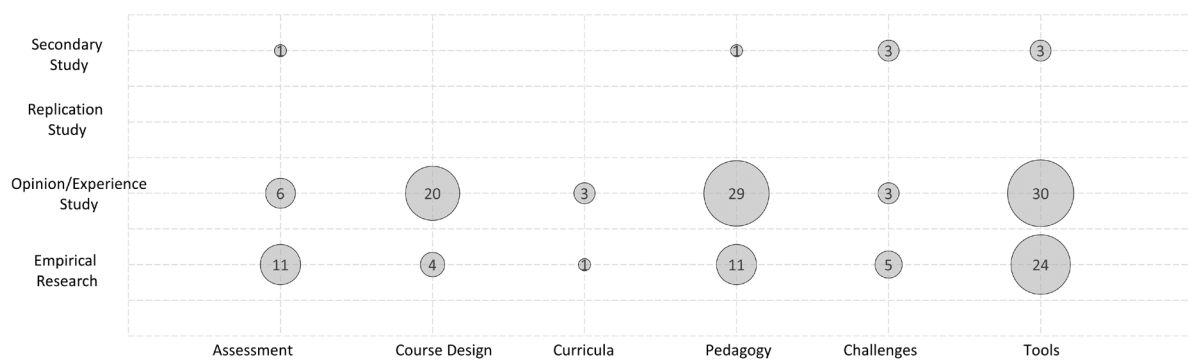


Figure 2.8 Visualization of a systematic map between the focused area of research (x-axis) and type of research (y-axis).

According to Figure 2.8, one apparent gap identified in the existing literature was the lack of empirical research and non-existent replication studies compared with opinion and experience studies. The opinion or experiential studies typically reported validation based on user experience surveys or uncontrolled comparison of student marks across different cohorts. These methodologies limit the findings in terms of general applicability. Replication studies are central to being able to generalize and confirm the research. The non-existing replication studies might reflect the focus on tools developed in-house and not openly available to other researchers. Moreover, most of the prior research (more than 50%), albeit not all, were experiential articles focused on Tools or Pedagogy. Furthermore, there appears to be a visible lack of research related to all the other identified focused areas. Apart from the higher-level dimensions analysed, further exploration was conducted on articles that discussed novice assessments. Not surprisingly, given the lack of articles in the area, only a limited number of studies were found that discussed the assessment of novice modellers. This area, in particular, warrants further study to understand how database modelling skills are developed and, in turn, how educators can assess these skills in a fair and appropriate manner. For example, how to develop formative assessments that

help students build these skills and scaffold development better when students hit concepts that are barriers to their learning progression.

Overall, there was a general lack of research in the introductory database education domain. From the conducted analysis, it was observed that many existing articles highlight the issues in teaching and learning database modelling but lack the initiatives to identify the root causes of these challenges. There were many tools and pedagogical interventions discussed in the literature. However, it's important to question whether there's enough work done to actually understand how the novices learn, the real nature of their struggles or the root causes of the issues to propose solutions. From a constructivist point of view, it is important to comprehend these causes before suggesting solutions (i.e., tools and pedagogies) to address them. Otherwise, proposing solutions without having a grasp of how these students work can result in less effective solutions that might be of no use.

Considering that understanding how novice database modellers learn to model was the focus of the work presented in this thesis, two main aspects emerged during the review of the literature deemed a good starting point for this work –the existing mechanisms that can explain the nature of the current database tasks given to novices and how to measure the complexity of these tasks, which can then be manipulated to conduct studies to understand the novices. The existing literature indicated some useful measures of complexity that can be used to inform the instrument design to conduct think-alouds with the novices. Further, some findings could be applied in the instrument design to overcome the cognitive load that arises in attempting database modelling tasks.

The overall objective of this research is to find an understanding of how novice database modellers learn to model, which was an apparent gap in the existing literature. The findings from the study might be applicable to achieving the objective of this research.

The next chapter focuses on the research methodology developed to conduct the research.

Addendum

As mentioned earlier, the literature search process was repeated at the time of writing this thesis to identify any relevant work published during 2021 and 2022, extending the SMS by two years. Twenty-one new publications were identified using the filtering process discussed in section 2.2. The publications were categorised based on the classification system developed as part of the doctoral research for this thesis to identify the type of research and the focus area.

Considering the venues of publications in 2021 and 2022, 8% were from conferences, 3% were journals and no symposiums. There has been a downward trend since 2011 in the number of related journal

publications. The analysis of the papers from 2020 onwards failed to find any particular change in trends related to the focus areas of publications beyond a modest increase in articles focusing on pedagogy.

According to Figure 2.9, more than half of the identified articles were categorised under the “pedagogy” category, where the majority were experience articles. Unsurprisingly, the majority of the published articles, as for the prior thirty years, related to project-based learning approaches (Borodzhieva, 2022; Shankaranarayanan et al., 2021; Weitzl-Harms, 2022). There were some articles on online and/or hybrid learning mechanisms (Meng & Chen, 2021; You et al., 2022) that were implemented mostly due to the COVID-19 pandemic. Borodzhieva (2022) discussed how PBL was used when they taught the topic of relational algebra in their databases course. In this approach, the students in a group of three were asked to develop an interactive application to visualise the basic operations in relational algebra using computer-based tools, MS Excel, MATLAB, and the platform LearningApps. All these software tools were used in other courses of their degree program hence, the students were familiar with the tools. Even though the outcomes were not empirically evaluated, the authors claim that PBL in an online course helps students learn and to develop many skills required for database design through practice, such as critical thinking, problem-solving, creativity, collaboration and communication. Another article discussed a service learning project implemented for a database course taught both on-campus and online (Weitzl-Harms, 2022). The students were asked to self-select a small database project for a real client. They were asked to interact with their client to discuss the project, elicit and analyse the requirements, and design and implement the solution. This is one of the real-world projects from the identified literature that has required students to work with a client on multiple occasions. The students were supported through the use of milestones and client feedback was integrated into the final evaluation of the students’ performance. While such an approach is common for capstone projects, it is not common in database courses. How this approach impacted student learning of database modelling and implementation was not discussed in the article and the authors suggested this as a future direction of investigation.

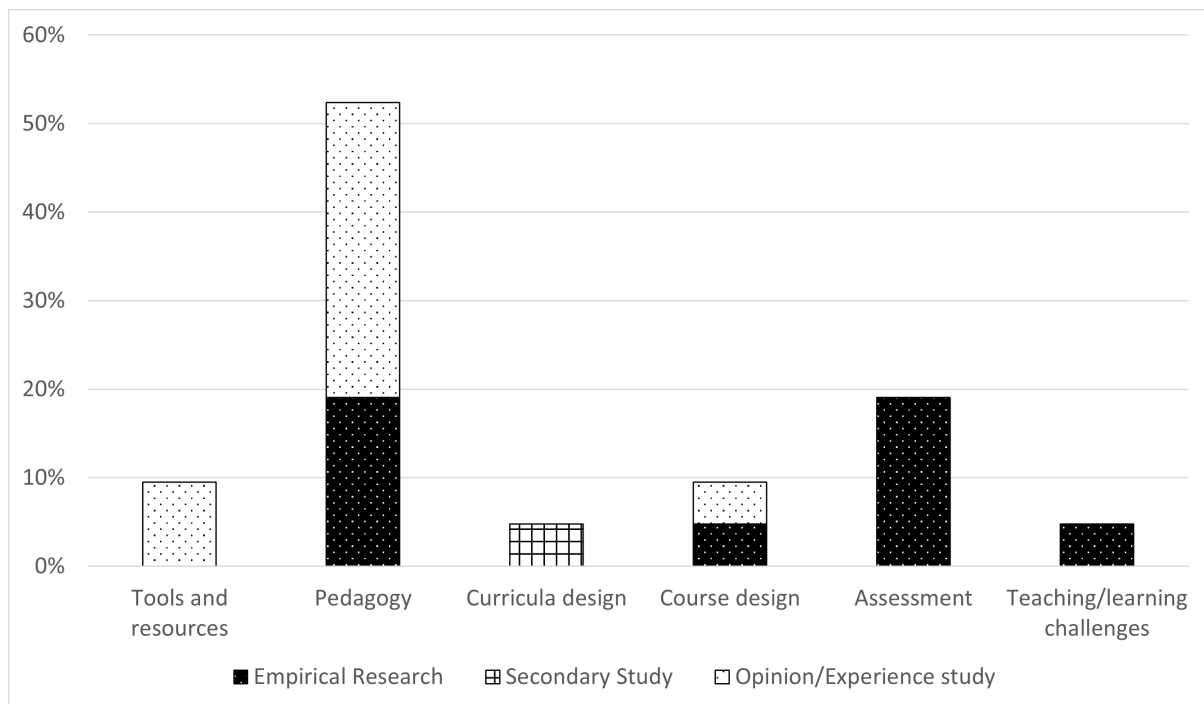


Figure 2.9 Focus areas of the identified articles based on the type of research for 2021 and 2022.

Another study integrated the BOPPPS (Bridge-in, Outcome, Pre-test, Participatory learning, Post-test, Summary) teaching model into their online teaching tool to stimulate the student’s interest in a mixed-method database course that involved a hybrid of online and in-person teaching. In their approach, they utilised the model to enforce the students to recall their prior knowledge and assess their strengths before doing the participatory learning activities.

Another published article examined the use of a collaborative learning mechanism, think-pair-share, in an introductory-level database course (Catania et al., 2022). The mechanism was employed in lectures/lab sessions and team peer review in projects with the intention of evaluating whether collaborative learning techniques could be beneficial in a database course. They ran a control and test group. The control group did not participate in the collaborative activities. They found that the students who passed and participated in collaborative activities had more consistent results than the ones who passed without participating in the activities.

The “Assessment” category had four published articles, all empirically evaluated studies. One of the researcher’s own articles that was drawn from the work reported in Chapter 4 of this thesis introduced a question classification framework to analyse novice database modelling tasks (Imbulpitiya et al., 2021). The tasks were assessed based on three dimensions of topics covered, question style and the intellectual complexity of the task. The intellectual complexity was measured using the cognitive dimension of the revised Bloom’s taxonomy and, to the best of my knowledge, the first study to analyse

the complexity of a database modelling task. This study identified a little variation in the topics and question styles provided to the novices. Further, the study identified gaps in the intellectual complexity of tasks where the examinations failed to provide tasks in Analyse and Evaluate levels of the revised Bloom's taxonomy's cognitive dimension.

Another study introduced an interesting assessment technique where they used association rules (a rule-based machine learning method used for discovering interesting relations between variables in large databases) to determine whether the student's performance in the database course affects the quality of the student's database designs produced in their capstone projects (Xu et al., 2021). The authors calculated each student's performance on database design based on a mark given to their developed ERDs, and then they performed various correlations with the student's GPA and grades for their Introduction to Database course. These correlations were used to identify appropriate association rules (e.g., For all the 11 groups who got C in overall ERD score, 9 groups did not take the Advanced Database course) based on an algorithm. Even though the work was preliminary, the authors have reported some interesting associations they have mentioned to be shared with the faculty handling the capstone project to better prepare the students for the capstone project.

Mason and Seton (2021) redeveloped an undergraduate database examination for non-English speaking background (NESB) students. They have provided students with dictionary-style definitions of the terms in several languages with the examination to reduce the extraneous cognitive load. To measure the effectiveness of the treatment, authors have compared the database examination with the performance in a programming course taken by the same student cohort. The authors found that international students in the database course performed significantly better than in the programming course.

As per Figure 2.9, the remaining thematic categories only had one or two articles classified into them. Interestingly, a secondary study was identified, which was categorised under the "curricula" category (Ishaq et al., 2022). This article was the first review, to our knowledge, that was published in the teaching and learning of databases area, and it focused on pedagogy, tools, and curricula. While the review conducted in this thesis was undertaken prior to Ishaq et al. (2022)'s review, it had not yet been published at the time of writing. The literature review conducted on this thesis spanned 32 years from 1990 to 2020, which was later extended till 2022, while Ishaq et al. (2022)'s review covered 27 years of literature from 1995 to 2022. The literature discussed in this thesis looked at the existing focus areas of research in more depth, while their work was only looked into in three areas: curricula, tools and pedagogy. Also, their venues were broader, including sites like ArXiv and Google Scholar, which might explain the larger number of articles (29,730) they identified compared to the 730 initial articles of this review. Further, in this review, AIS eLibrary was included to get access to database modelling-related

research in IS. Another major difference between the searchers was their inclusion of work in SQL, while in this review, the search was solely focused on database modelling. Interestingly, under their “Tools” theme, they have missed the work of the Intelligent Tutoring Systems research group from the University of Canterbury, New Zealand, which played a significant role in building intelligent tutoring systems in the novice database domain. Ishaq et al. (2022)’s study further discussed how the research in database systems education evolved based on their review of the past 25 years.

Chapter 3 Research Methodology

3.1 Introduction

This chapter presents an overview of the research framework designed to drive this research. Further, it provides a road map of the different research methods used in this thesis. A mixed-method pragmatic approach is taken since integrating both quantitative and qualitative methods is required to develop an instrument to better understand the factors that impact the learning of novice database modellers.

3.2 Research Approach

A mixed-method approach advocating a pragmatic worldview was adopted for this study. This approach integrates quantitative and qualitative methods in a single study and adopts and utilises the strengths of each technique to better understand the phenomena (Creswell, 2009). Pragmatist research designs often involve decisions based on '*what will work best*' to find answers to the questions under investigation. Several researchers have highlighted the significance of concentrating on the research problem and then combining pluralistic ways to acquire knowledge about the problem as the philosophical underpinning of mixed-method research (Morgan, 2007; Patton, 2002; Teddlie & Tashakkori, 2010; Thota et al., 2012).

Numerous scholars have talked about the applicability and feasibility of using multi-paradigm and mixed-method studies in computing education research. According to Fincher and Petre (2004), multi-method research can increase accuracy through triangulation in a single study or across multiple studies. Another article also highlighted the practicality of using a mixed-method approach to CSER to answer the research questions (Hazzan et al., 2006). In an analysis conducted on the teaching and learning of programming, the authors found mixed-method approaches were favoured in studies that focused on programming ability, aptitude, and understanding (i.e., teaching, learning, and assessing) (Sheard et al., 2009).

Since this research context and its inherent challenges are similar to the existing research that takes on mixed-method approaches, it was decided to use a pragmatic mixed-method approach in this study. Further, this approach will allow for a more complete and synergistic utilisation of data comparing quantitative and qualitative data. The quantitative data will be analysed statistically to develop a framework that can measure the difficulty of novice database modelling tasks that would inform the research instrument design of this study. This instrument will be then utilised to collect qualitative data from novice database modellers. Ultimately the two sets of findings will be triangulated and mapped to form an understanding of how students learn to model.

Thus, this research follows an accepted methodology where the focus is on using appropriate methods to answer the research questions and integrates both quantitative and qualitative methods to allow the researcher to better understand the factors that impact the learning of novice database modellers. Table 3.1 provides an overview of the utilised research methods in this thesis.

Table 3.1 Overview of the utilised research methods.

	Research method (Data collection)	Quantitative or qualitative?	Research method (Data analysis)	Related Research Question	Discussed chapter
1	Naturally occurring data (Database Examinations)	Qualitative	Open coding	RQ2.1	Chapter 4
2	Naturally occurring data (Students' solutions to database modelling assignments)	Quantitative	Statistical analysis: 1. Correlation test 2. Multiple regression 3. Principal component analysis	RQ2.2	Chapter 5
3	Think-aloud interviews	Qualitative	Reflexive thematic analysis	RQ3.1 and RQ3.2	Chapter 7
			Narrative analysis	RQ3.3	Chapter 8

As indicated in Table 3.1, several research methods will be used to collect and analyse the data.

A well-established approach in CSER is the use of naturally occurring data. Several researchers have utilised this method in their studies (Clear et al., 2009; Costantini et al., 2020; Sindre, 2020; Whalley & Kasto, 2014). Hence, naturally occurring data has been used as the data collection method for both phases 1 and 2 of the study, while think-aloud interviews were used in phase 3 (See Table 3.1).

At the beginning of the study, it was decided that an understanding of the nature of the existing database modelling problems given to novices was needed. To this end, database examination scripts were collected, and a data-driven coding method was used to develop a question classification framework. This framework was then used to classify a full set of modelling problems/tasks. A detailed description of this method and analysis is provided in Chapter 4

The main objective of this study is to understand how novices learn to model by understanding the processes they use and the challenges they face. Further, the learning trajectories of students with varying abilities were explored to understand how they learn to model. The idea is to create tasks of

varying difficulty levels within each modelling phase to facilitate the progression of the modellers. In order to build such a set of tasks, a framework is required to measure the difficulty of the tasks given to novice database modellers. Software metrics and educational taxonomies, an alternative to software metrics (see Chapter 4 for the question classification framework where one dimension was developed based on an educational taxonomy), were considered as potential measurements to develop the task difficulty. Several statistical analysis methods were used in developing these measurements, and the naturally occurring data from student solutions to assessments within a novice database course were used to conduct the analysis. A detailed discussion of the methods and the analysis is provided in Chapter 5

Once the tasks were developed, think-aloud interviews were used to collect the verbalisation of the thought processes of the novices while performing the tasks. The collected data were analysed using two different views with the aim of gaining insights into the difficulties and challenges faced by the novices using a reflexive thematic analysis (see Chapter 7) and comprehending the individual learning trajectories of novices using narrative analysis (see Chapter 8).

3.3 Ethical Considerations

Auckland University of Technology Ethics Committee (AUTEC) granted the consent of the ethics for this research. Three ethical approvals were requested for the three types of data collected, as given in Table 3.1. The ethics approval letters from AUTEC are provided in Appendix A.

The first ethics approval was gained to collect the naturally occurring data that occurred as a part of students' normal course work (students' solutions to the assignments they've done throughout the course), along with a prior knowledge survey. The data analysis was only started once the student's marks were finalised, and the participants were allowed to opt out of the study until before the commencement of the data analysis to remove any apparent power imbalance between the researchers and the participants.

The second ethics approval was sought to collect the database examination scripts and related course descriptors from other tertiary institutes. An invitation to participate was sent to all CS academics who subscribed to the SIGCSE mailing list, where they were asked to fill out the consent form and return it to the researcher. Once the consent was received, a link was sent to the participants to a submission form, which provided the facility to upload examinations and course descriptors via the request file feature of OneDrive. The raw data was anonymised at the point of collection and presented in an aggregated manner to protect the privacy and confidentiality of the participants.

The third ethics approval was obtained in order to recruit the participants for the think-aloud interviews. In getting the ethics approval for this, all the forms that were used during the participant recruitment and data collection were submitted to the committee. Considering that the researcher was a tutor for the courses where the participants were recruited in previous cohorts, and one of the supervisors was the lecturer in charge, several steps were taken to mitigate any perceived power imbalance and coercion to participate in the study during the ethics process. Even though the researcher was not involved in any teaching activities during the data collection, there was a possibility that a participant could be a repeating student. In such scenarios, they were informed about the fact that the researcher has no involvement or influence over the course or of their outcomes on the course. Further, the researcher undertook initial involvement with potential participants, data collection and data anonymisation. The supervisors only had access to the data in anonymised form.

3.4 Research Participants

The participants to collect the naturally occurring data and think-aloud data were recruited from students enrolled in the introductory database systems course offered in AUT's three-year undergraduate Bachelor of Computer and Information Science degree program. These participants were enrolled in INFS601 (Logical Database Design) and COMP508 (Database System Design), a mandatory course for students in the Bachelor of Computer & Information Science (BCIS) program. COMP508 was the course that replaced the INFS601 course due to a curriculum change that happened at AUT. While INFS601 was a level 6 course (equivalent to 2nd year), COMP508 is offered to level 5 students (equivalent to 1st year). Further, details of this curriculum change are discussed in section 6.2. Semesters at AUT are twelve weeks long, and students typically study three other courses concurrent with their database systems course. Both courses focused on providing the students with the essential concepts, principles and techniques in database management systems. They both had a particular focus on data modelling, design and implementation delivery, which took six of the twelve weeks of study. The courses were structured to differentiate between the different phases of data modelling and design (i.e., conceptual, logical, and physical phases). The relational data model was chosen as it is most commonly used in contemporary database applications. Both courses were mandatory for all students enrolled in the three-year degree program irrespective of their major (i.e., Digital Service, Networks and Cyber Security, Software Development, Data Science, and Artificial Intelligence). Further, it is also mandatory for the Software Engineering major in the four-year undergraduate degree program in Engineering. Hence, the course is designed to cater to a diverse set of students as an introductory database management course. Due to that, the two courses are closer to the type of courses usually adopted as part of Information Systems curricula at business schools rather than courses catered to only computer science students with more theory and mathematical representations.

Naturally occurring data was collected during two phases during the second semester of 2020 and the first semester of 2021 from the consented students who were enrolled in the INFS601 course.

The think-aloud study for this doctoral research was conducted in two phases during the first and second semesters of 2021. A pilot study was conducted during the 1st semester of 2021 prior to the main data collection phase. During the pilot study, the students were enrolled in the INFS601 course. However, in the 2nd semester of 2021, during the main data collection phase, the INFS601 course was replaced by COMP508, and the participants were recruited from the students who were enrolled in COMP508.

3.4.1 Recruitment

Recruitment was conducted in a way to ensure the informed and voluntary consent of the participants. Two main recruitments were conducted to invite participants to collect the naturally occurring data and to conduct think-aloud interviews. In collecting the naturally occurring data, both remote COVID-19 and in-person protocols were developed considering the prevailing situation at the time.

Recruiting participants to collect naturally occurring data

In collecting the naturally occurring data during the COVID-19 lockdowns, the researcher supplied the lecturer in charge of the course the invitation to participate to be placed on the course's Blackboard page – the learning management system at AUT. The researcher also appeared for ten minutes at the first online lecture of the INFS601 course to outline the research's purpose and nature. The invitation included a link to an online survey consisting of the participant information sheet, consent form and a prior knowledge survey (See Appendix B for the information sheet and consent form used to recruit participants to collect the naturally occurring data). Further, the students were asked to contact the researcher via email if they had any questions about the study. When there were no COVID restrictions, and face-to-face classes had returned, the researcher had a brief face-to-face in-class session during their lecture with the students, introducing the study, inviting them to participate, and allowing them to ask questions. The information sheets and the consent forms were circulated among students as hard copies.

Recruiting participants for Think-Aloud interviews

When recruiting the students to collect data in this phase, the same in-person protocol was followed as the one developed to collect the naturally occurring data (see Appendix D for the information sheet and consent form used for this think-aloud study). The only deviation from this process was that the researcher invited students to participate in their smaller lab classes (~50 students) rather than in the large lecture (~300 students). This ensured that information was provided more effectively and allowed

for more opportunities to ask questions about the research then and there. The students who consented returned a completed consent form online or in person, depending on the recruitment protocol. The students were free to withdraw from the study at any point before the data analysis phase.

3.4.2 Sampling methods

Two different sampling methods were utilised to select participants for this research study. One for selecting participants to collect data to identify the task difficulty predictors and one for the think-aloud sessions.

Sampling – To measure the difficulty of novice database modelling tasks

To develop a framework to measure the difficulty of novice database modelling tasks, naturally occurring data of all the volunteered participants were used without excluding anyone. This was mainly because sufficient data is required to conduct the statistical analysis to identify the correlations and determine the validity of the difficulty predictors. In order to generalise the findings of this phase, a sizable sample was required to represent the cohort. Further, an assumption was made that the cohorts the participants were selected from were representative of a typical class of novice database modellers. Recruiting participants in this phase was conducted using a non-probability purposive sampling technique.

Sampling – To conduct the reflexive thematic analysis and the narrative analysis

All the participants who fully completed all the think-aloud sessions required to complete the given tasks were considered for the reflexive thematic analysis since enough data was required to identify the common patterns from the data.

However, the cases to present the narrative analysis were selected from the group of students who completed all the think-aloud sessions based on a purposive critical case sampling method. Critical case sampling is a technique used when a single case or a small number of cases can be decisive or critical in understanding the phenomenon of interest. This technique is frequently applied in exploratory research, an approach that investigates research questions that have not been studied in-depth previously. The selected cases to be studied should be important cases that “*would yield the most information and have the greatest impact on the development of knowledge*” (Patton, 2002, p. 236).

Identifying the dimensions that make a participant a critical case is important in using the critical case sampling method. Two main dimensions were used to select the cases. The first, which ensured richer data, was the ability to think aloud while attempting the database modelling problems. The second dimension was the overall performance of the participants. It is important to select a group of

participants to demonstrate a range of performance levels and database design capabilities in order to provide a sufficiently broad view of the learning phenomena across the target population. The critical cases were selected from the participants who attended all of the scheduled think-aloud sessions to ensure a complete data set. Even though the findings cannot be applied to a larger population due to the use of critical case sampling, logical generalisations may still be made based on the extensive evidence gathered throughout the in-depth investigation with careful consideration. The participants were representative of the cohort based on their overall spread of marks.

3.5 Summary

This chapter discussed how a pragmatic approach was adopted to conduct this mixed-method research study by providing a road map to the utilised research methods. The overall research design was identified that comprised three main phases: 1) development of the question classification framework, 2) development of a typology of modelling tasks based on the task difficulty, and 3) identification of processes used and the challenges faced by novices and identification of their learning trajectories in order to understand how novices learn to build models.

The next chapter focuses on the development of a question classification framework developed to understand the nature of existing novice database modelling tasks.

Chapter 4 Classification and Analysis of Database Examinations

4.1 Introduction

In order to understand how novices learn to model, it's important to develop a research instrument consisting of novice modelling tasks that can be used in the think-alouds data collection phase. However, before developing these database modelling tasks to gain insight into how novices develop the necessary database modelling skills, it was necessary to comprehend the type of tasks used to assess them in a typical introductory-level course. Hence, one main objective of building the classification presented in this chapter was to understand how novices were assessed and what types of tasks are used. This knowledge could be used to inform the development of the research instrument used in the think-alouds. Another intention was to realise what's presented in the literature and what was actually utilised in reality to understand the nature of the existing tasks. Before developing the classification, the literature was searched in order to establish if any researchers had already developed a task typology for assessing novice database modellers. In order to achieve these objectives, this analysis addresses the following research question:

RQ2.1: What is the nature of database modelling tasks used to assess novice database modellers?

However, no studies were found that have attempted such a task within the domain of database modelling. But a number of educational taxonomies have been adopted in CSER, predominantly in the programming domain, to assess the nature of the tasks (Sheard et al., 2013; Thompson et al., 2008). This work was inspired by similar work in programming, and since this, to the best of our knowledge, is the first such attempt in database modelling education it was decided to focus only on the database examinations to collect the tasks. Arguably, using time-controlled examination tasks might not be the ideal solution for this classification since many courses involve at least one take-home assignment with a case study. However, it was decided to utilise the examination tasks as a preliminary point in devising this scheme.

This chapter presents the development of a question classification framework for the classification and analysis of questions in introductory-level database systems examinations. The chapter starts with a discussion of the existing frameworks utilised in CSER to analyse the difficulty of tasks given to the students. This is followed by an in-depth discussion of the application of Bloom's taxonomy as a predictor of task difficulty in CSER and data modelling in particular. Then, a classification scheme is developed to analyse the database modelling tasks based on the concept(s) covered by a question, the question style (e.g. multiple choice, true/false, etc.), and the intellectual complexity of the question. The intellectual complexity is measured using the cognitive dimension of the revised Bloom's taxonomy.

Each cognitive level of revised Bloom's taxonomy in relation to database modelling is then discussed with vignettes and guidelines. Finally, the results are presented and discussed.

4.2 Educational Taxonomies in CSER

Computer science educators have utilised various educational taxonomies to describe the complexity of tasks, classify student solutions with model solutions to tasks and estimate the overall difficulty of tasks (Fuller et al., 2007; Masapanta-Carrión & Velázquez-Iturbide, 2018). Much of this work is situated in the programming domain. The most adopted taxonomies have been Bloom's taxonomy (Bloom et al., 1956), the revised Bloom's (Anderson et al., 2001), and the SOLO taxonomy (Biggs & Collis, 1982). Bloom's taxonomy and its variants are typically used to explain the level of thinking that's needed to perform a task successfully.

Bloom et al. (1956) first introduced this taxonomy as a framework to describe learning outcomes or objectives of assessments in terms of the level of cognitive complexity based on the premise of constructive alignment (Biggs, 1996), that the assessments align with the learning outcomes. The original taxonomy was introduced with three domains: cognitive, affective, and psychomotor. The cognitive domain, where much of the work on curriculum development had been based, relates to the development of the mental competency skills of students. The affective domain relates to the changes in interest, attitudes and emotional behaviours of the objectives, while psychomotor pertains to physical skills.

Bloom's taxonomy classifies educational learning objectives into six levels of cognitive complexity, each building upon the previous one. These levels are presented in terms of complexity from simple to complex, starting from *Knowledge*, *Comprehension*, *Application*, *Analysis*, *Synthesis*, to *Evaluation*. As Bloom et al. (1956) indicated, all the levels after *knowledge* are presented as "skills and abilities", with the notion that *knowledge* is the necessary precondition for putting the skills and abilities into practice. The authors have provided brief definitions for each level in the Appendix of their handbook. For example, they define the *Application* level as "*the use of abstractions in particular and concrete situations*" (Bloom et al., 1956, pp. 201-207).

Later in 2001, a group of researchers published the revised version of Bloom's taxonomy titled *A Taxonomy for Teaching, Learning, and Assessment*, taking the attention away from the notion of educational objectives (Anderson et al., 2001). Unlike the original taxonomy, the revised version differentiated between cognitive thinking and problem-solving procedures using cognitive and knowledge dimensions. The knowledge dimension has four categories describing the knowledge levels that individuals can acquire: *factual*, *conceptual*, *procedural*, and *metacognitive* knowledge. In the original version of Bloom, all of the levels were expressed as nouns, but Anderson et al. (2001) argue

that it is more natural to think about these nouns as verbs because verbs describe the level of thinking needed to “do” the task in doing and successfully completing a given task at a given level, students to meet that level of cognition. Hence, the cognitive dimension of the updated framework replaces nouns with verbs to better reflect that learning is an active process. The cognitive dimension has six categories: *Remember, Understand, Apply, Analyse, Evaluate, and Create*. These six categories are associated with two or more specific cognitive processes, altogether nineteen, in the form of gerunds ending in “ing”. For example, *Recognising* and *Recalling* are associated with the *Remember* category. Further, the top two levels are essentially swapped in the new version, where the highest level of thinking has become “*Create*”. The reasoning behind this change was that *Create* represents the highest level of difficulty because, as Anderson et al. (2001) argue, the act of creation requires *Evaluation*. Figure 4.1 gives an illustrative comparison of the two versions.

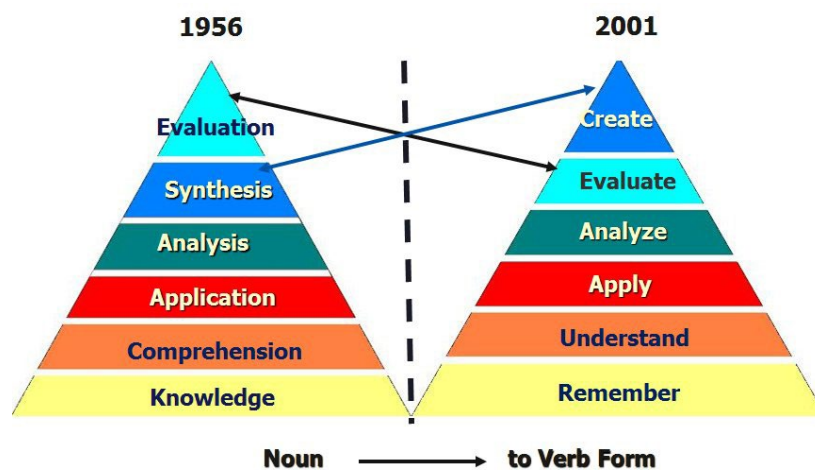


Figure 4.1 Comparison between Bloom’s vs revised Bloom’s taxonomy.

Note. From Bloom’s Taxonomy Revised: Understanding the Revised Version of Bloom’s Taxonomy, by L.O. Wilson, 2020, The Second Principle, <https://thesecondprinciple.com/essential-teaching-skills/blooms-taxonomy-revised/>. Copyright 2020 by L.O. Wilson. Reprinted with permission from the author.

Apart from Bloom’s taxonomy, the other widely adopted taxonomy in CSER has been the SOLO taxonomy (Biggs & Collis, 1982). SOLO is a model that can, like Bloom, be used to classify the complexity of the learning outcomes, allowing educators to evaluate students’ work in terms of quality. SOLO represents a continuum of increasingly integrated thinking in a student’s understanding of a subject and is described in terms of five hierarchical levels (Figure 4.2). In the lowest level, *Prestructural*, the student can acquire bits of information but cannot make any connections between them, and their knowledge has no structure and makes no sense. However, once the students start to see simple connections between the bits despite not fully understanding the significance of the connections, they start operating at a *Unistructural* level. As the students progress, they can see more connections

among the bits of information. However, they are yet to understand the full abstract idea, and the students are considered to be at a *Multistructural* level at this stage. At the *Relational* stage, students can finally see the connections and their significance in the context as a whole. Finally, the students operate at an *Extended Abstract* level when they are able to generalise and transfer their knowledge within and outside the subject domain.

In CSER, Bloom's, revised Bloom's, and SOLO taxonomies have been mainly used to describe the nature of different tasks, predominantly in novice programming (Castro & Fisler, 2017; Ginat & Menashe, 2015; Izu et al., 2016; Sheard et al., 2008). Apart from these main taxonomies, researchers have developed other taxonomies. For example, Fuller et al. (2007) proposed a taxonomy for computer science education that is specific to the discipline, which is a derivation of Bloom's taxonomy. At the time when this study was conducted, there was no published work that attempted to develop a classification system to estimate the cognitive complexity of database modelling.

However, there was similar work conducted in domain modelling using UML diagrams, which can be considered a subject area closely related to database modelling. For example, both class diagrams and ERDs are used for different purposes, where class diagrams represent the classes and the associations among them in a system, while ERDs represent the entities and the relationships between them in a database. However, the steps involved in developing the models are similar, like noun-verb analysis, which determines the potential classes/entities and attributes. Hence, it was decided to explore the possibility of using the cognitive dimension of the revised Bloom's taxonomy, relying on the work of Bogdanova and Snoeck (2019) in domain modelling to assess the cognitive skills required to complete novice database modelling tasks instead of other taxonomies to lay a foundation for such classification in database modelling.

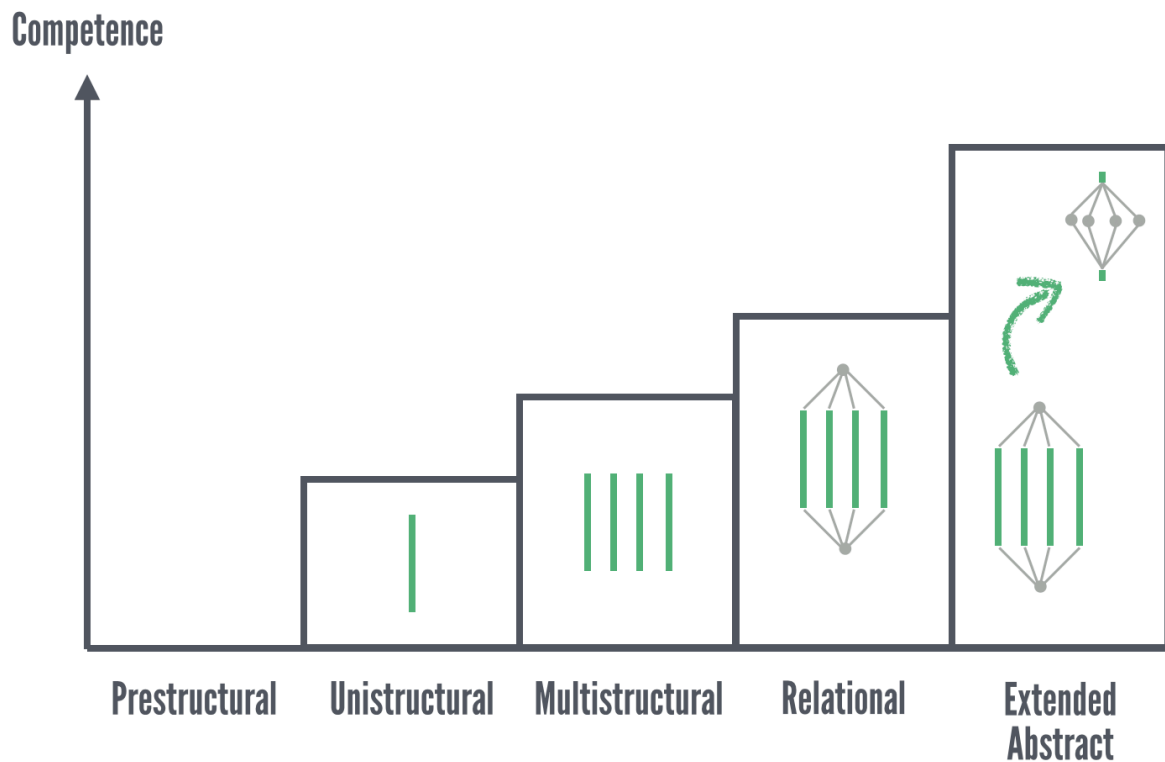


Figure 4.2 Levels of SOLO taxonomy.

Note. From Wikimedia Commons, by D. Belshaw, 2017,

https://commons.wikimedia.org/wiki/File:Structure_of_Observed_Learning_Outcomes_%28SOLO%29_Taxonomy.png. CC-BY CC0 1.0 public domain dedication.

4.2.1 Bloom's taxonomy and database modelling

As discussed earlier, there is very little work that evaluates mapping educational taxonomies to data modelling with the purpose of estimating the nature of novice modelling tasks.

Mohtashami & Scher (2000) highlighted the importance of developing summative and formative assessments in their database course using Bloom's taxonomy (Bloom et al., 1956). However, they do not go into detail about how Bloom's taxonomy was interpreted or how they categorised the assessment tasks using Bloom. But in their work, they point out the need to create tasks to enable students to move through the continuum of the levels: starting at the *Knowledge* level and then progressing up to the *Evaluation* level. Fuller et al. (2007) found that Bloom's and SOLO taxonomy are the most widely adopted in computing-related modules. They discuss how these taxonomies have been used in the computer science domain. Further, they also highlight possible issues of using these taxonomies using examples situated in different courses. In this, they included a brief example of a database course and a course related to software modelling, but their proposed two-dimensional adaptation of Bloom's

taxonomy was largely based on programming examples, which introduces doubt as to its applicability to other CS subject areas. These were the only studies that have attempted to measure the complexity of novice database modelling tasks or at least the ones that have mentioned such an approach. Due to this reason, the literature on domain modelling was also analysed.

A recent study by Bogdanova & Snoeck (2017) used the revised Bloom's taxonomy (Anderson et al., 2001) to classify learning outcomes that were derived from educational resources in domain modelling. To my knowledge, their study was the first to use an educational taxonomy to classify the learning objectives in the field of domain modelling. They identified similarities and gaps in these resources, namely the absence of tasks at the *evaluation level* and the tasks related to *procedural knowledge*. Their interpretation of the revised Bloom's taxonomy for domain modelling learning outcomes is provided in Table 4.1. In a follow-up study, the authors introduced an educational framework called CaMeLOT for conceptual data modelling based on the revised Bloom's taxonomy (Bogdanova & Snoeck, 2019). The proposed framework included three dimensions. The first dimension was the content area, which identified the concept being assessed, e.g., classes, modelling methods, model quality, modelling notation, etc. The authors provide detailed classification guidelines for the other two dimensions, *knowledge level* and *cognitive level*, which were based on the revised Bloom's taxonomy. This framework was not empirically evaluated however, the authors have requested feedback from experts regarding the proposed framework. The three experts provided mixed reviews on the framework but, at the same time, indicated the potential of the framework to improve their courses. The work conducted by Bogdanova and Snoeck (2017, 2019) inspired the classification scheme developed in this study, which also used the revised Bloom's taxonomy.

Table 4.1 The revised Bloom’s taxonomy’s cognitive levels in domain modelling.

Cognitive level	Interpretation: examples
Remember	recalling and reproducing previously learnt material: defining a modelling concept, naming a modelling notation.
Understand	interpreting previously learnt material (explaining, translating from one form to another), or by comparison: interpreting a provided model, providing a textual description of a given class or association, or giving an example (instance) of a given class or association
Apply	using the previously learnt information in new ways and/or implementing a learnt technique: modification of a given model, using a known technique to determine the appropriate association between two classes
Analyse	deconstructing material to understand its inner structure and the general principles of relationships between its parts: comparing two models for a domain or generalising a given model
Evaluate	make judgements based on given criteria, standards or guidelines: detecting errors in a given domain model by comparing it to a given description of the domain or choosing the model that describes the given domain best from a set of possible models and reasoning about that decision.
Create	use learnt material to create a new structure or enhance an existing one: construction of a model according to a given description or complete an incomplete model.

Note. Reprinted from “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitaya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

4.3 Research Method

4.3.1 Data collection

As mentioned in section 3.3, the data was collected with the approval from AUTECH. Please see Appendix A for the ethics approval letter and Appendix C for the related data collection documents. Nineteen introductory undergraduate-level database examinations from four countries and six courses were examined to develop this classification system (See Table 4.2). The examinations collected were relatively reasonable from 2011 to 2019 and included questions assessing database design skills. All the examinations were closed-book and paper-based. The Inclusion and Exclusion criteria presented in Table 4.3 were developed to screen the collected examination questions related to database modelling.

Table 4.2 Examination Demographics.

Institute	Country	Major	Year	No. exams	Duration (hrs)	Total Qs	Modelling Qs	Marks (%)
A	Canada	CS	3	1	1	12	7 (58.3%)	39
A	Canada	CS	3	1	1	12	10 (83.3%)	70
B	Canada	CS	2	1	2.5	31	10 (32.3%)	42
C	England	CS	2	1	3	11	7 (63.6%)	40
D	NZ	IS	2	6	3	4	2 (50.0%)	35-56
E	NZ	CS	2	1	2	14	7 (50.0%)	70
E	NZ	CS	2	2	2	13	7 (53.8%)	61,70
E	NZ	CS	2	1	1	10	5 (50.0%)	70
E	NZ	CS	2	1	3	16	12 (75.0%)	74
E	NZ	CS	2	1	3	18	12 (66.7%)	70
E	NZ	CS	2	1	3	19	11 (57.9%)	67
E	NZ	CS	2	1	3	20	12 (60%)	69
F	USA	CS	3	1	1	13	3 (23.1%)	25

Notes. Examination demographics include the year in which the undergraduate degree subject is taken (Year), the total number of questions per examination paper (Total Qs), and the number of data modelling questions per examination (Modelling Qs). In the % Marks column, if the column represents more than one examination, the range of marks allocated to data modelling questions in the examinations is provided. CS indicates Computer Science in the Major column, while IS denotes Information Systems, and NZ in the country columns indicates New Zealand, while USA indicates the United States of America. Reprinted from “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitaya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

A total of 122 questions were chosen based on the developed criteria for analysis from the collected examinations. When screening the questions, all the sub-parts of a question were counted as single questions, and questions not related to database modelling were omitted from the analysis based on the exclusion criteria. The examinations’ database modelling and design coverage ranged from 32% to 83%, with approximately 53% of the marks given to this area. Further, Table 4.2 indicates the total number of modelling-related questions derived based on the developed inclusion criteria.

Table 4.3 Inclusion (I) and Exclusion (E) criteria developed for question screening.

ID	Criterion
I1	Questions related to basic knowledge of database systems and data models.
I2	Questions related to conceptual and logical database models.
I3	Questions related to database normalisation, including functional dependencies and data anomalies.
I4	Questions related to distributed DBMS and DBLC, which are also associated with database modelling and design.
E1	Questions related to relational algebra.
E2	Questions related to SQL querying and manipulation of data.
E3	Questions related to transaction management and concurrency control.

4.3.2 Data analysis method

Several other computing education analyses were analysed in order to establish the classification scheme utilised in this study (Bogdanova & Snoeck, 2019; Simon et al., 2012; Thompson et al., 2008). Based on the existing work, three categories were included in the question classification framework: the concept(s) covered by a question, the question style (e.g., multiple choice, true/false, etc.), and the intellectual complexity of the question. Open coding (Glaser & Strauss, 1968), a data-driven approach, was performed on the data to identify the question style and the covered concepts. Course content plays a major role in determining Bloom's levels, and the secondary supervisor, who was the lecturer of the course, brought that knowledge into the researcher's interpretation. Open coding was conducted to get a consensus among the researcher and supervisors. This process was conducted until the saturation was reached at four examinations, where no new concepts or question styles emerged.

The intellectual complexity of the questions was classified based on the guidelines and vignettes (see section 4.4.3) developed using the cognitive dimension of the revised Bloom's taxonomy. Once the classification scheme was established, the author classified all 122 questions, including the ones used to build the classification scheme. In order to maintain the validity of the classification scheme, the author reclassified the questions three months after the initial classification. By this point, the researcher could no longer recall her initial coding. The Fleiss-Davies kappa statistic (Davies & Fleiss, 1982) was used to measure the intra-rater reliability of the question classification. The statistic indicated a good agreement (Banerjee et al., 1999) beyond chance ($\kappa = 0.83$, $\hat{p} < 0.001$).

4.4 The Classification Scheme

4.4.1 Concept dimension

A total of seven different database concepts were identified during the open coding process: *Theory – Data Models*, *Theory – Database Systems*, *Conceptual data models*, *Logical data models –Relational*, *Logical data models – Relational (Normalisation)*, *Distributed DBMS*, and *DBLC*.

When considering the concepts covered in the questions, it was decided to include questions related to *Distributed DBMS* and *DBLC* since the questions were related to database designs (see the examples provided in Table 4.4). Further, the questions related to logical data models were divided into two categories since there were two types of questions involving logical database models based on their design approach: top-down *Logical data model (Relational)* and bottom-up *Logical data models – Relational (Normalisation)*. The questions related to *Logical data model (Relational)* involved the questions that started from a general idea (e.g., case study) which led to the development of a logical database model. In contrast, the questions related to *Logical data models – Relational (Normalisation)* required the students to analyse specific details like data tables, forms or reports to identify any data anomalies and remove them using normalisation, resulting in a better design.

Table 4.4 Type of concepts in database modelling questions.

	Concept	Definition	Example
1	Theory – Data Model	The questions that assess the basic knowledge of data models include the definition of a model, types of models, model components, and data independence.	The basic building blocks of all data models are entities, attributes, relationships, and constraints. Please choose any THREE of these, and explain them.
2	Theory – Database Systems	The questions that assess the basic knowledge of database systems include the definitions of file systems and database systems, their importance and shortcomings, and database management systems.	Describe any two limitations in file system and explain how the Database approach and Database Management Systems have overcome these limitations.
3	Conceptual data models	The questions that assess the knowledge in explaining constructs in a conceptual model, reading and developing conceptual database models.	Consider the ER model given in Figure 1. This model represents the operations of a pharmacy chain. Please answer the following questions regarding this model. a) Can a pharmaceutical company have multiple phone numbers? If not, what do you need to do to allow this?
4	Logical data models – Relational	The questions that assess the knowledge in explaining the concepts using relational models or transforming a conceptual model into a logical model.	Transform the EER (Extended Entity Relationship) model developed into a relational schema. Show all relations and identify the appropriate entity and referential integrity constraints.
5	Logical data models – Relational (Normalisation)	The questions that assess the knowledge related to data anomalies, functional dependencies, and normal forms.	Examine the relation given below, which contains data about properties rented to clients in Auckland. It also shows details of the owners who own the property. Carefully analyse the relation and answer the following questions: a) In what level of normalisation is the above relation? Identify the primary key attributes. Briefly justify your answer.
6	Distributed DBMS	The questions that assess the basic knowledge of distributed database systems.	A multinational engineering company has decided to distribute its project management information at the regional level in mainland Britain. The current centralised relational schema is as follows (Primary key attributes are underlined). Given this information, answer the following questions: a) Explain the best distribution strategy for the BUSINESS and REGION tables.
7	DBLC	The questions that assess the knowledge of DBLC.	Discuss the distinction between top-down and bottom-up approaches to database design. Use examples to demonstrate your understanding.

4.4.2 Question style dimension

The second dimension of the classification scheme classified the questions based on the form of answer required by the question. Some of the questions were classified based on definitions of existing question types such as *multiple choice*, *short answer*, and *true/false* questions (Dietrich & Goelman, 2017; Simon et al., 2012). The remaining question styles were derived from the data, including *matching* (for example, a question where a list of definitions must be matched with a list of terms that may contain non-matching terms as distractors), *model* (a task that asks the student to develop models from scratch or parts thereof by mapping a model from one form to another), and *ordering* (e.g., a question where the students are required to reorder a jumbled set of steps in a process).

4.4.3 Cognitive dimension

In order to conduct the classification of the question's intellectual complexity, vignettes were developed to guide the process.

Several recent studies have produced vignettes and guidelines to classify introductory programming tasks using Bloom's cognitive dimension. In this research, the goal was to build a similar set of guidelines for novice database modelling tasks that was investigated as a predictor of task difficulty. This work was influenced by similar work conducted in programming (Lister, 2000; Sheard et al., 2013; Thompson et al., 2008; Whalley et al., 2006) and domain modelling (Bogdanova & Snoeck, 2017, 2019) fields.

A definition and a further explanation for each Bloom cognitive dimension category are provided with a vignette or example from the exams analysed. However, as highlighted in section 4.5.3, since there were no questions classified under *Analyse* and *Evaluate* levels, the author has introduced possible questions that could be suitable at these levels. It's important to keep in mind that while they were representative of the analysed examinations, the samples offered under these categories are by no means exhaustive.

Remember

At this level, the students are required to retrieve relevant knowledge like terms, concepts or modelling notations taught in the course from their long-term memory. *Remember* is the lowest level of the cognitive dimension. Anderson et al. (2001) suggested action verbs like *define*, *list*, and *name* help to define and signal tasks designed to target this level of thinking. Bogdanova and Snoeck (2019) have suggested questions like recalling or listing down learnt terms and concepts, matching definitions with

terms, duplicating a previously learnt model, notations or patterns, and drawing a graphical notation to a given term that can be written in this level.

Three examples of questions from the exams analysed in this research which were classified under the *Remember* level are:

1. List five important properties of a relation.
2. Define the terms i) Database and ii) Database Management System.
3. Discuss the distinction between top-down and bottom-up approaches to database design.

Discussion

Both questions 1 and 2 required the students to recall definitions and facts that were explicitly taught in the class. The students should have deposited these facts into their long-term memory with repeated exposure and usage throughout the course of the study. Action verbs (list, define, discuss) also help in recognising the intended cognitive level of questions however, this in of itself is not sufficient to determine that questions are at the *Remember* level of the cognitive dimension. The teaching context must also be considered (Gluga et al., 2012).

The third question appeared to be different from the first two questions on the surface. Without knowing the course content, it might be placed in a higher level cognitive category, possibly even *Analyse*, since the question requires students to go beyond mere understanding of the definitions to compare and contrast two different approaches to database design. However, since this difference is well explained and explicitly covered in the course, students just need to remember it to provide the correct response. This vignette demonstrates how important it is to have an understanding of the course structure when classifying exam questions using Bloom's taxonomy. When classifying question 3, it was assumed that the majority of students would operate at the lowest cognitive level to answer the question. However, not every student would necessarily respond at the *Remember* level if they did not specifically recall what was taught but knew enough about the concept to frame their own comparative interpretation.

Understand

At this level, students are required to understand terms and concepts and be able to interpret and determine the meaning of instructional messages. To answer a question at this level, the students must first remember and recall the previously learnt material and then construct some new meaning or interpretations based on the existing knowledge. The questions at this level can use action verbs like *explain*, *interpret*, *summarise*, *classify* and *compare*. The typical type of questions that can be written at this level, according to Bogdanova and Snoeck (2019), include translation between notation and text;

explaining concepts, models or tables; providing appropriate examples of particular modelling concepts; model and relating concepts with corresponding model elements.

Example one

The model shown below in Figure 4.3 is composed of four entities. Using the syntax of business rules that you have learnt in this course, write FIVE business rules that are reflected in the Entity Relationship Diagram provided.

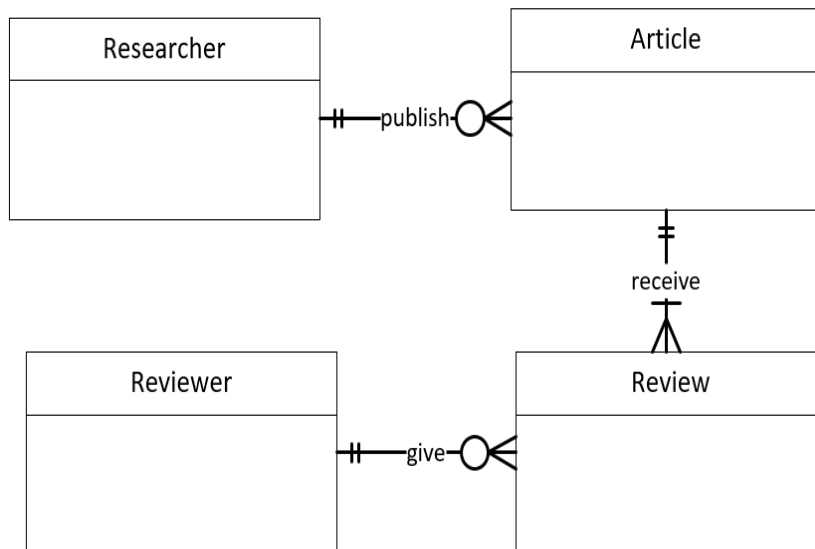


Figure 4.3 ERD diagram provided to extract the business rules.

Note. From “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitiya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

Discussion

This question requires the students to translate the ERD notations into business rules based on the relationship between two entities. When writing the business rules, the students should consider the relationship between two entities in both directions. For example, the business rule to express the relationship between articles and reviews would be “Each article must receive one or many reviews”.

Example two

The database shown below (Figure 4.4) is composed of four tables that reflect the following business rules: (1) “Each EMPLOYEE must have only one JOB_CODE, and each JOB_CODE must be assigned to one or many EMPLOYEES” (2) “Each EMPLOYEE must participate in one or many PLANs, and each PLAN must be assigned to one or many EMPLOYEES”.

For each table in the database, identify the primary key and the foreign key(s). If a table does not have a primary or foreign key you should indicate this in your answer.

Table name: EMPLOYEE		
EMP_CODE	EMP_LNAME	JOB_CODE
14	Rudell	2
15	McCade	1
16	Ruellardo	1
17	Smith	3
20	Smith	2

Table name: BENEFIT	
EMP_CODE	PLAN_CODE
15	2
15	3
16	1
17	1
17	3
17	4
20	3

Table name: JOB	
JOB_CODE	JOB_DESCRIPTION
1	Clerical
2	Technical
3	Managerial

Table name: PLAN	
PLAN_CODE	PLAN_DESCRIPTION
1	Term life
2	Stock purchase
3	Long-term disability
4	Dental

Figure 4.4 Tables provided for *Understand* example 2.

Note. From “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitiya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

Discussion

Both the business rules and the relevant tables are presented in this example. The students are required to find the primary key(s) that can uniquely identify each table (e.g., EMP_CODE in the Employee table) based on the business rules and the tables. In the case of the Benefit table, the primary key is a composite one since both the EMP_CODE and PLAN_CODE are needed to identify the data in the table uniquely. Since these keys also reference the Employee and Plan table, respectively, they also become foreign keys. The students must be able to identify the examples of primary and foreign keys in answering this question, thus demonstrating knowledge of entity integrity and referential integrity concepts, respectively.

Apply

At this level, students are required to solve a problem by carrying out or using a known procedure. The situation these questions are based on might be entirely new to the students, and they may need to decide on the appropriate method to solve the problem. However, the procedures are well-known and practised by the students. Bloom action verbs such as *choose*, *construct* and *apply* can further explain the questions at this cognitive level. The typical questions given by Bogdanova & Snoeck (2019) include: using a known design pattern (e.g. Line-Item pattern) to solve an object-oriented modelling task;

translating database models between forms by applying a known method or a set of guidelines; and applying a known method or heuristics to a new situation.

Example one

Transform the Enhanced Entity Relationship diagram provided (Figure 4.5) to a relational schema. In this diagram, for a university catering service, Staff may have many skills.

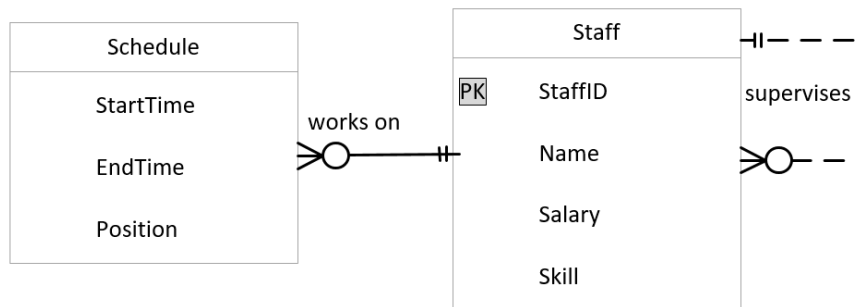


Figure 4.5 Part of the EER diagram provided for *Apply* level example 1.

Note. From “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitiya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

Discussion

To solve this question, the students must apply the rules they have been taught to use when transforming a conceptual model (e.g., ERD) into a logical one (i.e., relational schema). They need to be able to identify and transform different constructs like attributes (e.g., simple, multi-valued, composite), entities (e.g., strong, weak), and relationships (e.g., one-to-many, many-to-many) to the equivalent form in a relational schema. For example, the students should be able to identify *Skill* as a multi-valued attribute (in fact, it’s hinted at in the question stem). They then need to extract *Skill* as a separate entity and transform it as a relation into the relational model. The rules required to do this transformation are taught and practised in the course using various examples and scenarios.

Example two

Given the database table in Figure 4.6, which contains data about rental properties, the renters and owners, analyse the relation and answer the following question. At what level of normalisation is the relation?

C_No	P_No	C_Name	P_Address	RentStart	RentFinish	Rent	O_No	O_Name
CR76	PG4	John Key	6 Lawrence St, Auckland	1-Jul-13	31-Aug-14	350	CO40	Tina Murphy
CR76	PG16	John Key	5 Novar Dr, Auckland	1-Sep-14	1-Sep-15	450	CO93	Tony Brunt
CR56	PG4	Alice Stewart	6 Lawrence St, Auckland	1-Sep-12	10-Jun-13	350	CO40	Tina Murphy
CR56	PG36	Alice Stewart	2 Manor Rd, Auckland	10-Oct-13	1-Dec-14	375	CO93	Tony Brunt
CR56	PG16	Alice Stewart	5 Novar Dr, Auckland	1-Nov-15	10-Aug-16	450	CO93	Tony Brunt

Figure 4.6 Data table for the second Apply example. In the table, the prefix C = client, P = property and O = owner.

Note. From “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitaya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

Discussion

In the question, students tend first to identify the primary key in the given relation. Then, they need to identify the functional dependencies in the existing relationship to apply the taught steps to normalisation to figure out the existing normal form the relation is in. In this example, the relation is in first normal form (1NF) since there are no multi-valued or repeating attributes in the relation. For the relations to be in the second normal form (2NF), students must first find and remove partial dependencies. Similarly, they should identify and remove transitive dependencies to transform the relations into the third normal form (3NF).

Analyse

At this level, the students are required to break materials into their parts and detect how they work together as a whole. Anderson et al. (2001) explained this as “breaking material into its constituent parts and determine how the parts relate to one another and an overall structure or purpose”. This level is further described using action verbs such as *contrast*, *differentiate*, *organise* and *attribute* in the revised Bloom’s taxonomy. As mentioned earlier, no questions in the examinations we analysed were classified under this category. However, Object-Oriented (OO) domain modelling questions designed to operate at this level were proposed by Bogdanova & Snoeck's (2019) work. Inspiration can be taken from their work to design database modelling tasks that require analysis. Such questions might include mapping an analysis pattern to a given problem, explaining certain modelling choices, and determining the completeness and/or correctness of a given element (e.g., entity, primary key, etc.) within a model. An example question based on this idea might be something like: “**Identify** all entities and relationships in the case study provided **and develop** a set of business rules.”

In answering this question, first, students are required to conduct a noun/verb analysis to identify the potential entities and attributes. This task is an *Apply* level task. The students are applying a known process to select candidate entities and attributes in a new situation. In this process, they should be able to distinguish and deduce between nouns that actually indicate an attribute and ones that are really an entity and eliminate synonymous terms within a case study or scenario description. Once the entities and attributes are clearly identified, the students can then move on to identify the relationships and cardinalities between the entities to develop the business rules. Hence, this type of question can be themed under the *Analyse* level. But in order to be posed at this level, the problem domain should be new to the students, and the case study should include superfluous and irrelevant information. This irrelevant or unnecessary information, at least to a certain degree, makes the students filter out the relevant information in order to solve the question successfully. A question like this may be categorised as *Understand* if the problem description lacks extra and unneeded information or hiding the information in the detail and having to work out what information is superfluous.

Evaluate

The *Evaluate* level requires the students to make judgements based on criteria and standards, and the questions operating at this level are described using action verbs such as *criticise*, *evaluate*, *assess* and *justify* in the revised Bloom's taxonomy. As per Bogdanova and Snoeck (2017), the evaluation of models was significantly underrepresented in the materials they analysed in the OO domain modelling field. Later, they proposed questions that could be categorised in *Evaluate* level, such as identifying conventions/rules violations or critiquing a solution/the choice of example in a domain model.

The following is a question the lecturer of the course adopted into one of the assessments due to the identified gap, which is an example of an *Evaluate* level question. However, these types of questions might exist outside of examination papers, like in textbooks or take-home assignments, but none existed in the examination papers analysed for this study.

Example

A 'Fish Population' research effort is underway to record fish numbers in New Zealand. Observers go to various sites along rivers and make a note of the fish species and main body colour of the fish they see. The proposed Logical ER Diagram (given in Figure 4.7) for fish population observation shown below has various errors. Carefully study the entities and relationships; correct all missing/incorrect elements in the Logical ER Diagram, such as missing relationship names and incorrect cardinality and participation.

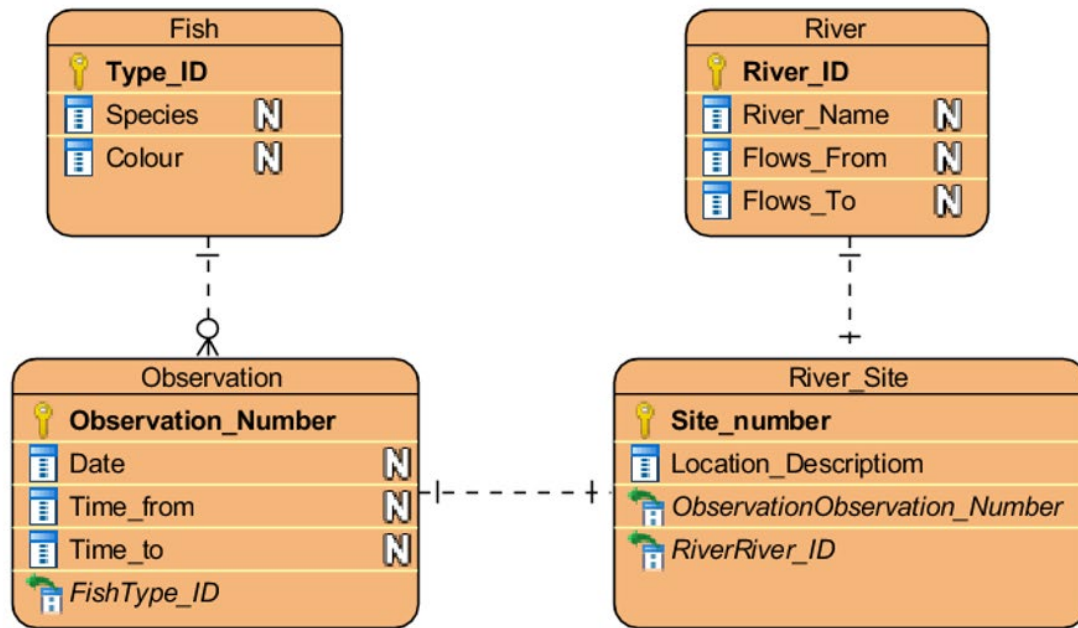


Figure 4.7 Logical ERD for the example.

Discussion

To answer this question, students must understand the problem description and critically assess the provided diagram to determine the existing errors. There might not always be a single right response. Instead, this type of question seeks to assess how well the student can critique, assess, and eventually identify mistakes to correct them.

Create

At the *Create* level, students are required to put elements together to form a novel, coherent whole or produce an original product. The action verbs that describe this level include *design*, *modify*, *develop*, and *construct*. All the questions categorised under this level in the examinations asked the students to develop a design artefact from a problem description or case studies of varying lengths and levels of detail. The following is a sample question categorised under this level. The actual case studies provided in the exams were more lengthier, typically around 400 to 500 words, which might include additional information or information hidden in detail.

Example

ABC Ltd. is in the business of repairing a variety of electronic devices, such as smart devices (phones, TVs and watches), laptops, tablets, and PCs. The company wants you to create a database to help them

run its business. Based on the provided information, draw a Logical ER Diagram using the Crow's foot model symbols.

Discussion

To develop a solution to this problem, students need to go through a few different steps. They need to first analyse the scenario to extract the business rules. These rules act as a blueprint to develop the ER diagram. In some of the examination questions analysed, such questions were scaffolded by supplying the question in stages, which act as a reminder to the students of the high-level steps that needed to be followed. For example, the steps include question parts for: determining all the entities (including all attributes) and relationships; determining primary key and foreign key attributes for each entity; resolving many-to-many relationships into two one-to-many relationships; and identifying weak relationships and strong relationships. Despite providing the steps, these questions were still considered to operate at the *Create* level since the students needed to apply the same intellectual thinking in order to develop the model from scratch.

4.5 Results and Discussion

4.5.1 Topics covered

Seven different concepts emerged related to database modelling from the examinations analysed. As per Table 4.5, the most assessed concept was conceptual data models followed by normalisation. Similarly, the highest average marks were allocated to the questions assessing the knowledge of conceptual data models. There were other questions that covered the topics like SQL, which were not included in the analysis however the average percentage of the mark was calculated based on the examination as a whole.

Table 4.5 Topics coverage and percentage marks by topic.

Topic	Coverage (%)	Mark (%)
Theory (Data Model)	16	5
Theory (DB Systems)	7	4
Conceptual Data Model	36	12
Logical Data Model (Relational)	7	7
Logical Data Model Relational (Normalisation)	24	9
Distributed BMS	6	5
DBLC	4	4

Note. Reprinted from “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitiya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

4.5.2 Question style

During the open coding process, five-question styles were identified from the examinations. Among them, the most frequent types of questions were *short answers*, and *model* questions (i.e., create a model based on a case study description) accounted for approximately 92% of the analysed 122 questions (See Figure 4.8 for the percentage coverage of question styles).

Even though there were no MCQs identified during the analysis, there is a possibility of missing such style of questions under a different assessment component. As an example, the COMP508 course at AUT had two quiz components where MCQs were utilised. So, this classification can be expanded to include the entire assessment programs of an institute, even though it was not done in this instance.

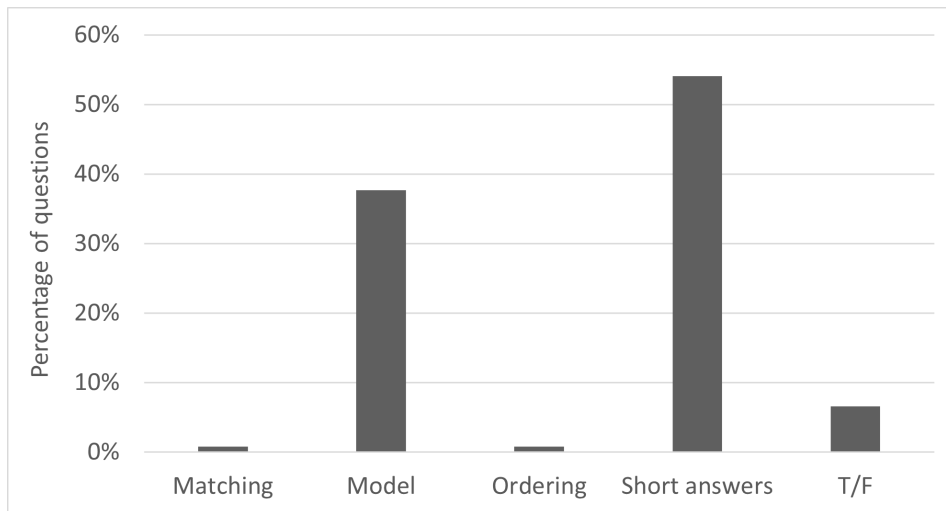


Figure 4.8 Coverage of question styles.

Note. Reprinted from “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitiya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

4.5.3 Intellectual complexity

All the analysed questions were either operating on the lower cognitive levels (*Remember*, *Understand*, *Apply*) or the highest cognitive level (*Create*) of the revised Bloom’s taxonomy. Interestingly, there were no questions categorised under the intermediate levels of *Analyse* and *Evaluate*. Figure 4.9 provides the details of the coverage of questions based on intellectual complexity according to the cognitive levels of the revised Bloom’s taxonomy. Further, the question styles were analysed along with the identified cognitive levels (See Figure 4.8). *Short answer* questions were the most frequented question style, mainly posed at the *Remember* and *Understand* levels. Unexpectedly, variations of question styles were observed for *Model* questions. The majority of *model* related questions were categorised into *Apply* and *Create* levels however, there were few questions posed at the *Understand* level as well.

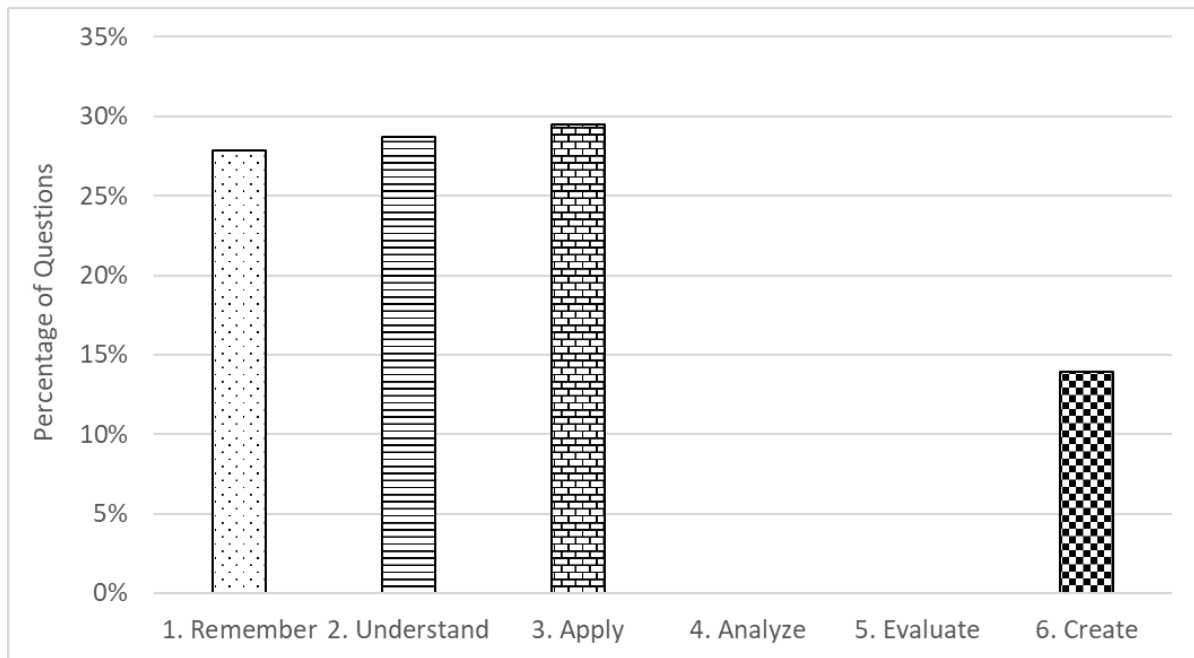


Figure 4.9 Coverage of revised Bloom's taxonomy's cognitive levels.

Note. Reprinted from “Examining the Exams: Bloom and Database Modelling and Design” by A. Imbulpitaya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

Figure 4.10 summarises the percentage of questions by question style and Bloom's cognitive level of a question. Based on the conducted analysis, it was noted that most of the questions were very similar in nature, question style and the concepts covered. It's evident that much work is needed to develop a wider variety of questions for summative and formative assessments to assess novice database modellers effectively.

Surprisingly, no Multiple-Choice Questions (MCQs) or variants thereof (e.g., Multiple Answer MCQs – (Petersen et al., 2016)) were among the analysed examination questions related to database modelling. Researchers have recently focused on developing more effective MCQs to assess novice programmers (Abreu et al., 2019; Lister, 2001, 2000; Simon et al., 2012; Woodford & Bancroft, 2005). These efforts have paved a pragmatic way to assess novice programmers' knowledge, particularly within large CS classes. Also, several of these studies have attempted to develop MCQs to assess Bloom's levels from the lowest cognitive level up to the *Evaluate* level. Apart from MCQs, another type of question that can be used to assess novice database modellers could be asking the students to identify and fix errors given a flawed model. Also, asking the students to complete a partially completed model or design by providing a problem description could be another possibility. Developing Parson's puzzles for modelling is another option that has not been attempted by the researchers in either the database or domain modelling area. These questions are quite important since they can assess the student's

comprehension, particularly when reading and understanding the model is vital to answering such questions. However, from the analysed examination questions, there were none that attempted to test the student's ability to read and explain a model.

There were gaps identified in the cognitive or intellectual complexity of the questions, which were quite similar to the findings discussed by Bogdanova and Snoeck (2017) in domain modelling. There was a clear rift between the *Apply* and *Create* level questions from the analysed questions, particularly with a lack of *Evaluation* level tasks. It's important to guide students through the cognitive process levels when assessing their knowledge of database skills (Mohtashami & Scher, 2000). It's quite a challenge to differentiate the majority of students in terms of achievements by excluding *Analyse* and *Evaluation* questions. In this way, the students who cannot work at the *Create* level but can analyse or even evaluate a database model are only given a chance to show off their skills at the *Apply* level. Essentially, this positions these students in the same category as those who can only operate at the lower cognitive levels of the revised Bloom's taxonomy.

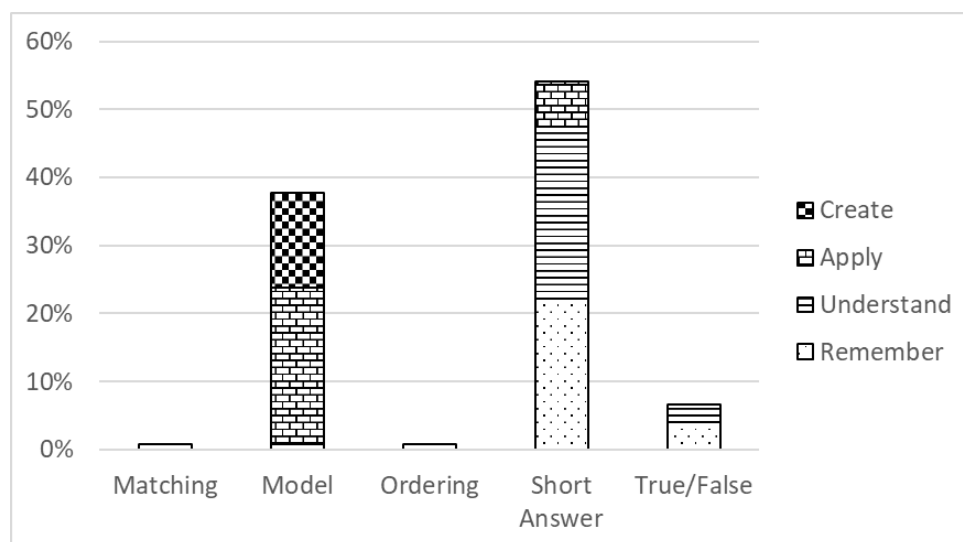


Figure 4.10 Coverage of Bloom's cognitive level with question styles.

Reprinted from "Examining the Exams: Bloom and Database Modelling and Design" by A. Imbulpitiya, J. Whalley, and M. Senapathi, 2021, Proceedings of the Twenty-Third Australasian Conference on Computing Education. <https://doi.org/10.1145/3441636.3442301>. Copyright 2021 held by the owner/author(s). Author's own work.

4.6 Summary

This analysis was conducted to investigate the nature of the existing questions used to assess novice database modellers in their examinations. Hence, a question classification framework was built that can be used to analyse the nature of the novice database modelling questions. In building this framework,

the main intention was to understand the different question styles used and the concepts tested in typical novice database assessments. Altogether 122 questions were analysed which were collected from 19 examinations, six different institutions and four countries. From the analysis, it was evident that there was, holistically, little variation between the different examinations. However, the extent to which database modelling concepts were tested varied. The lowest contribution of marks allocated for database modelling was 25% in institution F, while in institutions A and E, around 70% of the marks were allocated to assess database modelling skills. Further, most of the concepts tested were related to conceptual models (i.e., ERDs and EER diagrams) and normalisation. The styles of questions provided to the students were quite limited.

The intellectual complexity of the questions was classified based on the cognitive levels of the revised Bloom's taxonomy. Several guidelines and vignettes were developed that guide the use of the revised Bloom's taxonomy when determining the intellectual complexity of novice database modelling tasks. There were gaps in the cognitive levels of the questions: none of the questions were assessed at the *Analyse* and *Evaluate* levels. Arguably, it's important to assess these skills separately since they are critical in determining the level of competency achieved by the students while also being critical steps in the cognition required for the development of quality database models. In this context, the study is focused on novices, and this affects the level of cognition that a task requires. For example, what for an expert might involve *Remember*, for a novice, it might involve *Apply* or *Create* level thinking (Bogdanova & Snoeck, 2019).

The framework presented in this chapter was developed to gain an initial understanding of the nature of the questions given to the novices in order to help guide the development of the research instrument used for this research. The use of naturally occurring data, in the form of examination questions, is common in CSER and has become a recognised form of data for studies into what students know and how students are learning (Clear et al., 2009)³. Further, the use of examinations to do this is that there was a need to measure the student's difficulty where they successfully solved a problem as a proxy for the student's difficulty; controlled assessment ensures that the difficulty is that experienced by the student without assistance, which we could then correlate with the question's complexity.

³ Due to COVID AUT stopped controlled assessments which meant the data collection beyond the first phase of the study was not possible. This limited both the type of questions available for this analysis as well as the sample size.

Some of the results reported may be specific to the nature of the data source of examinations. However, given the same trend was observed by Bogdanova and Snoeck (2017), it is likely that this finding would hold when including other assessment items in a future analysis. Moreover, in the literature and in the courses that form part of this research, almost all courses appear to include a project-based take-home assignment. Such assessments typically provide students with a case study and require them to create design artefacts like models from scratch.

In deciding Bloom's level, an assumption is made that all the students operate at a similar level of understanding since the topic they encounter is relatively new to them. However, they might not operate at the same level realistically if they had some prior knowledge, which was a limitation. For this reason, the student's prior knowledge level was considered when conducting the think-alouds presented in Chapter 8. Another limitation when using the examination questions is the size of the task since the examination questions can be relatively less complex compared to modelling in a real scenario. Hence, it is important to be vigilant when making conclusions based on simpler modelling tasks. The next chapter focuses on the development of a sequence of tasks with varying difficulty levels by adapting various metrics identified in the existing literature.

Chapter 5 Measuring the Difficulty of Novice Database Modelling Tasks

5.1 Introduction

The proposed research instrument for the think-aloud interviews will be comprised of database modelling tasks in which concepts are presented to participants in sequences of increasing difficulty. In order to build these modelling tasks, it would be desirable to have a measure of the difficulty and complexity of the database modelling tasks given to the participants. This chapter begins with a look at potential metrics for determining the difficulty of database modelling tasks taken from the literature. While metrics have been studied in the context of the software engineering industry and large-scale systems, as far as I am aware, no research has been done on using database design metrics to predict the difficulty of typical assessment tasks given to novice modellers. As a result, the prior literature on applying metrics to assess the quality of database design at the industrial level was taken into account in this study.

A search and comparative analysis of the existing literature on database metrics was undertaken. It was evident from this focused review that the body of work on metrics to measure database model quality is much smaller than that for program code quality. Moreover, the review revealed that most database design quality metrics had not been empirically evaluated. Overall, this review conducted to identify database metrics revealed ten papers (Table 5.1), of which five were research studies conducted in the industry. The participants were experienced modellers, and their main focus was to compare alternative designs and identify the most effective or ‘best’ solution. The remaining studies involved either final-year undergraduate students or graduate students with relatively advanced database skills as participants, and the designs were more complex than those given out for novices. None of the studies found in the literature were focused on novice modelling.

During this review, it was inevitable that similar work conducted on object-oriented domain modelling would be discovered in the search due to the similarities in teaching and learning it has with database modelling. Even though object-oriented metrics were discussed in relation to program code, only one paper was found in the literature, which proposed a technique to assess student-generated object-oriented designs (Tempero et al., 2018). The metrics used were based on counting the objects created at runtime in the developed program. The authors’ argument is that the objects created are the reification of the design, which holds in an object-oriented design. Another study by Amrit and Tax (2014) looked at metrics that can explain the understandability of a UML model. However, their study was mainly focused on the modelling notations (i.e., symbol redundancy and ease and accuracy with which different

graphical symbols can be differentiated) and the aesthetic (i.e., size and placing of graphical elements, font selection, etc.) of conceptual model understandability. Apart from these studies, the only paper that was even remotely close to measuring the difficulty of novice modelling tasks in domain modelling using UML was conducted by Bogdanova and Snoeck (2019). They proposed an educational framework for describing the cognitive complexity of domain modelling tasks using the revised Bloom's taxonomy (Anderson et al., 2001). Their framework assesses the cognitive skills required to complete a task using the cognitive levels of the revised Bloom's taxonomy's cognitive dimension.

Because of the lack of work on using metrics to evaluate novice data modelling tasks, the work on using metrics to measure the difficulty of novice programming tasks was also explored to inform the selection of metrics to evaluate novice database modelling tasks. Several researchers have used software metrics to measure the difficulty of novice programming tasks (Elnaffar, 2016; Kasto & Whalley, 2013; Whalley & Kasto, 2014). Researchers emphasise that software metrics should not be the sole measure of difficulty in the context of novice programming tasks and that using a combined measure of metrics and cognitive complexity is needed (Elnaffar, 2016; Kasto, 2016). A similar argument was made by Moody (2003), who agreed that using a combination of measurements effectively solves the problem of evaluating the quality of data models.

The research discussed in this chapter aims to investigate the correlation between a set of database metrics and the difficulty level of novice database modelling tasks by addressing the following research questions:

RQ2.2: What type of measurements can predict the difficulty of database modelling tasks?

RQ2.3: Does the performance of students in database modelling tasks indicate a hierarchical structure in their mastery of different types of novice modelling tasks?

The primary objective is to determine if these metrics can be used to inform the design of the tasks for the think-aloud study. The chapter starts with a discussion of the candidate metrics source from the literature. Then, the selection of metrics for this study is discussed. Finally, a dataset for evaluation of the selected metrics is presented, followed by the presentation and discussion of the analysis results.

5.2 Database Metrics

Database model quality metrics measure the structural complexity of a given database model. This seems like a reasonable dimension to consider when attempting to develop a question difficulty framework because one aspect of the difficulty of developing a database model lies within the structural complexity of the model (Batra & Wishart, 2004; Genero et al., 2008).

Moody (2003) proposed 29 metrics to evaluate the quality of database models (ER diagrams). They also used an expert subjective rating to review the quality factors apart from the metrics—only five of the metrics managed to pass the cost-benefit validation conducted. Moody (2003) noted, “An unexpected finding was that subjective ratings of quality and qualitative descriptions of quality issues were perceived to be much more useful than the metrics” (p. 14). Considering Moody’s claim, it’s worthwhile to explore any subjective assessments of the quality of a database model along with objectively calculated metrics.

At the time of writing, only ten papers were found, excluding similar studies published by the same authors, in the literature that explored software metrics for measuring the quality of database models and none investigating their use in the model of novice database modelling tasks (See Table 5.1). Most of these previous studies investigated the use of static metrics to measure the quality of expert or industry-generated database models. These industrial-level database models contained far more entities and relationships than a database modelling task typically given to novices. The work also relied on experienced modellers as research participants rather than the novices being investigated in this thesis. The motivation for using metrics was also different in these studies, where the metrics were employed to compare modelling alternatives to identify the most effective solution rather than to determine the difficulty of the modelling task. Even the studies that involved undergraduate students (e.g., (Calero et al., 2001b; Genero et al., 2008)) used more complex designs, and the participants were in their final year of study. A summary of the related work identified in the literature is presented in Table 5.1. From this, it is concluded that none of the studies were designed to evaluate the database models of novices.

Prior studies conducted in programming (discussed in section 5.3) found that code readability was related to the difficulty of novice tasks. For this reason, it was decided that the readability of designs was worth investigating as one of the measures for this thesis research. However, in this case, readability metrics were derived from the case study text used to define the design problem. Here, the assumption is that being able to read and understand the case study is central to being able to complete the modelling task.

Table 5.1 Comparative analysis of metrics in to measure the quality of database designs.

Author/s	Domain	Measurement type+	Empirically validated?	Scope	Proposed Metrics
(Kesh, 1995)	ERD using Chen model	Objective and subjective	No	Applied to a database design project in an organisation	Behavioural Quality: Usability (user & designer), maintainability, accuracy, performance Ontological Quality: Structure – suitability, soundness, consistency, conciseness Content – completeness, cohesiveness, validity <i>Subjectively measured – suitability (5 point scale), soundness</i>
(Si-Said Cherfi et al., 2002)	EER diagram using Chen model	Objective	Yes – partially Validated only for the specification dimension	Hospital management case study with eight different schemas – data administration/data architecture group in the organisations.	Quality dimensions: Specification , Implementation, Usage <u>Proposed metrics for specification dimension:</u> Legibility: Ease of which a model can be read – clarity (no of inheritance links, no of roles in a relationship, no of line crossings), minimality (redundant element types of entity, relationship, and inheritance link) Expressiveness: no of concepts (element types) Simplicity: no of entities, relationships, inheritance links Correctness: syntactic correctness – no of characteristics to be verified for an element, no of errors
(Moody, 2003)	ERD	Objective and subjective	Yes	Applied to five projects in two organisations.	Proposed candidate metrics: 29 Useful metrics: 3 + 2 new metrics identified during the process.

					Metrics to measure eight quality factors completeness, integrity, flexibility, understandability, correctness, simplicity, integration and implementability. The metrics which passed the cost-benefit test: System complexity (E+R), Number of data items duplicated in existing systems/projects, development cost estimate and new metrics – reuse level, number of issues by quality factor <i>All quality factors were measured subjectively as well in 5 point scale.</i>
(Genero et al., 2002)	ERD using Chen model	Objective	Yes	Four Complex ERDs - 3 rd Year undergraduate students in computer science.	Proposed structural complexity metrics: 12 Number of entities (NE), number of attributes (NA), number of derived attributes (NDA), number of composite attributes (NCA), number of multivalued attributes (NMVA), number of relationships (NR), number of M:N relationships (NM:NR), number of 1:N relationships (N1:NR), number of binary relationships (NBinaryR), number of IS_A relationships (NIS_AR), number of reflexive relationships (NRefR), number of redundant relationships (NRR) Identified correlations between the understandability time (time to answer the questions based on the ERD) and the modifiability time (time to modeify the provided ERD) and the measures NE, NA, NR, N1:NR, NM:NR, NBinaryR.
(Genero et al., 2008)*	ERD	Objective and subjective	Yes After 3 pilot studies, only 6 metrics were selected for validation.	Eight complex ERDs - Professors and PhD students	Proposed structural complexity metrics: 12 Validated metrics: 6 NE, NA, NR, NIS_AR, N1:NR, NM:NR Understandability (UT): – time to understand the ERD (in minutes) Understandability effectiveness (UEffec) – number of correct answers Understandability efficiency (UEffie) – UEffec/UT <i>Understandability: Subjective – 7 point Likert scale</i>

(Calero et al., 2001a)	Relational, object relational and active databases	Objective	Partially with limited participants – only validated DRT and NFK	Final Year undergraduate students in computer science.	Database complexity Relational databases – Number of attributes (NA), Depth of referential tree (DRT), Number of foreign keys (NFK), Cohesion of the schema (COS) Object relational databases – schema size, complexity of references between tables Active databases – Number of attributes, Distance, Triggering potential
(Calero et al., 2001b) *	Relational database	Objective	Yes	4 oracle databases from an organisation - Final Year undergraduate students in computer science.	Database complexity Number of foreign keys, Depth of referential tree The controlled experiment conclude that the Number of foreign keys metric is a good indicator of relational database complexity.
(Pavlić et al., 2008)	Database type is not mentioned.	Objective	Yes	9 projects – organisations databases	Relation weight (W) = Number of attributes (A) + number of keys (K) + number of counted indexes (I) + number of foreign keys (F) Database complexity (DC) = $\sum_{i=1}^n (W)_i$, i = no of relations
(Mishra et al., 2012)	Relational database using ERD/ERD Chen model	Objective	Yes	5 Student projects – not explained clearly which levels of students	Proposed metrics: 6 Number of simple attributes (SA), multivalued attributes (MVA), many-to-one participation constraints (M:1P), descriptive attributes (DSA) – attributes in a many-to-many relationship and depth of inheritance tree (DIT), number of derived attributes (DA) Entity size (ES) = SA+MVA+M1:P+DIT Entity set size (ESS) = ES*M (No of tuples for each entity set) Total entity set size (TESS) = $\sum_{i=1}^n (ESS)_i$, n= total no of entity sets

Size of derived attributes (SDA) = $\sum_{k=1}^p$(Lines Of Codes)_k, p= total no of derived attributes Relationship size (RS) = DSA + Degree (Total no of entity sets participating in a relationship) Relationship set size (RSS) = RSS *N (No of tuples of the relationship type) Total relationship set size (TRSS) = $\sum_{j=1}^m$(RS)_j, m= total no of relationship sets Total Size = TESS+TRSS+SDA					
(Sinha et al., 2014)	Relational database	Objective	No	Public library database	Database complexity Database Complexity Indicator (DCI) is introduced based on designs Entity Relationship Schema (ERS) and Object Relational Schema (ORS). ERS – Number of entities, Number of tables, For each table – Number of attributes, number of foreign keys, Number of relationships with referential integrity, Number of entries ORS –Number of objects, Number of clusters, For each cluster – Number of entries

Note. * - Follow up study, the study that was the basis for this later work is presented in the previous row, + - measurement type indicates whether the factors are getting evaluated objectively (e.g., metrics) or subjectively (e.g., rating – using a Likert scale).

5.2.1 Structural complexity metrics

Towards a measure of the difficulty of novice database modelling tasks, the structural complexity of the model was included as a candidate metric since it is evident from the literature that structural complexity has been used with some success to identify the understandability of models (Genero et al., 2008; Moody, 2003).

Briand et al. (2001) explain how a complex structure of software (measured by the structural properties of an object-oriented design) results in high cognitive complexity. In building the hypotheses for their study, they utilised the causal relationship effect between the structural class properties, cognitive complexity, and external quality attributes. Further, the authors defined cognitive complexity and the effect of structural properties in a class with cognitive complexity as follows:

The structural properties of a class, measured by the coupling, cohesion, and inheritance measures, affect the cognitive complexity of the class. By cognitive complexity we mean the mental burden of the persons who have to deal with the class (developers, inspectors, testers, maintainers, etc.). (Briand et al., 2001, p. 4)

The work of Genero et al. (2008) theorises an ER diagram's structural properties affect the cognitive complexity that is associated with the diagram. Genero et al. introduced a collection of structural properties (e.g., number of entities, number of attributes, number of relationships, etc.), which they determined comprised structural complexity. Their study concluded that easily calculated metrics such as structural properties are an indicator to measure the understandability of ER diagrams.

Based on the existing work, using the structural properties of a database model seems an appropriate approach to identify potential metrics to measure the difficulty of novice database modelling tasks. A detailed description of how these metrics were identified for this study is in section 5.4.

5.3 Use of Metrics in CSER

Considering the lack of existing database modelling education work in measuring the task difficulty, it was decided to explore further in the CSER literature. This section presents a discussion of the use of software metrics in CSER with a focus on the area of programming. A notable amount of work has been carried out by researchers to apply software metrics to measure the quality of student-generated code. Other researchers have tried to adapt these software metrics to predict the difficulty of different types of novice programming tasks such as code comprehension, tracing, and writing. Two key types of measures have been explored: software metrics and readability metrics.

Early work related to software metrics investigated the use of metrics as a mechanism for providing formative feedback to novice programmers (Mengel & Yerramilli, 1999). Metrics were later evaluated

as a means of automating feedback on the quality of student code submitted for summative assessment (Cardell-Oliver, 2011; Pettit et al., 2015).

Other researchers examined the usefulness of software metrics as a way of estimating the difficulty of novice programming. They believed that metrics could potentially be used as a mechanism to guide the design of programming assessments. In a preliminary study, Kasto and Whalley (2013) empirically evaluated several static software metrics (i.e., number of statements, number of operands, cyclomatic complexity, etc.) in an attempt to predict the difficulty of novice code comprehension tasks. In their study, they concluded that some of the identified metrics correlated to the difficulty of code tracing examination questions but need further research to determine the potential use of software metrics in other types of programming questions. In a follow-up study, they identified several metrics that correlated with the difficulty of the program writing tasks and concluded that software metrics could be a beneficial tool for assisting in the design and selection of questions when creating an examination (Whalley & Kasto, 2014).

The work of Elnaffar (2016) utilised the findings of previous studies to construct a predictor of the difficulty of code writing tasks (Predicted Difficulty Index - PDI) by linearly combining the successful software metrics identified in prior literature. Despite building the predictor solely based on the metrics, Elnaffar emphasised the need to address the obstacles caused by code semantics, combining the amount of thinking necessary and the cognitive load put on students by a given question. Elnaffar (2016) further mentioned,

In future research, we may augment the code metrics with another dimension that addresses the cognitive level each question may require by using one of the common educational taxonomies such as SOLO. (p. 517)

This echoes an earlier comment by Whalley & Kasto (2014), who also noted that,

Metrics should not be used as a silver bullet but used in conjunction with more subjective measures of difficulty such as SOLO or Bloom's classification which consider the level of thinking and/or knowledge required. (p. 110)

Duran et al. (2018) proposed a set of cognitive complexity metrics for programming tasks. In their work, the authors emphasised the usefulness of metrics in providing a perspective on program complexity but also pointed out limitations discussed in the existing literature. They noted that metrics alone might not capture factors such as students' prior knowledge, which can significantly influence the perceived complexity of a program. They further mention the issue of having a low correlation between perceived complexity and metrics.

In a more recent study, Scalabrino et al. (2021) took a more comprehensive approach by considering various code-related (e.g., cyclomatic complexity, number of comments, number of operators etc.), documentation-related (e.g., comments readability, comments and identifier consistency etc.), and developer-related (e.g., programming experience) metrics to measure the code understandability. Their study failed to identify any metrics that correlate with code understandability. They suggested that code understandability could be too subjective to be effectively captured using metrics alone.

The conflicting findings in different studies can be attributed to several factors, including the specific programming tasks analysed or the context in which they are being performed.

Researchers have also examined the use of taxonomies alone or as part of a framework, including metrics (Castro & Fisler, 2017; Izu et al., 2016; Kasto, 2016; Muñoz Barón et al., 2020). All these earlier-mentioned studies used instructor examples to ascertain the metrics. Further, the difficulty of a task was based on student performance, either as a number of students who gave fully correct answers or based on a mark awarded directly. Then correlation statistics were applied to see if there were any statistically significant relationships between the metric (e.g., increased line of code) and the difficulty. None of these studies utilised the responses given by the participants to calculate the metrics. However, some previous work touched on the idea that there are many different possible solutions to a programming problem which would result in different metrics if the student solutions were analysed (Kasto et al., 2014). Some researchers have discussed how a pool of solutions can be created in the case of small code snippets from typical programming tasks and how the solutions would vary (Glassman, 2014; Luxton-Reilly et al., 2013). Given the existing work in programming and the highly abstract nature of database modelling, it would be safe to assume that similar issues are likely to occur in using metrics to assess the task difficulty in database modelling.

5.3.1 Readability metrics for novice programming tasks

In general, readability may be defined as the ease with which text can be read. Several well-known readability metrics have been applied to measure the readability of English text. These metrics typically involve counting the number of syllables, words and sentences in a piece of text and calculating a single numeric value according to some formula. For example, Flesch Reading Ease (Flesch, 1948), Flesch-Kincaid Grade Level (Kintsch & van Dijk, 1978), Gunning Fog Index (Gunning, 1952) and SMOG (McLaughlin, 1969) have been utilised to measure the difficulty of a text.

Börstler et al. (2007) mentioned that readability is a basic requirement for understanding a code, but they found no measurements that could be used to measure software readability at the time. Hence, following the idea of Flesch, the authors have introduced a Software Readability Ease Score interpreting different elements of a code. In an extension to their work, the authors looked at measuring the software

readability of code examples given to students learning object-oriented programming to examine their understandability of good code examples (beauty) from bad code examples (beast) (Jürgen Börstler et al., 2016).

Starsinic (1998) adapted the Flesch-Kincaid readability grade as a Perl script that could be used to measure the difficulty of simple programs. He designed the metric to support and develop junior programmers within his company. Whalley & Kasto (2014) used an adaptation of Starsinic's readability metric to measure the readability of small pieces of code written by novice programmers. Similarly, many researchers have since used readability metrics to identify the level of understandability of a code and have investigated the application of readability metrics as a predictor to measure the difficulty of a program code. Readability may be considered a reasonable metric since empirical research has found a strong relationship between the ability to comprehend and write code (Lopez et al., 2008).

The question of whether readability can serve as a metric for predicting the difficulty of a novice database modelling task has emerged when going through the above-discussed work in programming. It is indeed plausible that readability could play a major role in determining a modelling task's difficulty since often novice modelling problems are presented in the context of a brief and less complex problem space than in real-world systems, and they are presented with the problem as a textual statement or a case study (Shebaro, 2018). However, at the time of writing this thesis, no published literature has examined the effect of the problem statement on the difficulty of the database modelling task to the best of the researcher's knowledge. But, the issue of problem context has been researched extensively in mathematics and other fields. As discussed earlier, there has been some work in CSER that looks at problem context and its influence on the difficulty of novice programming tasks. The literature that came closer to this idea in CSER was the work of Leinonen et al. (2021), which investigated the effects of contextualised problems on problem-solving among introductory-level programmers. Their research has demonstrated that the context in which problems are presented can impact student performance, either positively or negatively, depending on the student's familiarity with the problem. They specifically examined the linguistic complexity of problem descriptions to determine the readability of the given tasks for students.

Given the existing research on measuring readability in the context of novice programming, particularly the work in problem descriptions, it is reasonable to consider adopting a readability metric to measure the linguistic complexity of the provided case studies to novice database modellers, which could use as a predictor for task difficulty.

5.3.2 Measuring task difficulty in CSER

Researchers in CSER have utilised taxonomies such as Bloom's and SOLO to predict the difficulty of novice programming tasks. A discussion of these taxonomies and the use of revised Bloom's taxonomy's cognitive dimension to classify novice database modelling tasks are provided in Chapter 4. Bloom's taxonomy is a widely recognised hierarchical framework that categorises cognitive skills required to complete a task at increasing levels of complexity, which is expected to be matched to task difficulty. In fact, researchers in CSER have explored these taxonomies to empirically evaluate whether the taxonomies can really predict the perceived difficulty experienced by novices when engaging in programming tasks.

Another important aspect of this existing research was to explore how they evaluate their difficulty variables. The researchers have used multiple methods or approaches to define the difficulty of a task, mainly using the students' perceived difficulty (e.g., (Duran et al., 2018)) or using fully correct answers (e.g., (Whalley & Kasto, 2014a)). Some researchers have discussed the fact that students' perceived difficulty alone might not be a good indicator to measure the difficulty of a task since their knowledge and experience can influence their level of perception, hence the students' perceived difficulty was not applied in this thesis's analysis (Genero et al., 2008). Instead, the perceived difficulty of a task was considered from the instructor's perception along with a performance-based measure. One issue that arises when identifying a performance-based measure is that during the marking of assessments, partial marks are allocated to solutions that may not be fully functional. Researchers often aim to assess the difficulty of a task by examining the percentage of students who can successfully answer a question rather than focusing on partial answers that may not align closely with the correct solution. While partial answers may contain elements of the correct answer by chance, they may still deviate significantly from the expected outcome. This is why opting for fully correct answers could be a preferred approach for measuring task difficulty and preferably an approach that was considered in this analysis.

Suppose an appropriate set of metrics can be identified and combined along with the cognitive classification presented in Chapter 4. In that case, the difficulty framework may be able to measure the difficulty of novice database modelling tasks reliably.

5.4 Protocol for Metrics Selection

As discussed in section 5.2, the database-specific metrics presented in the literature were for larger models and for use in industry and, as such, may not be relevant to smaller-scale designs based on constrained scenarios used in teaching and learning. Moreover, this thesis work comprised the first attempt to apply metrics to measure and predict the difficulty of novice modelling tasks. However,

several quality metrics were discussed in the literature that could be relevant, but this raised the question of which of these metrics are worth evaluating in the novice context.

The GQM (Goal-Question-Metric) paradigm is a widely used systematic approach in software engineering and management that aims to improve the quality of software development processes and products (van Solingen et al., 2002). Researchers have used the same approach to define metrics that measure the difficulty of novice programming tasks (Kasto, 2016). Hence, it was decided to apply the same approach in this study to identify potential metrics that can be utilised to measure the difficulty of novice database modelling tasks. The GQM approach involves three main steps. The first step is the *conceptual* level, which involves identifying specific goals that need to be achieved, which should be clearly defined and measurable. Once the goals are established, the second step (aka *operational* level) is developing the questions. These specific questions are developed to determine whether the goals are being achieved and should be answerable with objective data. The third step is named the *quantitative* level, where the metrics are defined that are used to collect data and answer the questions, which should be specific, measurable, and relevant to the questions. The data collected in this study is objective, so only the *objective* dimension of the *quantitative* level was considered to identify the potential metrics. This GQM approach was used in this study to identify the metrics that would be evaluated.

From the potential metrics given in Table 5.1, some metrics were directly eliminated even before the GQM selection. Specifically, the ones related to the behaviour of the physical databases, considering that several past research was focused on the industrial domain, which was not relevant to the scope of this study. For example, one of the few empirically validated metrics suggested by Moody (2003) was based on the data model's business and technical dimensions. As an example, the quality factor *implementability* was measured using the development cost estimate metric, which is unsuitable in the education domain. Also, some metrics were ruled out due to the diagram types which are not taught for the novices used in this study. For example, the metrics proposed by Sinha et al. (2014) are based on the Object Relational Schema diagram.

Therefore, only the structural metrics from database models were considered in this research to measure the complexity of the models as they were more appropriate in the context of education and, more specifically, in assessing novice database modelling tasks. In addition, readability was considered as a potential metric since database modelling tasks typically involve reading and understanding case studies before the modelling process. A detailed discussion of the metric selection using the GQM approach is discussed in section 5.5.

5.5 Selection of the Metrics

Goal

The main goal of this study was to identify metrics that are able to distinguish between different levels of complexity in the context of novice database modelling tasks to measure the difficulty of tasks. As identified from the prior literature, it was deemed suitable to look at the metrics related to the structural properties and the readability of the case study to measure the difficulty of novice database modelling tasks.

Questions

The following were the developed questions to achieve the goal.

1. Do the structural properties of a database model relate to the difficulty of novice database modelling tasks?
2. Does the readability of a case study affect how difficult it is to develop a database model?

Once the goal and questions were established, metrics were selected, as discussed below.

Metrics

In order to answer the first question, several structural properties related to database models were considered from the prior literature. These structural metrics are presented in Table 5.1. Among the existing work, the research framework introduced by Genero et al. (2008) provided the metric suite that best fits the type of modelling tasks given to novice learners. Genero et al. (2008) theorised that an ER diagram's structural properties affect the cognitive complexity associated with the diagram. Since cognitive complexity is somewhat difficult to measure, they have introduced a collection of structural model properties that are categorised as "structural complexity." They have proposed several metrics focused on the modelling elements to measure the impact of the structural properties on the understandability of a model.

As provided in Table 5.2, Genero et al. (2008) have proposed twelve structural properties as metrics to measure the structural complexity of an ER diagram. Among these candidate metrics, eight were selected for this study (indicated from M1-M8). In the context of novice database modelling tasks, some metrics (M10-M12) were eliminated from consideration. This decision was made because these concepts were not covered or assessed in the novice tasks. Further, M13 was also not used as a separate metric since all the relationships in the diagrams developed by students were binary, which was the same as the number of relationships.

Table 5.2: Selected and eliminated metrics.

Goal		To measure the difficulty of novice database modelling tasks.
Question 1		Do the structural properties of a database model relate to the difficulty of novice database modelling tasks?
Selected metrics		Metric Definition
M1	Number of Entities (NE)	The Number of entities within an ER diagram, including the weak and strong entities.
M2	Number of Attributes (NA)	The number of attributes defined within an ER diagram, including the multi-valued and composite attributes. The attributes subtype entities inherited from supertype entities are only counted once.
M3	Number of Composite Attributes (NCA)	The number of composite attributes within an ER diagram.
M4	Number of Multivalued Attributes (NMVA)	The number of multi-valued attributes within an ER diagram.
M5	Number of Relationships (NR)	The number of relationships within an ER diagram, excluding the ISA relationships. The resolved many-to-many relationships in logical ERDs were counted as one relationship.
M6	Number of ISA Relationships (NISAR)	The number of ISA relationships within an ER diagram. Considered each supertype/subtype pair as one ISA relationship.
M7	Number of one-to-many Relationships (N1:MR)	The number of one-to-one and one-to-many relationships within an ER diagram.
M8	Number of many-to-many Relationships (NM:NR)	The number of many-to-many relationships within an ER diagram. The resolved many-to-many relationships in logical ERDs were counted as one relationship.
Question 2		Does the readability of a case study affect how difficult it is to develop a database model?
M9	Flesch-Kincaid Grade Level	
Eliminated metrics (related to structural complexity)		
M10	Number of Derived Attributes (NDA)	The number of derived attributes within an ER diagram.
M11	Number of N-Ary Relationships (NN_AryR)	The number of N-Ary relationships within an ER diagram.
M12	Number of Reflexive Relationships (NRefR)	The number of reflexive or recursive relationships within an ER diagram.
M13	Number of Binary Relationships (NBinary)	The number of binary relationships within an ER diagram.

Note. Reproduced from “Defining and validating metrics for assessing the understandability of entity–relationship diagrams”. by M. Genero, G. Poels and M. Piattini, 2008, *Data & Knowledge Engineering*, 64(3), pp. 534–557. Copyright 2008 by Elsevier. Reproduced with permission from Elsevier.

To answer the second question, the literature in CSER was considered where the researchers have attempted to use any metrics to measure the linguistic complexity of a problem description. As mentioned in section 5.3.1, the only work that has used readability metrics to measure the linguistic complexity of a problem statement was by Leinonen et al. (2021), where they used Flesch-Kincaid Grade Level to identify the readability of the problem descriptions of their assignments in a CS1 course. Even though they have claimed that any difference in the student performance is unlikely due to the complexity of the language of the problem description, it was decided that Flesch-Kincaid Grade Level is a suitable metric to measure the readability of the provided case studies considering that these could potentially be more complex than the problem statements used in Leinonen et al. (2021)'s study. Hence, to answer the second question, the Flesch-Kincaid Grade Level was selected (see M9 in Table 5.2).

Once the metrics were selected, the next step was to evaluate these metrics to check whether they were effective at measuring the difficulty of a modelling task. This is presented in the following section 5.6 in detail.

5.6 Metrics Calculation

In calculating the metrics M1 to M8, an attempt was made to identify a software tool to facilitate the calculation of these metrics. First, auto-grading tools from the previous literature in the area were evaluated for this use. None fully facilitated the calculation of the metrics but instead made a comparison of the elements in a single student model and the instructor's model (Higgins & Bligh, 2006; Thomas et al., 2007). Additionally, none of these tools was freely available and seemed to be restricted to use in research only and not beyond the particular study. Next, functions provided by open-source and freely available CASE tools were examined to see if an automated calculation of metrics was possible without the need to code solutions. The "visual diff" function to compare databases by showing their differences visually in Visual Paradigm was identified, which was not freely available. Also, from the documentation, it appeared that Visual Paradigm does not facilitate the type of calculations necessary for this research (Visual Paradigm, 2009). Finally, it was decided to calculate the metrics (i.e., count of the elements) from the instructor's model by hand. Even though it was time-consuming, it was faster than developing an application to automate the calculation of metrics.

The readability metric (M9) was calculated using the Flesch-Kincaid Grade Level. Flesch Kincaid Grade Level (FKGL) assess the readability of a text based on two factors: sentence length and word length, as seen in (1).

$$FKGL = 0.39 \left(\frac{\text{total words}}{\text{total sentences}} \right) + 11.8 \left(\frac{\text{total syllables}}{\text{total words}} \right) - 15.59 \quad (1)$$

A freely available online readability checker (<https://readabilityformulas.com/free-readability-formula-tests.php>) was used in calculating the Flesch-Kincaid Grade Level. The Flesch-Kincaid Grade Level is equivalent to the United States (US) grade level of education. It shows the required education to be able to understand a text. A text with a comparatively high score in Flesch-Kincaid Grade Level, meaning the text is less readable and difficult to understand. Table 5.3 provides an interpretation of the Flesch-Kincaid Grade Level along with the New Zealand (NZ) equivalent of the US grade level of education.

Table 5.3 Interpretation of Flesch-Kincaid Grade Level.

Flesch-Kincaid Grade Level	US education level	Comprehension	NZ equivalent
5.0-5.9	5 th Grade	Very easy to read	Year 6
6.0-6.9	6 th Grade	Easy to read	Year 7
7.0-7.9	7 th Grade	Fairly easy to read	Year 8
8.0-9.9	8 th & 9 th Grade	Conversational English	Year 9 & 10
10.0-12.9	10 th , 11 th & 12 th Grade	Fairly difficult to read	Year 11, 12 & 13
13.0-15.9	College	Difficult to read	University
16.0-17.9	College Graduate	Very difficult to read	University Graduate
18.0+	Professional	Extremely difficult to read	Professional

Note. Adapted from Flesch, R. (1979). How to write plain English: a book for consumers and lawyers.

<https://pages.stern.nyu.edu/~wstarbuc/Writing/Flesch.htm>

In order to evaluate the selected metrics, a correlation analysis was conducted between the metrics and the difficulty of a task. The metrics were calculated using the instructor's model answer for a selected set of novice modelling tasks from the assessments, and the student's performance and the instructors' perceived difficulty were used as separate measurements of task difficulty. In other words, the correlation between student performance and metrics and instructor difficulty rating and metrics were tested independently. The metrics that had a stronger correlation with difficulty were considered effective to be used in predicting the difficulty of novice database modelling tasks.

5.7 Data Collection Method

As discussed in section 3.4, the data for this analysis was collected from the consented students enrolled in INFS601 in two phases across two different cohorts during the 2nd semester of 2020 and the first semester of 2021. There was a total of 88 students who consented to participate, and a total of 17 questions were analysed. The final marks of the students who consented to provide their data were

compared to the marks distribution of the entire cohort to ensure that the sample adequately represents the overall cohort.

The questions analysed were from three different assessments conducted during the semesters: midterm assessment, normalisation and logical databases design assessment, and database project. The two online quizzes given to the students in both semesters were excluded from the analysis since all the questions assessed the students' knowledge of SQL. Similarly, from the three considered assessments, the questions not related to database modelling were excluded, and 17 out of 26 were used in the analysis (See Table 5.4). Please see Appendix J for all the questions used in identifying the predictors of novice database modelling task difficulty.

Table 5.4 Questions selected for analysis.

Semester	Assessment	Total number of questions	Included	Excluded
Semester 2, 2020	Midterm assessment	5	4	1
	Normalisation and logical databases design assessment	2	2	0
	Database project	6	3	3
		13	9	4
Semester 1, 2021	Midterm assessment	5	4	1
	Normalisation and logical databases design assessment	2	2	0
	Database project	6	2	4
		13	8	5

Once the students' solutions for the tasks were collected, their marks were checked to decide how to determine the task difficulty. Although fully correct answers were initially considered, two other measurements were also considered to calculate the difficulty of the tasks based on the student's performance using a percentage of top quartile answers and a percentage of answers having a grade A.

From the identified measures, the percentage of fully correct answers was later omitted due to the abstract nature of database modelling tasks, which made it rare for students to receive full marks. Instead, the percentage of answers in the top quartile was deemed useful to measure the difficulty because, most of the time, a developed modelling solution can be relatively correct even if it doesn't receive full marks because the solutions can vary depending on the student's perception. Hence, the focus shifted to considering the proportion of responses that demonstrated a higher level of proficiency. Further, the difficulty of the tasks was also measured via a subjective judgement of the subject experts (e.g. lecturers and teaching assistants) based on how difficult or easy they found the tasks. This was measured using a five-point Likert scale (1 very easy → 5 very difficult), a more perception-based subjective measurement, while the percentage of answers in the top quartile was a performance-based measure.

Several statistical analysis techniques were utilised in developing this difficulty framework. Correlations tests were conducted to measure the strength of the identified difficulty predictors. A Principal Component Analysis (PCA) was used as a factor reduction technique to determine a difficulty predictor model. Further, stepwise multiple regression was used in an attempt to develop a hierarchical structure to comprehend correlative and causative relationships between different types of database modelling tasks. The statistical analysis mentioned here was performed using the IBM SPSS statistical software (IBM Corp., 2021).

5.8 Data Analysis and Results

The database modelling tasks were categorised under different types of tasks such as *requirement translation*, *comprehension*, *model transformation* (*transform models from one form to another*) and *model creation*.

Table 5.5 provides the metrics calculated for the instructor's model along with the participant's performance for each question measured using the percentage of answers in the top quartile. Therefore, question 8 was the easiest one with 80% of difficulty, a comprehension task where the students were asked to identify the mistakes of a given ER model, while question 4 was the most difficult with 25% of difficulty, which was a model development task given during the first assessment the students had.

Table 5.5 Complexity metrics for instructors' model and difficulty of questions.

	Q 1	Q 2	Q 3	Q4	Q5	Q6	Q 7	Q8	Q 9	Q10	Q 11	Q 12	Q13	Q14	Q15	Q 16	Q17
Question type	Model creation (Develop ERD)						Comprehension		Requirements Translation		Model transformation						
											Relations to ERD					ERD to Relations	
N	75	77	74	8	9	10	77	10	73	9	75	73	8	9	9	70	10
Difficulty (%)	33.33	28.57	31.08	25	33.33	30	71.43	80	45.21	44.44	34.67	34.25	37.5	55.56	44.44	27.14	30
No of Entities (NE)	8	7	8	8	7	10	5	5	8	10	4	4	4	4	4	8	5
No of Attributes (NA)	36	42	50	36	31	58	10	10	NA	NA	15	12	16	12	12	18	18
No of Composite Attributes (NCA)	10	9	9	10	6	9	0	0	NA	NA	2	0	0	0	2	0	0
No of Multi-valued Attributes (NMVA)	0	1	0	0	0	0	0	0	NA	NA	0	0	0	0	0	1	1
No of Relationships (NR)	5	7	8	5	4	11	4	4	8	11	2	2	2	2	2	4	4
No of ISA Relationships (NISAR)	0	0	2	0	0	0	4	4	NA	NA	0	0	0	0	0	0	0
No of one-to-many Relationships (N1:MR)	2	6	8	2	2	8	1	1	8	8	1	1	1	1	1	3	3
No of many-to-many relationships (NM:NR)	3	1	0	3	2	3	3	3	0	3	1	1	1	1	1	1	1
Readability metric	7.83	5.95	11.05	7.83	11.16	11.22	7.32	7.32	11.05	11.22	NA	NA	NA	NA	NA	NA	NA

5.8.1 Correlation of metrics vs task difficulty (performance-based difficulty)

The following hypothesis was used to evaluate the statistical test conducted to identify the impact of the database metrics on the task difficulty:

H_0 : There is no significant correlation between the proposed metrics and the performance – based task difficulty.

H_1 : There is a significant correlation between the proposed metrics and the performance – based task difficulty.

Before the correlation analysis, a Kolmogorov–Smirnov test was conducted, which indicated that data distributions of data were not normal. Hence, a non-parametric test statistic (i.e., Spearman’s correlation coefficient) was used to conduct the analysis. Spearman’s correlation coefficient (with a significance level set at $\alpha = 0.05$) was used to calculate the significance of the correlation of each metric to the task difficulty (See Table 5.6). As per Schober and Schwarte (2018), Spearman’s correlation is interpreted as being very weak ($0 < r \leq 0.19$), weak ($0.20 < r \leq 0.39$), moderate ($0.4 < r \leq 0.59$), strong ($0.6 < r \leq 0.79$) and very strong ($0.8 < r \leq 1$) based on the value of correlation.

Table 5.6 The correlations between performance-based difficulty and metrics.

Metrics	Spearman’s Correlation (two-tailed)	
	r_s	p-value
Number of Entities (NE)	- 0.418	0.095
Number of Attributes (NA)	- 0.810**	0.000
Number of Composite Attributes (NCA)	- 0.526*	0.044
Number of Multi-valued Attributes (NMVA)	- 0.560*	0.030
Number of Relationships (NR)	- 0.306	0.232
Number of ISA Relationships (NISAR)	0.466	0.080
Number of 1:N Relationships (N1:MR)	- 0.472	0.056
Number of M:N Relationships (NM:NR)	0.051	0.845
Readability metric - Flesch Kincaid grade level	- 0.083	0.820
** Correlation is significant at the 0.01 level (2-tailed)		
* Correlation is significant at the 0.05 level (2-tailed)		

Based on the data in Table 5.6 for NA, NCA, and NMVA metrics $p\text{-value} < \alpha$ value, the null hypothesis stated earlier can be rejected, indicating that these three metrics have a correlation to the performance-based task difficulty. However, only the NA metric indicated a very strong correlation ($r_s = -0.810$, p (two-tailed) = 0.000) to the difficulty of the task, while the NCA and NMVA attributes were moderately correlated to the difficulty of the task. As expected, all these correlations were negative, which indicates that, for example, the more attributes there are in a model, the lower the students’ performance may be, and the more difficult the question may be. Surprisingly, none of the other metrics indicated a significant correlation with task difficulty. Specifically, metrics like the NE are presumed to increase the structural

difficulty of modelling related tasks, nor the readability metric, where novice modellers often find it difficult when the case studies are too complex to understand.

These findings somewhat contradict the findings of a similar study conducted by Genero et al. (2008) to evaluate a set of complexity metrics that significantly correlates with the understandability effectiveness (measured using the number of fully correct answers) of database modelling tasks. The analysis conducted here was inspired by Genero's work, considering that it's one of the few, if not the only, that has empirically evaluated the metrics related to the database comprehension tasks. In their study, the authors have mainly looked into understanding what factors drive the maintainability of conceptual data models, intending to improve the conceptual modelling process. Initially, they introduced twelve metrics that can be used to measure the ER models however, after conducting a couple of pilot studies, six metrics (NE, NA, NR, NM:NR, N1:NR and NBinaryR) out of the proposed twelve have been utilised in the final study. The participants of this study were 17 professors and 11 PhD students at the Department of Information Systems and Technologies at the University of Castilla-La Mancha, Spain, where they were provided with eight different ER models to answer comprehension-level questions by reading the models. They have further used ER models from different domains and assessed the participant's familiarity with the domains to be moderate.

Compared to Genero's study, the major difference that can be seen in this study was that it focused on the novices, where the participants were first-year undergraduate students, and the majority started learning about database models for the first time. Further, Genero's study had a selected set of ER models that demonstrated a good spread of metric values that had statistically significant correlations with performance based on their pilot studies. In contrast, the questions that were analysed in this study were limited to the assessment tasks provided as a part of the participant's coursework, where the author couldn't control the questions given.

Also, in this study, five out of six metrics suggested by Genero were adopted, excluding the NBinaryR as it duplicates the NR metric. Along with that, the NCA and the NMVA were included since it might be more meaningful to further divide the NA into different types, given that our participants were novices as opposed to the expert participants in their study. Additionally, a readability metric was calculated using the FKGL to measure the linguistic complexity of the provided case studies for the tasks.

Further, in measuring the performance, Genero has used the following measures:

- Understandability Time (UT): The time needed to understand an ER diagram (expressed in minutes)
- Understandability Effectiveness (UEffec): The number of correct answers

- Understandability Efficiency (UEffic): The number of correct answers divided by UT
- Subjective Understandability: Rating of the difficulty of the questions using a five-point Likert scale by the participants

In contrast, this study utilised the effectiveness measure using the percentage of the top quartile of correct answers and subjective difficulty using a similar rating process done by the subject experts (by the course leader and the teaching assistant). However, collecting the other measures like time and efficiency was not possible due to the limitations of collecting data that occurred as a part of the coursework.

The authors of Genero's study have found that the number of relationships and the number of one-to-many relationships were the only metrics that had a significant correlation with the understandability effectiveness and subjective understandability. Contradictory, in this study, none of the metrics related to relationships indicated any significant correlation with difficulty measured via the top percentage quartile of correct answers. Instead, significant correlations were found in the metrics related to attributes. This could be due to several reasons, mainly since Genero's study was focused on comprehension-level database modelling tasks while this study has focused on different types of modelling tasks. Another major reason for the contradicting results could be the difference in the participants where this study is related to novice database modellers compared to the database modelling experts in Genero's study. Further, the ER models used in Genero's study were far more complicated than the ones the participants of this study attempted.

However, surprisingly, in both studies, a remarkable result was that the number of entities that can be deemed as one of the main structural metrics to measure the complexity of a database modelling task was not significantly correlated with the difficulty. Hence, there was no evidence to claim that the number of entities in a conceptual model can impact the difficulty of a database modelling task.

Further, similar types of modelling tasks were grouped to identify the correlation between the metrics and the difficulty in establishing whether there could be different findings compared to the earlier outcomes (See Table 5.7). The question types *Comprehension* and *Requirement translation* did not correlate with task difficulty, hence the correlation analysis of only *Model creation* and *Model transformation* type tasks are presented in Table 5.7 and Table 5.8. Despite grouping similar types of database modelling tasks, no complexity metrics were significant to the task difficulty apart from *Model transformation* tasks. In this type of task, the NE, the NMVA, the NR and the N1:MR indicated a stronger negative correlation with task difficulty. This partially emphasises the findings of Genero related to the relationship metrics. Also, the NISAR and the NM:NR didn't indicate any correlation since the metric values were constant for all the questions, hence without any variance. This, again, can

be an indication of the similarity of the questions provided to the novices. Further, the readability metric measured using the FKGL didn't indicate a significant correlation for *Model creation* tasks despite those tasks requiring the students to develop a model based on the provided case study.

Table 5.7 The correlations between performance-based difficulty and metrics for different types of tasks.

Question Type	Spearman's Correlation (two-tailed)			
	Model creation		Model transformation	
Metrics	r_s	p-value	r_s	p-value
Number of Entities (NE)	-0.141	0.790	-0.802*	0.030
Number of Attributes (NA)	-0.338	0.512	-0.730	0.063
Number of Composite Attributes (NCA)	-0.329	0.525	0.316	0.490
Number of Multi-valued Attributes (NMVA)	-0.399	0.434	-0.791*	0.034
Number of Relationships (NR)	-0.338	0.512	-0.791*	0.034
Number of ISA Relationships (NISAR)	0.133	0.802	-	-
Number of one-to-many Relationships (N1:MR)	-0.188	0.722	-0.791*	0.034
Number of many-to-many Relationships (NM:NR)	-0.092	0.862	-	-
Readability metric - Flesch Kincaid grade level	0.368	0.473	NA	NA
* Correlation is significant at the 0.05 level (2-tailed)				

5.8.2 Correlation of metrics vs task difficulty (perception-based difficulty)

Further, perception-based difficulty was measured via the judgement of the subject experts about how easy or difficult they found the given tasks would be for novice modellers. In carrying this out, the subject experts were asked to rate the difficulty of the tasks using a five-point Likert scale from very easy, quite easy, neither difficult nor easy, a bit difficult to very difficult. The following hypothesis was used to evaluate the statistical test conducted to identify the impact of the database metrics on the task difficulty:

H_0 : There is no significant correlation between the proposed metrics and the perception – based task difficulty.

H_1 : There is a significant correlation between the proposed metrics and the perception – based task difficulty.

Cohen's kappa (Landis & Koch, 1977) was used to determine whether the two raters agreed about the difficulty level of the database modelling tasks used to assess novice database modellers. Cohen's kappa statistic for inter-rater reliability (Altman, 1999) showed there was no agreement between the two raters ($\kappa = 0.511$, $p=0.004 > 0.001$). After a discussion between the two raters, they managed to agree about their difference in the difficulty ratings.

Kendall's Tau-b correlation coefficient analysis was used to identify the impact of the metrics on the subjective difficulty considering the tied ranks in the data. A mean difficulty value was calculated based on the ratings of the two raters to conduct this. Table 5.8 provides the summary of the analysis based

on the type of tasks. In contrast to the earlier discussed correlation analysis with performance-based difficulty, only *Model creation* type tasks indicated a significant correlation between database metrics and the performance-based difficulty of the tasks. The NA, the NR and the N1:MR indicated a very strong correlation to subjective difficulty at a 0.01 significance level, indicating a correlation between the metrics and the perception-based difficulty. The findings here are similar to the findings of Genero's study, where they indicated the number of relationships and the number of one-to-many relationships impact the subjective perception of the understandability of the ER diagrams. Apart from those two common metrics, this study identified that the NA can also impact the difficulty of *Model creation* tasks.

Table 5.8 The correlations between perception-based difficulty and metrics for similar types of tasks.

Question Type	Kendall's Tau-b Correlation (two-tailed)			
	Model creation		Model transformation	
Metrics	r_s	p-value	r_s	p-value
Number of Entities (NE)	-0.341	0.381	-0.403	0.287
Number of Attributes (NA)	-0.849*	0.024	0.065	0.858
Number of Composite Attributes (NCA)	0.087	0.827	0.169	0.666
Number of Multi-valued Attributes (NMVA)	-0.387	0.351	-0.338	0.388
Number of Relationships (NR)	-0.849*	0.024	-0.338	0.388
Number of ISA Relationships (NISAR)	-0.387	0.351	-	-
Number of one-to-many Relationships (N1:MR)	-0.783*	0.049	-0.338	0.388
Number of many-to-many Relationships (NM:NR)	0.250	0.519	-	-
Readability metric - Flesch Kincaid grade level	0.077	0.838	NA	NA
** Correlation is significant at the 0.01 level (2-tailed)				
* Correlation is significant at the 0.05 level (2-tailed)				

At the beginning of this study, the main aim was to develop a task difficulty framework based on both performance-based and perception-based difficulty measures. Based on similar work done in novice programming, the speculation was that integrated measurements might provide a more reliable identification of the difficulty level of the developed modelling tasks. However, the findings of the analysis indicated that predicting the difficulty of database modelling tasks using metrics might not be useful. This led to the belief that the best option would be to rely on the experience of the subject experts to classify the difficulty of the database modelling tasks subjectively. The guidelines and vignettes discussed in Chapter 4 were used to classify the questions according to the cognitive levels based on the revised Bloom's taxonomy's cognitive dimension. Then Kendall's Tau-b correlation coefficient analysis was conducted to identify the impact of the subjective ratings on the difficulty of database modelling tasks calculated based on both performance-based and perception-based measurements. Surprisingly, the cognitive levels were not correlated with the difficulty of the database modelling tasks. Figure 5.1 provides a linear graph that further illustrates the relationship between the cognitive complexity and the difficulty of the questions.

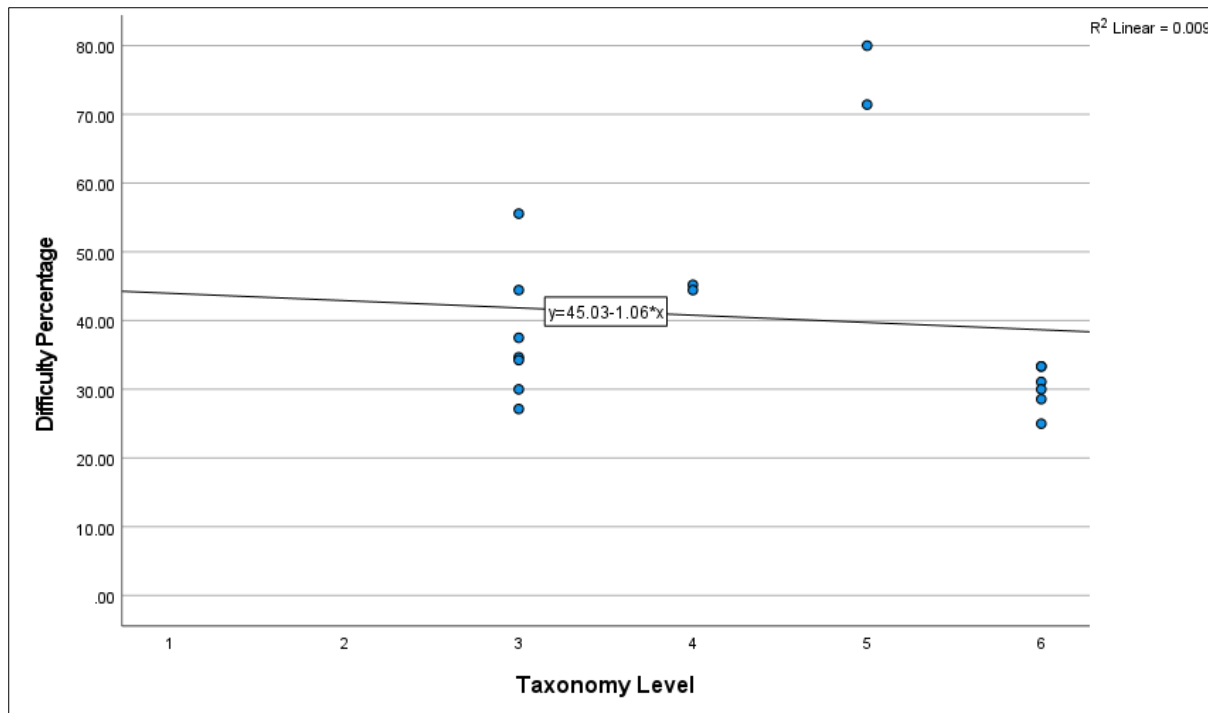


Figure 5.1 Relationship between the question difficulty (y-axis) and cognitive complexity (x-axis) of the question.

Note. In x-axis the taxonomy levels are as follows, 1=Remember, 2= Understand, 3=Apply, 4=Analyse, 5=Evaluate, 6=Create

Despite the surprising outcome of identifying the metrics, a further attempt was made to build a predictive model of intercorrelated variables that can be used to measure the difficulty of a database modelling task. Principal Component Analysis (PCA), a variable reduction technique, was used to reduce the number of metrics to identify meaningful factors that can explain the difficulty of a database modelling task. Further, feature selection using Information Gain (IG), which is one of the popular data mining methods used to minimise the dimensions and the computational complexity of a dataset, was also considered for this purpose. However, none of the approaches was deemed useful, at least in the context of this study and data. Details of the conducted PCA can be found in Appendix E.

5.8.3 Learning hierarchy

One initial hypothesis in this research was that there would be a hierarchical structure in the mastery of different types of novice database modelling tasks. However, upon analysing the data, this hierarchy was not observed. This could be possibly due to the nature of analysed modelling tasks, which may not have effectively targeted the specific aspects of the hierarchical structure, or it is possible that a clear hierarchy does not exist. Nonetheless, it is more likely that there is indeed a hierarchical structure and knowledge involved in mastering different types of tasks. The conducted analysis is presented in

Appendix K. However, it is acknowledged that further work is needed in the future to gain a comprehensive understanding of the learning hierarchy and its implications on the development of database modelling tasks.

5.9 Summary

The research presented in this chapter focused on an attempt to identify metrics that could predict the difficulty of novice database modelling tasks to develop a predictive model of intercorrelated variables. Four different types of database modelling questions were analysed in order to investigate the relationship between database metrics and the difficulty of database modelling tasks. The actual difficulty of each task was determined using two measurements: 1. Performance-based measure by calculating the percentage of top quartile answers and 2. Perception-based measure by using the ratings of difficulty by subject experts. Furthermore, an analysis was conducted to identify a possible hierarchical structure for mastering different types of modelling tasks from novices.

Through the conducted correlation analysis, it was not possible to conclusively confirm a correlation between the database metrics and the difficulty of novice database modelling tasks. A correlation analysis was not performed by SPSS on both *Requirements translation* and *Comprehension* tasks due to the fewer number of tasks and their similarity. Relating to the *Model creation* tasks, only NA, NR, and N1:MR metrics impacted the perception-based difficulty, but no metric indicated a correlation to the performance-based difficulty. In *Model transformation* tasks, NE, NMVA, and N1:MR metrics indicated a correlation with the performance-based difficulty of a task, while no metrics indicated any correlations with the perception-based difficulty. Overall, the results of this analysis indicated that most of the identified metrics were not strongly correlated with the difficulty of the database modelling tasks. Due to this, developing the predictor model was unsuccessful, along with developing the hypothesised learning hierarchy. The findings related to metrics indicated that the best option to identify the complexity of database modelling tasks would be to rely on cognitive complexity instead of database metrics. However, contradicting the belief, the applied cognitive levels also didn't indicate any correlation with the difficulty of the database modelling tasks, which was a surprising finding.

The main limitation of this analysis was the context of the collected data and not having enough tasks to calculate the metrics in each type of task. As per the ethics requirement, it was required to gain informed consent, and the first cohort's recruitment was during the first wave of COVID-19, which had a low uptake, but the second cohort had a much better uptake, with recruitment being in-person. However, the expected outcomes were not shown mainly due to the low number of questions and also gaps in the skills based on Bloom's cognitive levels. Hence, this warrants further investigation.

The findings of this analysis had a significant impact on the direction of the study since its primary objective was to identify a predictor to determine the difficulty of database modelling tasks, which was to be used to develop a typology of tasks with increasing levels of difficulty to be utilised in the research instrument to conduct the think-aloud studies. However, given the issues with COVID-19 and the continuous lockdowns in New Zealand, a decision was taken to consider alternative options to develop the research instrument.

The next chapter focuses on the design of a set of database modelling tasks utilised in the think-aloud sessions.

Chapter 6 Development of Think-aloud Modelling Tasks

6.1 Introduction

Chapter 4 and Chapter 5 discussed the attempt to develop a framework for describing novice database modelling tasks, and Chapter 5 looked at measures for predicting task difficulty with the intention of using those measurements as predictors for task difficulty in developing a set of tasks to collect the think-aloud data. Even though these attempted frameworks didn't provide the anticipated outcome, this chapter explains how the tasks were developed in an alternative manner. All these tasks were designed within the context and pedagogical constraints of AUT's novice database design courses (INFS601 and COMP508). Hence, this chapter begins with details of the context of the study in terms of: course delivery; teaching approaches; and course content; and how the curriculum changes outside of the control of the researcher affected the research instrument and the data collection.

This chapter then goes on to discuss the task design process. This process includes the choice of concepts covered by the tasks and how the tasks were developed. Apart from the central tasks, several redirected tasks were developed as a part of a scale of interventions to be used when the participants struggled while attempting the main task (See section 7.4 for the interviewer intervention protocol). These tasks were presented in a particular sequence to trigger a reorganisation of the participant's knowledge structure. Each task in the sequence is linked to the one before it, and/or it may reference some of the participant's prior knowledge to facilitate learning or development as a new knowledge structure. Finally, the chapter provides an overview of the tasks to be used in the think-alouds.

6.2 Novice Database Design Courses at AUT

As indicated in section 3.4, the participants for this research were the students enrolled in the introductory-level database course offered at AUT. The teaching and pedagogical approaches used in both courses are discussed in the following sections.

6.2.1 INFS601 teaching approach

Both the INFS601 and COMP508 courses assume no prior knowledge of database design. Both courses were taught over 12 weeks with a two-hour lecture followed by a two-hour tutorial per week⁴. During the lectures, the lecturer introduced the key concepts (See Table 6.1) using a slide deck, and problem-based learning was applied to discuss and review exercises with students during the lecture class. The weekly tutorials were used to allow the students to practice more on similar questions, reinforcing what they had learnt during the lectures. The lectures and labs were conducted in person however, there was a separate online lecture session since there were students attending online who were affected by COVID-19 at the time. A separate online tutorial session was also conducted, particularly for the students who stayed overseas and could not enter the country due to restrictions imposed by the pandemic.

The Visual paradigm tool using crow's foot notation was used for modelling purposes, while Oracle SQL developer was used as the DBMS. The lecturer also followed a problem-based learning pedagogy during the in-person lecture classes. Once the concepts were taught, the lecturer used several small problem cases to work alongside the students to make them familiar with the application of the taught content. Participatory teaching methods were emphasised in lectures and tutorials, where several weekly activities were provided to the students, which were discussed and reviewed afterwards as a group. As mentioned earlier, this course is mainly structured around relational database models, a common approach used in many introductory-level database courses. In recent work, the authors highlighted the central role of the relational model in courses like this (Goldweber et al., 2022). Their case study at the University of Minnesota describes a course similar to the one at AUT.

⁴ See Appendix G for the learning outcomes of the courses.

Table 6.1 Main topics and key concepts covered in the INFS601. The abbreviations are as follows: MA – Midterm Assessment; N&LDD – Normalisation and LDD Assessment; DBP – Database Project.

	Topic	Key concepts covered	Tasks in assessments
1	Foundations of Database Management	Database use, importance, file-based and database systems; DBMSs and functions; Types of databases	
2	Structured Language	Query Introduction to SQL; Basic SQL statements; Operators –restricting and sorting data; SQL joins; SQL functions; Table creation, alteration and constraints; Data definition language; Data manipulation language	MA-Q5: Construct SQL queries DBP-Q4: Database implementation based on the relational schema DBP-Q5: Populate data DBP-Q6: Construct SQL queries Quiz 1 and 2
3	Data Models	Types of data models; Three-level architecture; ERDs and main components –entities, attributes, relationships; Crow’s foot notation; Develop an ERD –case study analysis, business rules, Cardinalities; Relationship strength (i.e., weak and strong) and degree (i.e., unary, binary, ternary); Advanced data modelling –EER diagrams	MA-Q1: Identify errors in an EER model MA-Q4: Create an ERD N&LDD-Q1: Create an ERD DBP-Q1&2: Develop business rules and construct a logical ERD based on the case study
4	Database Design	DBLC; Database design strategies; Transforming a conceptual model into relations; Candidate, primary and foreign keys; Mapping rules	MA-Q2: Transform ERD into a logical model DBP-Q3: Develop the relational schema
5	Normalisation	Normalisation and its role in the database design process; Normal forms; Data anomalies; Steps in normalisation; Functional dependencies and keys;	MA-Q3 N&LDD-Q2 DBP-Q3: Validate the schema by ensuring the relations are in 3NF
6	Transaction Management and Concurrency Control	Database transactions and properties; Role of concurrency control in database integrity; Locking methods; Optimistic methods for concurrency control; Database recovery management	

INFS601 starts with teaching the students the basics of database systems, followed by conceptual data modelling (ER modelling), where the students are focused on understanding the requirements and business rules. Next, the students learn about the logical database model, which is about creating well-structured relations based on the requirements identified and presented in the conceptual model (i.e., mapping the ER diagram into relational schema). Despite logical database model results in well-structured relations, there might still be existing anomalies among them. Hence, students are introduced to normalisation, a process that can be used to remove anomalies from a given set of relations (Hoffer et al., 2016). Finally, the students learnt to implement the developed model using a relational database management system, Oracle.

The students were assessed using two main assessment components: coursework and database project. The coursework consisted of marks for two quizzes (6% each), normalisation and logical database design assessment (8%), and midterm assessment (30%). The students were asked to complete their weekly formative lab exercises, but no marks were offered for the completion of the lab exercises. And the database project contributed 50% of the overall mark. Despite conducting the lectures in person, the university was following the assessment plan developed for the COVID-19 lockdown. Due to that, all the assessments were take-home assignments apart from the quizzes. However, the midterm assessment was a time-controlled assessment released at a certain time, where the students were given five hours to complete and submit their work. The midterm assessment and normalisation and logical database design assessment consisted of questions related to conceptual and logical database models and normalisation. Additionally, the midterm assessment consisted of questions to construct SQL queries based on a provided database design. The database project allowed the students to work on a rather complex case study in which they were required to follow the DBLC by identifying the business rules, developing a conceptual model, and transforming it into a well-structured logical model where the students were asked to use normalisation to validate the developed logical model by demonstrating each of the relations is in 3NF by displaying the functional dependencies between attributes in each relation. Then, finally, develop a database based on the logical model and create SQL queries that support the data retrieval requirements of the business described in the case study.

6.2.2 COMP508 teaching approach

As mentioned earlier, the main change that happened between the two semesters was the replacement of INFS601 with the COMP508 course due to the curriculum change. Most of the taught content remained the same however, the new course was a level 5 course equivalent to 1st year instead of the previous level 6 course which was offered in the 2nd year. But, both courses were offered as introductory-level database systems courses since the majority of the enrolled students were novices to databases. If there were any difference, it would have been only the fact that the students enrolled for

INFS601 were more versed in computing since they were in their 2nd year. Content wise, in this course, the materials related to EER Modelling were removed and instead provided more time on teaching Entity Relationship modelling since the cohort was 1st year students (See Appendix G for the expected learning outcomes from the students). Further, in the logical model, the students were asked to produce a logical ERD instead of a relational schema. This was mainly done to simplify the process and also due to the fact that logical ERD using crow's foot notation was an easier approach to represent the information (e.g., cardinality, participation) to develop the relational database more intuitively for novices in the diagrammatic version.

The staffing resources for the course were two senior lecturers and four teaching assistants. The course consisted of twelve weekly (i) two-hour lectures and (ii) two-hour laboratory sessions. But the delivery mode was also changed almost around the mid-semester from in-person to entirely online due to the highest level of restricted lockdown that occurred because of the COVID-19 pandemic. A flipped classroom approach was used to deliver the course, where resources were uploaded beforehand for the students to review. During the face-to-face sessions, the time was used to emphasise the key learning points and undertake problem-based learning activities like INFS601. Further, several self-directed learning resources were provided to the students, such as pre-recorded videos, readings, and practice quizzes. The same approach was continued after the lockdown for the remainder of the semester, transitioning into an online mode.

During the semester, the students were assessed using two assessment components similar to INFS601. In the coursework component, however, the normalisation and logical databases design assessment was replaced with two SQL quizzes, each contributing 10% of the overall mark. Otherwise, the assessment components remained the same, where 30% and 50% were allocated to the midterm assessment and database project. The midterm assessment remained the same as in INFS601, which was a time-controlled take-home assignment, but in this, there were only four questions instead of five (i.e., both the comprehension and model transformation questions were removed and replaced by a question to write business rules from a provided ERD). The database project remained the same however, instead of asking to validate the developed relational model using normalisation, the students were only asked to produce the logical ERD by adhering to the mapping rules required while transforming the conceptual model into a logical model to ensure the final ERD is in 3NF (e.g., transforming composite attributes, resolving multi-valued attributes).

6.3 Task Design Process

Chapter 4 discussed the attempt to develop a framework for describing novice database modelling tasks, and Chapter 5 looked at measures for predicting task difficulty. The idea was to use this framework of

bloom level and database metrics to inform the design of a series of database modelling tasks for the think-aloud study. However, the proposed database metrics did not show any conclusive correlation with task difficulty measured via the performance of students on task. As a result, the more subjective measurement (the cognitive levels of the revised Bloom's taxonomy) was explored as a measure of difficulty, but again, no correlation was observed between the Bloom level and student performance on a task. Due to the inability to establish a predictive task difficulty framework within the constraints of the research, a pragmatic decision was made to use well-known questions that have been tested in classes and written in textbooks. Developing an entirely new framework to predict task difficulty would have required significant time and resources that were not available for this study. While developing a predictive task difficulty framework could have provided additional insights, it was not the central focus or core objective of the research. Instead, the study focused on evaluating participant performance and understanding their approaches to the given tasks within the given time frame. Hence, it was decided to develop the tasks to cover database development activities taught to the students under the planning, analysis and design phases of the system development life cycle based on the textbook used in the INFS601/COMP508 course (Hoffer et al., 2016). The first three tasks were related to developing a conceptual model using a top-down approach, starting from analysing the case study, identifying the business rules, and developing the conceptual model. Then, Task 4 was transforming the conceptual model into the logical database model. These four tasks were continuous tasks built on top of each other based on the same case study.

The tasks discussed earlier were related to a typical top-down database design approach taught to novices. The participants were required to work with a case study to identify business rules, develop a conceptual model and then transform it into a logical model that can be implemented into a physical database, and normalisation concepts were used to verify the quality of the relational scheme to ensure they were in 3NF. However, there are situations where there is a need to reverse-engineer older database systems where many of the tables can be subjected to redundancies and anomalies, which could be converted into a well-structured set of relations using normalisation (Hoffer et al., 2016). In a more recent study, it has also been suggested that a bottom-up approach would be a more relevant approach to teaching:

Particularly, in recent years there is a shift from self-describing data-driven systems to a problem-driven paradigm that is the bottom-up approach where data exists before being designed. (Ishaq et al., 2022, p.2)

Hence, there was an additional set of tasks to understand how novices approach normalisation problems. The purpose of including normalisation was to teach students the technique of developing well-structured relations.

So far, in the existing literature, the researchers have only utilised a case study to develop a conceptual model for understanding novice database modellers' cognitive processes (Batra & Antony, 1994; Rosenthal et al., 2020, 2022). None of the researchers had attempted to scaffold the tasks or segment the case studies into logically smaller parts given to their research participants. However, in this study, it was decided to scaffold the task of developing a database design into a sequence of tasks across multiple sessions. The tasks were designed to have participants already apply the concepts and processes they had learnt in the course to a new problem context. Further, the first four tasks based on the same case study were developed so that the novices must transfer their knowledge gained by completing the prior task(s) to complete the current task. Each task requires furthering the knowledge gained on the prior task and also verification of the knowledge gained in the prior task. But, the task related to normalisation was based on a different problem scenario, as it takes a bottom-up approach to developing a logical ERD instead of a top-down approach.

Database design is typically considered more complicated, particularly for novices, due to its abstract nature and the higher number of interrelated concepts required to solve a task that could incur a higher cognitive load on novices when attempting database modelling (Amin et al., 2019; Katz, 2018b; Katz & Shmallo, 2016). Therefore, previous research on reducing learners' cognitive load in database courses was explored to construct the research instrument. Researchers have investigated how to reduce learners' cognitive load in database-related courses using worked examples (Knorr, 2020), progressive practice – by having the learners begin working on simple versions of the task and gradually providing them with the complex versions until the task is accomplished (Bunch, 2009). Mason et al.(2016) discuss redesigning instructional materials by applying CLT to resequencing the content, modifications to the learning materials format and changes to assessment activities. Later studies used keyword glossaries to remove the extraneous cognitive load on the examination papers (Mason & Seton, 2021; Mason & Seton, 2020). This work in the database domain inspired some of the decisions made in developing the research instrument to reduce the overall cognitive level of the participants.

The number of interrelated concepts in database modelling increases the element interactivity required to complete a task that can impose a high intrinsic load on the working memory. Particularly, novice database modellers find it difficult to think like expert database modellers (Watson, 2006), where their limited prior knowledge can strain the working memory since they still need to engage more in the cognitive processes necessary for constructing mental models and schema development compared to the experts. Hence, different pedagogical recommendations introduced in CLT, often referred to as *effects*, were considered in designing this research instrument of the think-alouds with the intention of reducing the overall cognitive load (i.e., reducing the intrinsic and extraneous loads) that can release the cognitive resources in principle and then be allocated to increase the germane load where learning happens. The used effects are as follows:

i) Segmentation strategy

Segmentation strategy, a technique belonging to the *sub-goal effect* (Catrambone, 1998), was applied to break down both the development of the relational database model and normalisation problems by breaking the complex task into a manageable series of smaller tasks where one task scaffolded the participants into the next task. The intention of the segmentation was to reduce the intrinsic load by presenting the task in a way where the participants could focus on one piece of information or one step at a time, reducing the cognitive load associated with processing and integrating multiple elements simultaneously. The concept of segmentation was initially discussed by Miller (1956), who stated that short-term memory could only hold 5-9 chunks of information (seven plus or minus two), where a chunk is any meaningful unit. The concept of chunking and the limited short-term memory capacity associated with it can be considered the basis for Sweller's work.

ii) Intervention model and redirected tasks

Further, as indicated in section 7.3.1, several interventions were developed to assist the participants during their think-alouds. The interventions were developed primarily with the intention of reducing the intrinsic cognitive load by providing support, guidance, and instructional strategies to help them manage and gradually reduce the intrinsic cognitive load. It can be argued that by providing these interventions, the participants can focus their cognitive resources on the relevant information and skills they need to develop, thus reducing the overall cognitive load associated with the intrinsic complexity of the task. As a part of the interventions, several *redirected tasks* (i.e., smaller and simpler tasks focused on one or few concepts at a time) were introduced in the research instrument, which was used when the participants couldn't progress at all when they were overwhelmed with complex tasks, possibly due to cognitive overload. The idea of redirected tasks was inspired by the *isolated-interacting element effect* (Sweller, 2010). When a task is redirected, it means that the learners' attention is shifted from the original task to a different, possibly simpler or more focused task with the intention of reducing the extraneous cognitive load associated with the original task. Redirected tasks may make learning easier and minimise intrinsic cognitive load brought on by the task's inherent complexity. Further, in this study, the redirected tasks were developed for two main purposes: to build knowledge so the participants can solve more complex problems and for the researcher to understand the source of the difficulty faced by the participants.

iii) Structured presentation of the case study

Further, the case study for the database model development problem was presented in sections according to the main entities using a structured worksheet rather than the conventional approach of presenting the entire study all at once. This design decision to use segmentation in presenting the case study was influenced by the teaching staff's experience and empirical research that supports the observation that students struggle when faced with analysing an entire case study at once. The research conducted by Bock & Yager (2005) acts as a justification for this decision. Their study found that using a structured worksheet significantly improved students' learning outcomes in identifying constructs such as entities, attributes, and relationships compared to the control group who didn't use the worksheet when developing a conceptual model. Following a similar approach to Bock and Yager's research, the case study in this study was divided into several sections. The main goal of this approach was to reduce extraneous cognitive load that could incur when a case study is presented in a traditional manner, which may overwhelm novices with a large amount of information and data to process all at once.

iv) Sequenced sessions

Holding the sequenced sessions allowed students to focus on smaller pieces of the task, arguably reducing the extraneous cognitive load. On the other hand, the time between working on the problem and recalling what was done in a prior session probably added more cognitive load. However, working on a problem over time reflects a student's typical learning experience where multiple courses are taken and tasks are done over time.

Further, in deciding the problem description for the database model development problem, a familiar domain was adopted as a deliberate decision to reduce the cognitive load that an unfamiliar domain may bring. These strategies were applied to reduce the overall cognitive load imposed on the participants while attempting tasks. However, the application of CLT was not evaluated since there was no control group to measure the effects. It was not the main research direction of this study, yet a limitation acknowledged by the author. However, the tasks depict realistic scenarios that students might face in larger assignments.

Another objective of the redirected tasks, apart from possibly reducing the cognitive load, was to use it as a tool to discover what aspects of a central task caused the participant difficulty or stopped them from progressing by isolating the concepts as much as possible. The redirected tasks were designed to relate to the least possible number of concepts possible to lower the complexity associated with tasks that involve many interrelated concepts. This idea was adopted to estimate the difficulty of a task, at least to a certain extent, based on the number of related concepts. Such an approach has been suggested for

the assessment of novices, where the authors suggest that “complex assessments can be decomposed into atomic elements that can be assessed independently” (Luxton-Reilly et al., 2018, p. 47). Thus, here in this doctoral research instrument design, an assumption is made that the fewer concepts a task involves, the less complex the task is and the easier the task will be to solve successfully.

When designing think-aloud sessions, aligning them with the timing of teaching relevant concepts and skills necessary for completing the task is crucial. This enables students to have a solid understanding of the topic before engaging in the think-aloud process. Additionally, providing students with ample opportunities to practice and apply their learning through task-based activities to consolidate their learning is important. Hence, the think-aloud sessions were arranged when the participants were introduced to the concepts and skills required to complete the tasks through lectures and laboratory classes.

6.3.1 Task overview

As indicated in Table 6.2, the participants were assigned two main tasks during the study. The first main task was the database model development problem, which involved a top-down approach. In this task, the participants were provided with a case study description and were required to analyse it in order to develop a logical database model. The second task was the normalisation problem, which involved a bottom-up approach. For this task, the participants were given an existing set of data and were required to analyse it to identify any data redundancies and anomalies. They had to apply the steps in normalisation to transform the data into a normalised set of tables, reducing data redundancy and minimising data anomalies. This task required the participants to follow a bottom-up approach by starting with the existing data and gradually building up the normalised tables. By assigning these two tasks with different approaches, the study aimed to assess the participant’s skills and understanding of both top-down and bottom-up approaches to data modelling for a comprehensive evaluation of their abilities to analyse requirements, design logical models, and apply normalisation principles effectively.

In designing the research instrument, the database model development problem was presented as four scaffolded tasks built on top of each other. The idea was to reduce the potential overall cognitive load of database development tasks due to their abstract nature and the multiple interrelated concepts required to solve them. Further, the provided case study description was segmented into manageable sections so that the research instrument would naturally allow the participants to focus on one section of the case study at a time, which might reduce the intrinsic cognitive load that could arise from dealing with multiple information simultaneously. Each section provided participants with a dedicated space to list down the identified entities, attributes, and relationships with the intention of allowing them to focus

on one section at a time, aiding their understanding and analysis of the case study. An extracted section from the case study is given in Figure 6.1.

T1. Consider the given below section from the case study and identify all the likely entities, attributes, and relationships.

Trainees: It is necessary to store details about all the trainees who participate in the courses. Each course participant is identified by their trainee ID, and this is stored along with their name (composed of first name and last name), date of birth, gender, address (composed of street number, suburb, city, postcode), and contact number. Along with these details, the previously acquired qualifications of the trainees should be stored in the database. Since most of the time these trainees are referred to the course by their employers, it is also necessary to store the details of the employers including the name of the company, address, and contact number. Each employer can be uniquely identified using the employerID. An employer can refer many employees to a training course, and a trainee can be only referred to a course via one employer. However, there can be few trainees who are not referred by an employer.

Figure 6.1 Extract from the full case study given in Appendix H.

Further, in this study, the case study description was intentionally selected to ensure that the participants were familiar with the domain. By selecting a familiar domain, the researcher sought to minimise the extraneous cognitive load associated with comprehending and interpreting the case study. Similarly, the normalisation problem was also provided in three tasks that were scaffolded into each other. While attempting the main tasks, if the participants struggled, several interventions were provided to help them progressively (i.e., clarification, prompt, hint, redirected task, exact solution). These interventions were provided in a progressive manner, starting from a simple clarification, prompt, hint, and redirected task to provide the exact solution.

In designing the research instrument, a key assumption was made regarding the time required to complete a single task, estimating it to be within a 30 to 45-minute session. Hence, each task was allocated to be completed within a 60-minute session, apart from Tasks 1 and 2 in the database model development problem, where both were allocated to a single 60-minute session. This assumption was an informed judgment made by the researcher based on prior experience teaching laboratory classes for these courses. This assumption was crucial for planning the research instrument and ensuring that sessions were designed to facilitate the smooth execution of the study. However, it is important to acknowledge that individual differences among the participants and unforeseen circumstances can affect the number of sessions conducted.

Eight main and eleven redirected database modelling tasks were created for this research. These tasks were mainly classified according to the concepts required to solve the problem. Based on the analysis presented in Chapter 5 it was evident that using the cognitive levels of the revised Bloom's taxonomy was not a reliable method to predict task difficulty, at least within the context of the collected data. It did not work explicitly for working out the difficulty of specific database modelling tasks, but Bloom's has been shown to be a useful framework for looking at the cognitive complexity of similar modelling tasks in OO analysis. Hence, it was decided to retain Bloom's cognitive levels as a holistic way of looking at the cognitive complexity of various types of tasks when developing the research instrument.

An overview of these tasks is provided in Table 6.2.

Table 6.2 Overview of the tasks.

		Conceptual knowledge													
	Task	Redirected task/s	Business rules (syntax)	Entity	Attribute	Relationship	Cardinality	Entity types	Attribute types	Relationship types	Mapping rules	Functional dependency	Data anomalies	Normal forms	Cognitive difficulty
Database model development problem	T1			✓	✓	✓									An
		T1.1		✓	✓										U
	T2		✓	✓		✓	✓								Ap
		T2.1		✓			✓								U
	T3			✓	✓	✓	✓	✓	✓	✓					C
		T3.1				✓	✓								U
		T3.2						✓	✓						Ap
		T3.3							✓						An
		T3.4		✓	✓	✓	✓								Ap
	T4						✓	✓	✓	✓	✓				Ap
		T4.1									✓				R
		T4.2						✓		✓	✓				U
Normalisation problem	T1								✓			✓	✓	✓	Ap
	T2											✓	✓	✓	Ap
		T2.1												✓	R
		T2.2										✓			U
		T2.3							✓			✓		✓	U
	T3		✓	✓	✓	✓									Ap

Note. Where R – Remember, U – Understand, Ap – Apply, An – Analyse, E – Evaluate, C – Create in the cognitive difficulty column in the order of lowest cognitive level to the highest one.

The relationship between the first four tasks under the database model development problem is illustrated in Figure 6.2. As per Figure 6.2, there were intermediate interventions like prompts, clarifications and hints before providing the participants with the redirected tasks. However, the prompts and clarifications were given based on the participants' progress, which was not directly relevant to the tasks. Hence, the diagram only provides some examples of the potential hints used during the sessions. Further, the implied steps required to complete each task are provided in Table 6.3. A detailed discussion of each task and relevant redirected tasks are presented in the following sections, 6.3.2 to 6.3.5.

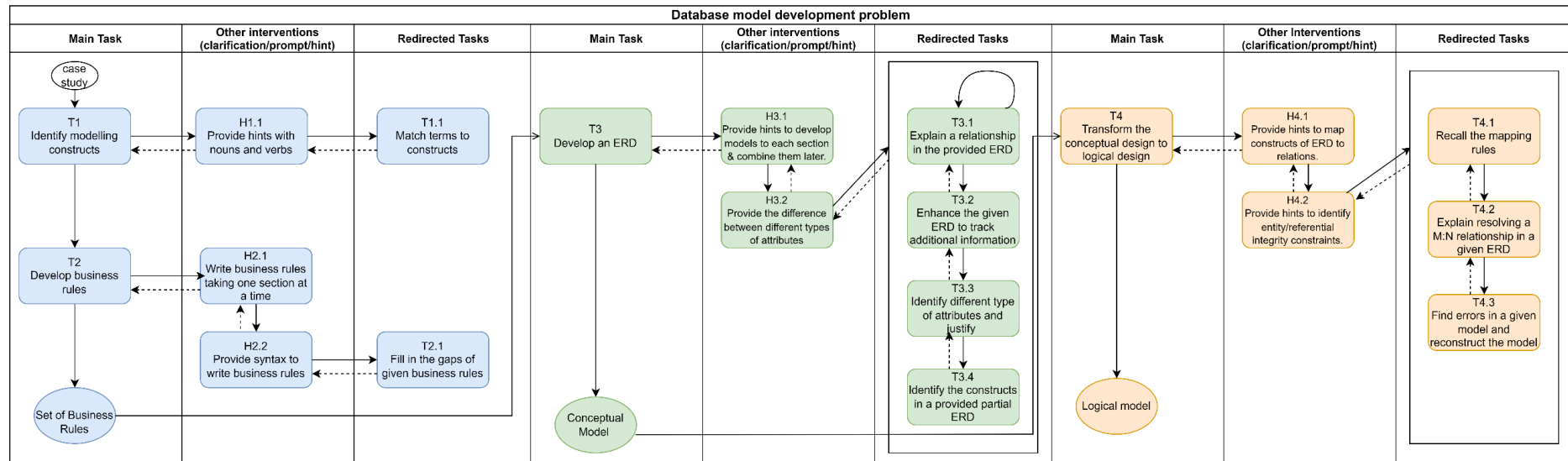


Figure 6.2 The flow of the tasks in the database model development problem.

Note. Tasks and the outcome of a task required for another task are coloured according to the session.

Arrow $\longrightarrow$ represents the forward question flow where the participants are redirected to scaffoldings with less difficulty.

Arrow $\dashrightarrow$ represents the backward question flow where the participants are redirected back to the main task to continue.

Table 6.3 Steps required to solve the tasks.

	Task	Steps required to solve the task
Database model development problem	T1	• Conduct a noun-verb analysis. • Relate the knowledge of entities, attributes, and relationships to nouns and verbs. • Eliminate the unnecessary noun and verbs with proper reasoning (e.g., duplicate terms, vague, ambiguous, out of scope).
	T2	• Translate the identified constructs into business rules. • Recall the syntax of writing the business rules. • Knowledge of cardinalities and participation to complete the business rules.
	T3	• Identify the main entities and relationships from the developed business rules. • Develop the initial ERD. • Identify attributes and primary keys that adequately describe the entities. • Revise and review the developed ERD.
	T4	• Transform the conceptual ERD into a logical design (e.g., relational schema, logical ERD). • Identify single-valued and multivalued attributes, strong vs weak entity types, and different relationship types (e.g., one-to-many, many-to-many relationships). • Resolve multivalued/composite attributes, one-to-many and many-to-many relationships when transforming the design into a logical one by applying the taught mapping rules. • Review and ensure the final design is in 3NF (e.g., make sure the multivalued attributes or many-to-many relationships are resolved and no redundant attributes).
Normalisation problem	T1	• Determine the primary key of the provided table and its current normal form. • Prompts to recall that normalisation is based on the analysis of functional dependencies and the need to determine the primary key prior to that. • Identify the existing current normal form.
	T2	• Based on T1, illustrate the functional dependencies using a dependency diagram. • Decompose the relation based on the taught process by removing functional dependencies simultaneously.
	T3	• Transform the identified set of 3NF relations into a Logical ERD.

6.3.2 Task 1 and Task 2 – Analysis of the case study

At the first think-aloud session, the participants were provided with a case study (See Appendix H), and they were given the following tasks:

Task 1 – Consider the ‘Trainees’ section given below from the case study and identify all the likely entities, attributes, and relationships. Follow the same process for sections ‘Courses’ and ‘Instructors’.

Task 2 – Using Task 1 as a guide, develop a set of business rules for the above case study. Each relationship must be described by two rules, one in each direction.

The case study was at a similar level of linguistic and conceptual complexity as the ones the students were given during their take-home assignments but was from a different problem domain, which was not used during the classes. As noted in the thesis, the participants did not undergo examinations due to the online delivery that happened because of the lockdowns. Hence, the specific examination questions were not considered for the instrument design. However, the questions given at an examination would be similar, albeit may be on a smaller and with less conceptual complexity.

In Task 1, the participants were expected to identify all the likely entities, attributes, and relationships. Since the participants had attempted similar tasks before, they were already familiar with the expectations. They were aware that they needed to identify potential constructs by eliminating redundant ones or merging synonymous ones. Overall, Task 1 aimed to assess the participants' ability to analyse and differentiate between various constructs in the context of database design, which required them to operate at *Analyse* cognitive level.

On completion of Task 1, in Task 2, the participants were asked to use the entities and attributes extracted as a guide to identifying the business rules that explain the requirements to develop the database, which was an *Apply* level task. The participants who correctly completed Task 1 were only required to translate the identified constructs into a taught syntax to present the business rules. To answer this, the participants needed knowledge about the syntax of the business rules, the fact that two business rules were required to explain a relationship between two entities, and the concepts of participation and cardinality, which explain the minimum and maximum number of instances the entities participate in a relationship. The model answer for Task 2 is:

1. Each EMPLOYER may refer one or many TRAINEES.
Each TRAINEE may be referred by zero or one EMPLOYER.
2. Each COURSE may generate zero or many PROGRAMS.
Each PROGRAM must be generated by one and only one COURSE.
3. Each PROGRAM must generate one or many CLASSES.
Each CLASS must be generated by one and only one PROGRAM.
4. Each PROGRAM must register one or many TRAINEES.
Each TRAINEE must be registered to one and only one PROGRAM.
5. Each INSTRUCTOR may teach in zero or many PROGRAMS.
Each PROGRAM must be taught by one or many INSTRUCTORS.

Redirected tasks

Task 1.1

T1.1 – Match each of the terms (drawn from the description above for Trainees) given under the list of options to the correct modelling construct column of the below-given table.

Entity	Related attributes	Related entity (if any)	Relationship name

List of options: Trainee ID, Trainee fname, Trainee lname, date of birth, Employer, employerID, employer name, employer address, employer contact no, mobile phone number, office phone number, street number, suburb, city, postcode, course, course participant, refer, referred by, store

The required answer is indicated in Table 6.4.

Table 6.4 Model answer for Task 1.1.

Entity	Related attributes	Related entity (if any)
<i>Trainee</i>	<i>Trainee ID, Trainee fname, Trainee lname, date of birth, mobile phone number</i>	<i>Employer</i>
<i>Employer</i>	<i>employerID, employer name, employer address, employer contact no, street number, suburb, city, postcode, office phone number</i>	<i>Trainee</i>

This task was developed to support the participants if they were struggling with noun-verb analysis. The redirected task addressed the problem by narrowing the focus to only the trainee section of the case study, excluding the other sections. To aid participants in identifying the entities and attributes, they were provided with a list of options and asked to select the relevant ones from the given words. The intention of the task was to help participants by giving them a starting point and reducing the cognitive load associated with identifying potential constructs from scratch. By recognising that nouns could potentially represent entities or attributes, participants were encouraged to differentiate between these two categories. In addition, the participants were also prompted to fill in the last column if they observed a connection between entities. This step provided participants with a head start in identifying relationships and encouraged them to consider the connections between different entities.

Task 2.1

T2.1 – Fill in the gaps to complete the given below business rules from the ‘Trainees’ and ‘Instructors’ sections of the case study.

1. Each Employer may refer zero or more TRAINEES.
2. Each TRAINEE must be refereed by zero or one EMPLOYER.
3. Each Instructor may teach zero or more PROGRAMS.
4. Each Program must be taught by one or more INSTRUCTORS.

List of options: zero or one, Instructor, one or more, Employer, teach, Program, zero or more

Note. The answers are presented in italics.

Task 2.1 was a simple fill-in-the-blanks task again focused on certain sections of the case study only. The objective of the task was to allow the participants to deal with only a couple of concepts since the business rules were partially completed. However, additional options were provided in both tasks to ensure they were not mechanically doing the redirected tasks.

6.3.3 Task 3 – Develop a conceptual model

The purpose of Task 3 was to assess the participant’s ability to develop a conceptual model based on the business rules they derived in Task 2. In order to complete this task, the participant must possess a comprehensive understanding of the problem domain. Additionally, the business rules in Task 2 must be accurate and thorough. This is necessary to effectively translate them from their grammatical form to a conceptual model. However, the main task was guided using several sub-tasks that helped the participants to realise what was required in the model.

Task 3 – Based on the business rules developed in Task 2, construct a conceptual Entity Relationship Diagram (ERD) using the Crow’s foot model symbols.

- i. Determine the main attributes in each entity, including all primary and foreign keys.
- ii. Identify the main relationship types (connectivity) between the entities and cardinality and participation (mandatory/optional) for each relationship.
- iii. Describe any assumptions you have made while constructing the model.

In developing the conceptual model, the participants were required to refer to the identified constructs and the business rules they had developed during Tasks 1 and 2. This approach ensured that participants could use their prior work and build upon the foundational understanding they had already established

by referring back to the identified constructs in Task 1 and use them to develop the conceptual model. Similarly, the business rules in Task 2 provided additional context (i.e., cardinalities, mandatory/optional participation, etc.) for the conceptual model. Further, in attempting Task 3, the students were required to have adequate knowledge of the fundamental concepts and constructs of ER modelling, such as entity types (i.e., strong and weak entities), cardinality or relationship cardinality constraints (entity instances related – maximum and minimum cardinality/participation), the relationship types (identifying and non-identifying relationships), types of attributes (e.g., simple, composite) and primary/foreign key attributes. While Tasks 1-3 were structured and presented in steps in order to reach the correct endpoint, the participants needed to operate at Bloom's *Create* level. The model solution for Task 3 is given in Figure 6.3.

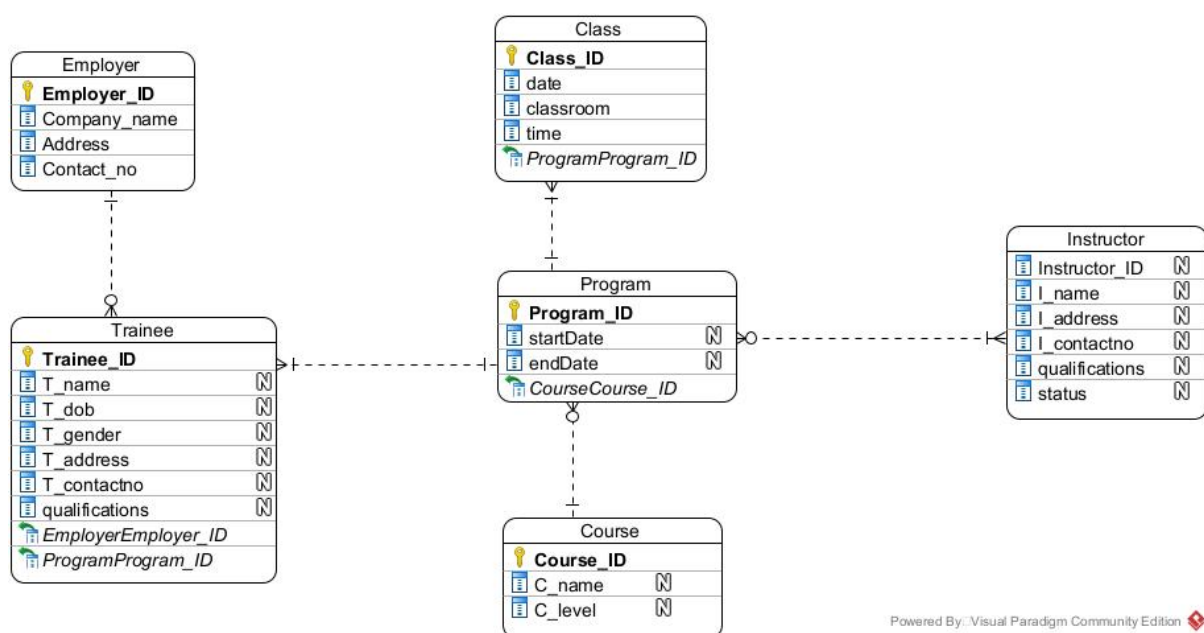


Figure 6.3 Model answer for Task 3

Redirected tasks

The ER diagram shown in Figure 6.4 and the problem text provided below, formed part of the redirected tasks Task 3.1, Task 3.2 and Task 3.3.

Production tracking is important in many manufacturing environments (e.g., the pharmaceuticals industry, children's toys, etc.). The following ER diagram captures important information in the tracking of production. Specifically, the ER diagram captures relationships between production lots (or batches), individual production units, and raw materials.

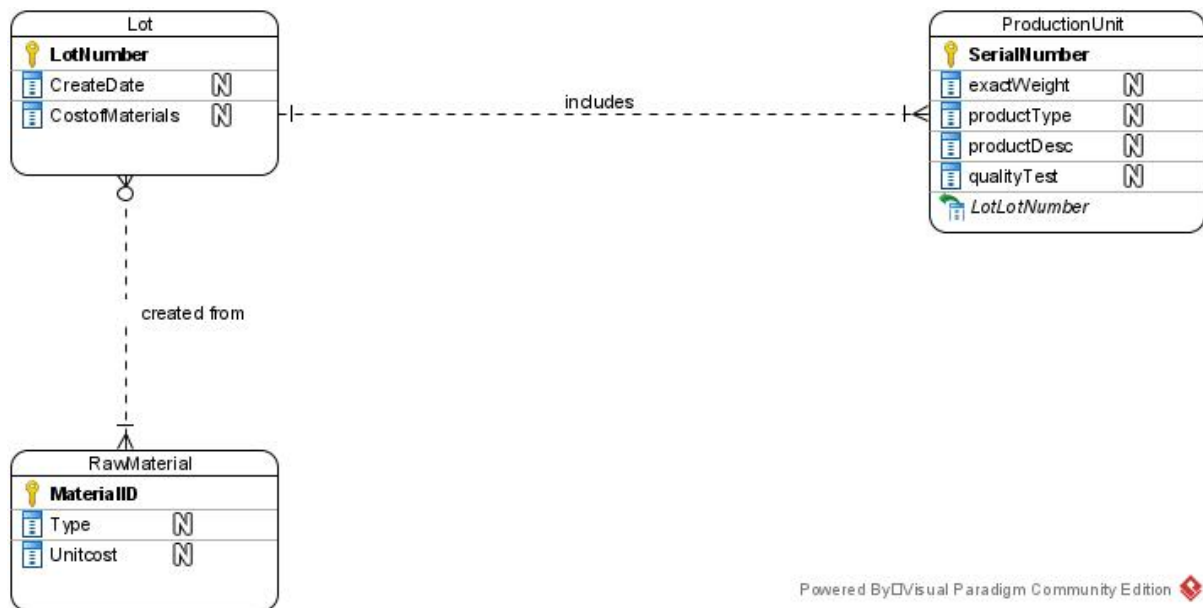


Figure 6.4 ERD related to Task 3.1, Task 3.2 and Task 3.3.

The following redirection tasks, 3.1 to 3.3, were designed to get the participants to operate at increasing levels of cognitive complexity (see Table 6.2) from *Understand*, *Apply* to *Analyse*, respectively. All of the tasks were considered to be less complex than Task 3 itself.

Task 3.1

T3.1 – The current ER diagram represents a relationship between the raw materials and production lot. Can you explain to me the relationship between these two entities in the context of the manufacturing environment?

Task 3.1 required the participants to read and interpret an ERD in order to explain the many-to-many relationship depicted in the diagram between the entities **RawMaterial** and **Lot**. A typical answer might involve the participants explaining that the raw materials can be in the system without being allocated to a specific lot when explaining the relationship between **RawMaterial** and **Lot** in the same direction. For a complete view, the participants can explain the relationship to the other direction from **Lot** to **RawMaterial**, since the relationships in an ERD can be read in both directions.

The main objective of this task was to understand whether participants could relate and explain a many-to-many relationship in a different domain, separate from the main task. The cognitive level required for this task was at the *Understand* level, which involved comprehending and explaining the concepts of relationship and cardinality. By focusing specifically on these concepts, the task was designed to simplify the cognitive demands compared to the main task. This allowed the participants to concentrate

on understanding the relationship and cardinality aspects without being overwhelmed by the interrelated concepts required in the main task.

Task 3.2

T3.2 – If the company wants to track the number of units of each raw material used in the production lot, how would you suggest doing that? Please redraw the diagram to indicate the suggested change.

Task 3.2 required the participants to use the knowledge gained in Task 3.1 to modify the provided ERD. Basically, the question hinted at the need to resolve any many-to-many relationships by applying what the students had learnt about entity types (i.e., associative entity) and attribute types (i.e., composite primary key and foreign keys), which required the participants to operate in *Apply* level. The model answer for Task 3.2 is provided in Figure 6.5.

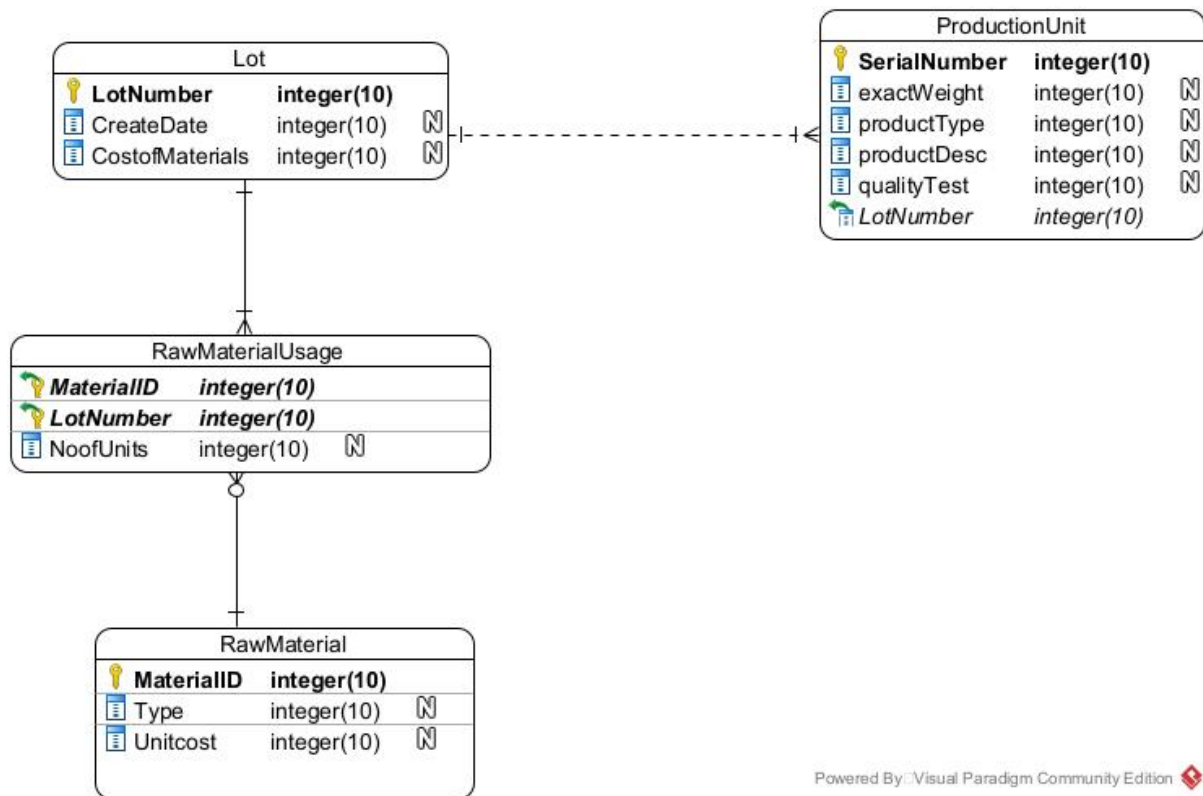


Figure 6.5 Model answer for Task 3.2.

Task 3.3

Participants were asked to identify potential attributes that could be derived and composite attributes where they only needed to focus on the conceptual knowledge of attribute types. However, the task further asked them to explain their modelling choice, making it rather difficult compared to Task 3.1 and Task 3.2, which required the participants to operate at *Analyse* level.

T3.3 – Please identify an attribute in the above ER diagram that could represent a derived attribute and a composite primary key. Explain why/how they might represent a derived attribute and a composite primary key.

The required answer from the participants was **CostofMaterials** as a derived attribute and the **MaterialID, LotNumber** (See Figure 6.5) as a composite attribute. For example, **CostofMaterials** could be computed based on the material's unit cost from **RawMaterial** and the number of units required for a lot, hence why a derived attribute.

Task 3.4

T3.4 – Below is a partial ERD (See Figure 6.6) representing the 'Trainees' section of the case study. Provide appropriate names for entities E1,E2; primary keys PK1,PK2, composites attributes A1, A2; multi valued attribute A3; attributes A4..A8; relationship R and cardinalities C1,C2.

This task was developed to help the participants if they struggled further despite the previous redirected tasks. The developed ERD was partially completed with information from the first section of the case study. The students were required to complete the partial diagram, hoping this would trigger the knowledge required to complete it based on their existing knowledge and the information gained from the previous tasks. This still required the participants to work with several concepts (i.e., entity, attribute, relationship, cardinality) however, the cognitive difficulty was much lower since they needed to apply the identified information on the partially completed diagram rather than developing it from scratch.

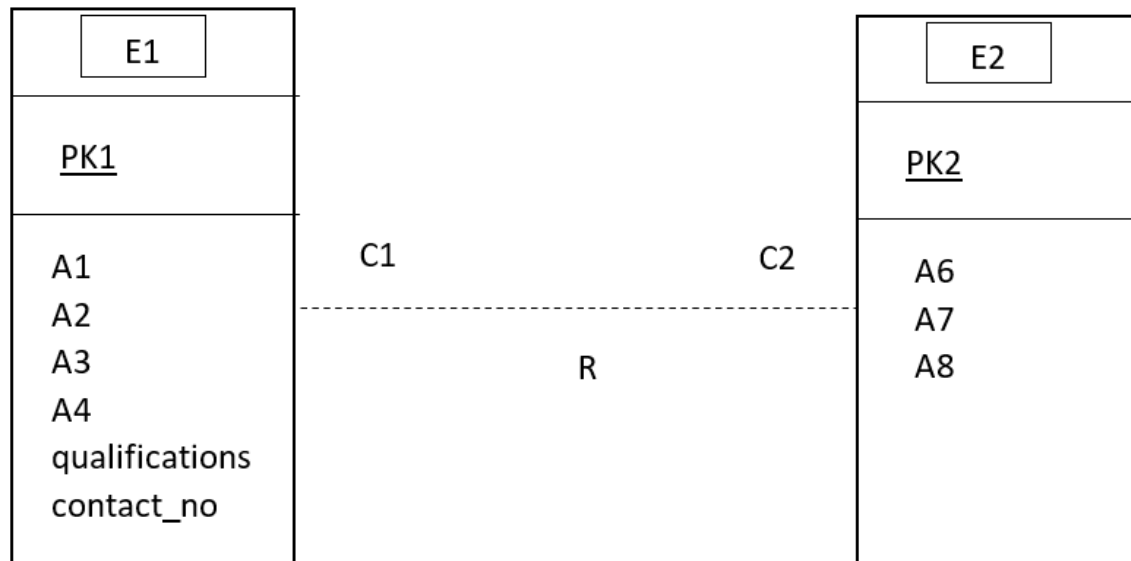


Figure 6.6 Partial ERD related to Task 3.4.

Entity E1:	<u>Trainee</u>	Entity E2:	<u>Employer</u>
Primary key PK1:	<u>TraineeID</u>	Primary key PK2:	<u>EmployerID</u>
Composite attribute A1:	<u>Trainee_name</u>	Attribute A6:	<u>Company_name</u>
Composite attribute A2:	<u>Trainee_address</u>	Attribute A7:	<u>Company_address</u>
Attribute A3:	<u>date of birth</u>	Cardinality C1:	<u>one-to-many</u>
Attribute A4:	<u>gender</u>	Cardinality C2:	<u>zero-to-one</u>
Relationship:	<u>Rrefer</u>		

6.3.4 Task 4 – Transform the conceptual model into a logical model

In this task, the participants were required to transform the developed conceptual ERD into a relational model. The primary objective of this task was to accurately capture the requirements of the system, ensuring that they were translated into a relational schema that could be implemented using a selected database model. To successfully complete this task, participants needed to analyse the conceptual ERD and map the entities, attributes, and relationships into appropriate tables, columns, and keys in the relational schema. The transformation process required participants to make design decisions, such as determining primary and foreign keys, establishing relationships through appropriate constraints, and resolving any potential data redundancies or anomalies by applying normalisation to ensure the relational schema is in the 3NF.

Task 4 – Transform the developed conceptual model into a logical design (e.g., relational schema or logical ERD). You must ensure that your final design is in the third normal form (3NF).

The participants were required to understand multiple concepts like entity types (e.g., strong and weak entities), attribute types (e.g., composite, multivalued attributes), relationship cardinality constraints (e.g., one-to-many, many-to-many relationships), relationship types and apply mapping rules (a set of rules to transform the ERD to relations) to solve this task. As mentioned in section 6.2, the required solutions from the participants were slightly different in both courses, as indicated in Figure 6.7 and Figure 6.8.

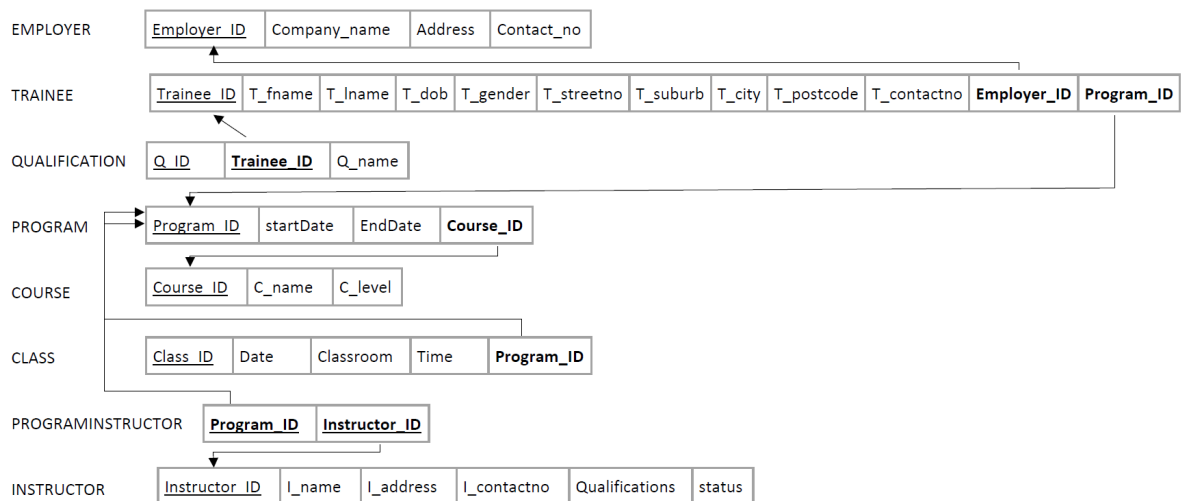


Figure 6.7 Model answer for Task 4 in INFS601.

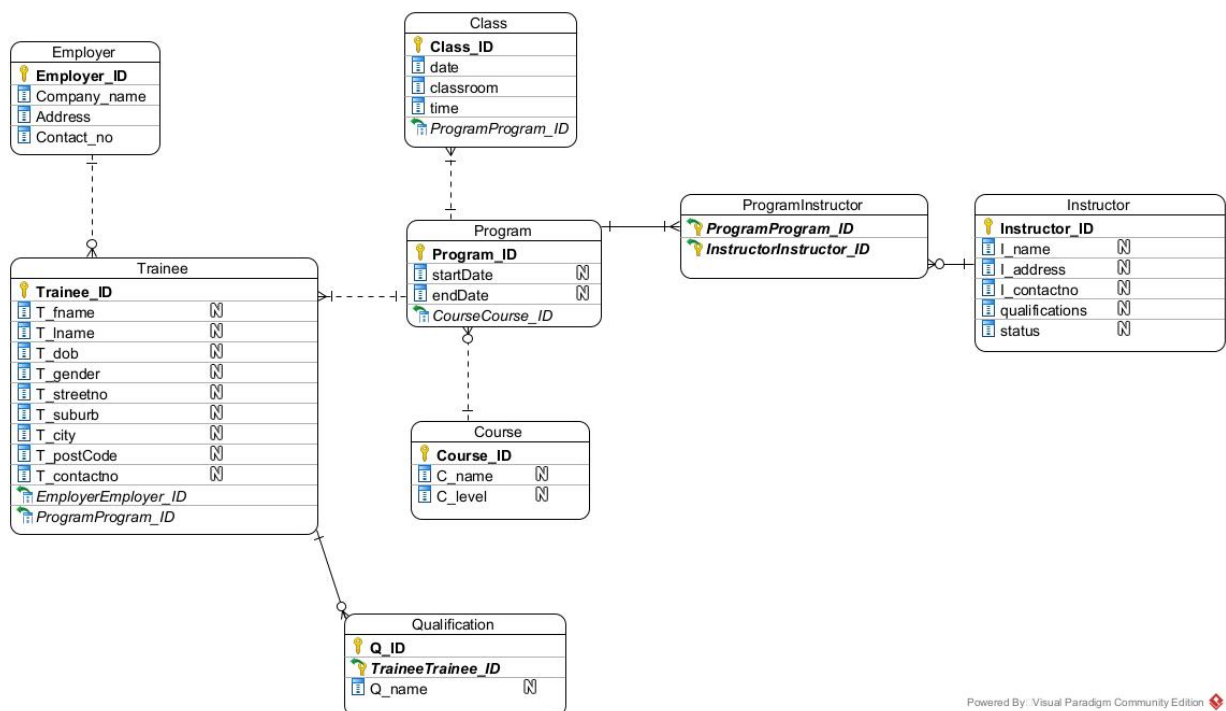


Figure 6.8 Model answer for Task 4 in COMP508.

Redirected tasks

Task 4.1

T4.1 – Can you recall the mapping rules required in transforming this ERD to a relational schema?

This task was developed to trigger the participant's cognitive schema on the mapping rules and was used if they struggled to transform the conceptual model into a logical design. Compared to the main task, the participants only needed to focus on recalling the taught mapping rules. This means this task targets the lowest cognitive level of Bloom's taxonomy, *Remember*. To solve this problem, the participants need to recall the mapping rule related to multivalued attributes – introducing a new entity/relation with the primary key of the original entity as a part of the composite primary key in the newly introduced entity/relation.

Task 4.2

T4.2 – Can you explain how to resolve the many-to-many relationship given in Figure 6.9? If you want to store the grade students receive for each class, where would you place that?

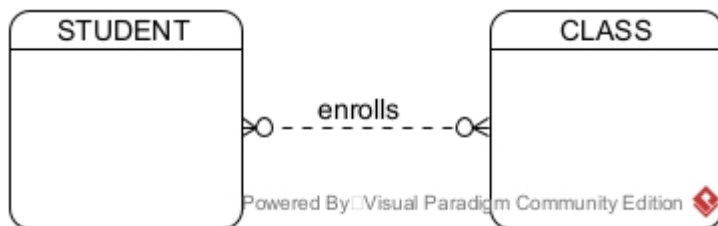


Figure 6.9 ERD for redirected task T4.2.

This task required the participants to recall the mapping rule related to many-to-many relationships and then explain how it could be resolved in a logical design. Further, they need to identify where to place the attribute grade. Since this task required the participants to recall and explain the mapping rule in relation to the partial conceptual diagram in Figure 6.9, it can be categorised under the *Understand* level in the revised Bloom's taxonomy's cognitive levels. The participants should be able to explain that a new associative entity was required between the student and class entities where the primary key is a composite one (combination of primary keys from student and class), and the grade needed to be placed in the newly created entity.

6.3.5 Normalisation problem

Normalisation was taught (in both courses, INFS601 and COMP508) based on the concepts covered in the textbook **Modern Database Management** (Hoffer et al., 2016). As normalisation is considered arguably one of the most challenging topics to teach (Amin et al., 2019) and learn (Wang et al., 2020) in database design, it was decided to provide the participants with a separate problem on normalisation. The participants were provided with one main normalisation problem presented as three tasks, which were similar to the exercises they practised in class and were structured based on the examples given in the course textbook. Figure 6.10 shows how these three tasks were delivered and scaffolded within the session.

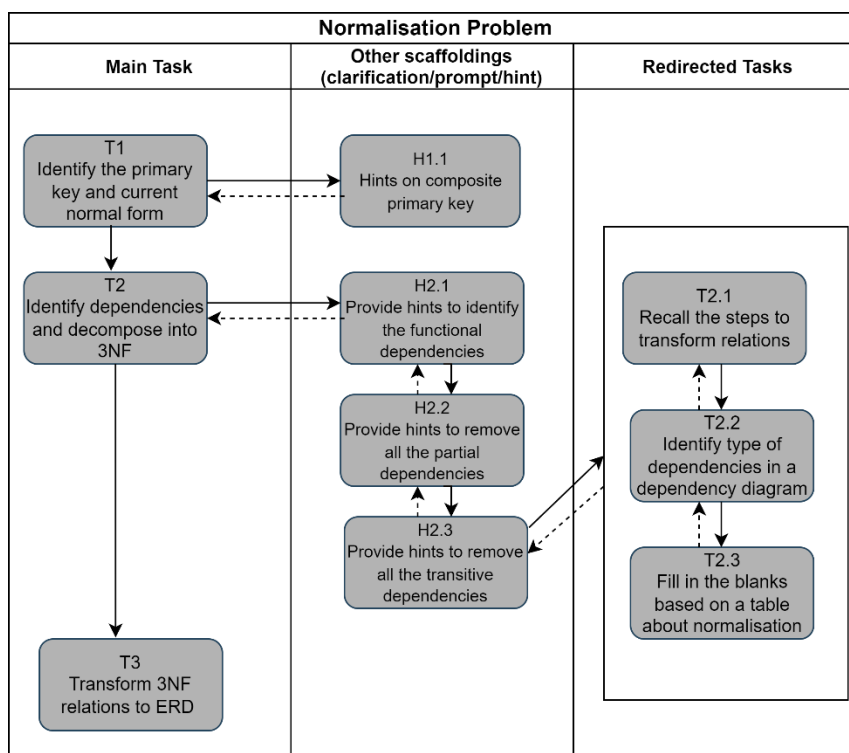


Figure 6.10 Task flow for normalisation problem.

The normalisation problem required the participants to analyse a data table (See Figure 6.11) susceptible to data redundancies and transform it into a set of 3NF relations by initially identifying the primary key attributes/s and functional dependencies of the given table. Then, they were asked to develop a logical ERD that represents the operations accurately, and the entities reflect their conformance to 3NF from the normalised relations. The participants needed to apply existing knowledge to solve the task. Further, as for the database model development problem (Tasks 1-4), the participants need to build on the knowledge they acquire from one task in the problem to the next.

S_ID	SLName	SFName	C_ID	CName	Grade	L_ID	LLName	LFName	Semester
001	Rose	Johnson	DB01	Database	A	L001	Ruger	John	Sem1
001	Rose	Johnson	GRE01	German	B	L002	Findling	Holger	Sem2
002	Anthony	Joshua	DB01	Database	A	L001	Ruger	John	Sem1
002	Anthony	Joshua	GRE01	German	B	L002	Findling	Holger	Sem2
003	May	Parker	GRE01	German	A	L002	Findling	Holger	Sem2
004	Susana	Grant	ITA01	Italian	B	L003	Fresco	Luise	Sem2

Figure 6.11 The data table provided for the normalisation problem.

Task 1 –Identify the primary key/s for the provided data table and the current normal form of the data table. Briefly justify your answer.

Task 1 is intended to prompt the participants' recall of important constraints that a relation must satisfy before doing the normalisation process: (1) there are no repeating groups in the relation (thus, there is a single fact at the intersection of each row and column), and (2) a primary key has been defined. The participants were required to analyse the data table to identify this information. To determine the current normal form, the participants needed to understand the requirements for a relation to be in a normal form from UNF (un-normalised form) to 3NF. The correct answer required from the participants was that the relation was currently in 1NF because it had no repeating groups and had a composite primary key: S_ID and CID. To answer this, the participants needed to understand several concepts, such as attribute types (i.e., composite attributes), functional dependencies, data anomalies and normal forms and apply them correctly to determine the current normal form, which requires the participants to operate at *Apply* level.

Task 2 – Identify all the dependencies using a dependency diagram and decompose the relation into a set of 3NF relations. State clearly any assumptions you make.

Normalisation Task 2 can be answered in two steps. First, the participants were required to present the identified dependencies, preferably using a dependency diagram (See Figure 6.12) distinguishing between different functional dependencies (i.e., partial, transitive, full) presented in the provided data table.

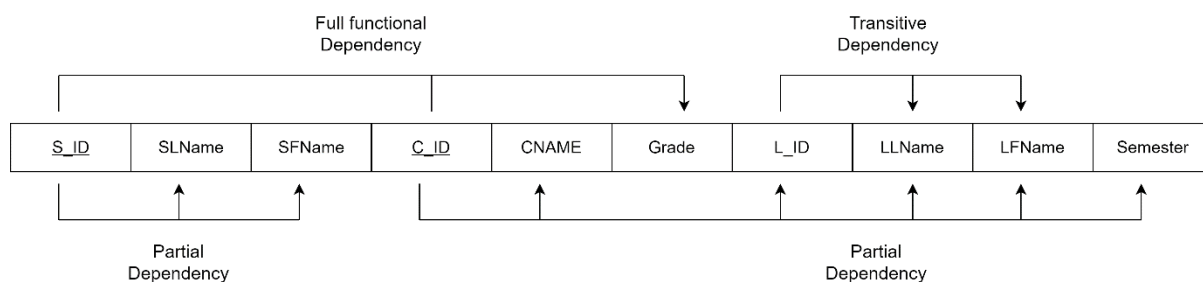


Figure 6.12 Dependency diagram illustrating different functional dependencies in the data table given in Figure 6.11.

In the next step, the participants were required to decompose the relation into a set of 3NF relations from the current 1NF based on the identified dependencies in step one. The participants needed to draw on the knowledge acquired earlier to do the decomposition. Then, they were required to apply a step-by-step approach to remove partial dependencies to move from 1NF to 2NF and remove transitive dependencies to move the relations from 2NF to 3NF. The solution for this step is provided in Figure 6.13, which can be considered a much more difficult task compared to Task 1, as the students were required to apply a taught process to develop the solution.

STUDENT: S_ID → SLName, SFName
 COURSE: C_ID → CName, semester, L_ID
 STUDENT_COURSE: S_ID, C_ID → Grade
 LECTURER: L_ID → LLName, LFName

Figure 6.13 Decomposed relations in the 3NF.

Redirected tasks

Task 2.1

T2.1 – Can you explain the steps required to transform a relation from UNF to 3NF.

This task was utilised when the participants struggled when they were unable to proceed due to not remembering the normal forms and their meanings. The objective was to trigger the participants' knowledge of normal forms which they gained during the classes. Cognitively this task can be categorised in the lowest level of difficulty, *Remember*.

Task 2.2

T2.2 – Based on the dependency diagram given below (See Figure 6.14), identify the type of dependencies given in D1, D2, and D3 and justify your answer.

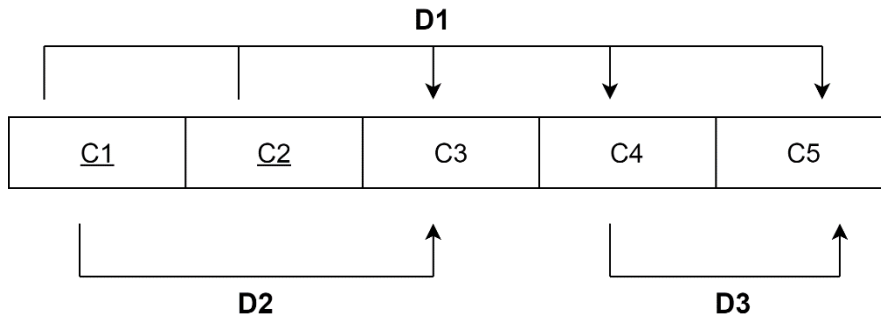


Figure 6.14 Redirected Task 2.2 where the correct answer D1= Full functional, D2 = Partial, and D3 = Transitive.

Task 2.2 required the participants to identify different types of dependencies from a simple diagram in the situation where they struggle with the concept of functional dependencies. This task required the participants to recall a familiar diagram and apply their understanding of its structure and components to identify the type of dependencies in the current diagram. Further, they only need the knowledge of normal forms to answer the redirected task in contrast to the multiple concepts required to answer the main task.

Task 2.3

T2.3 – Examine the table given below (Figure 6.15) and fill in the blanks.

1. staffNo and branchNo is the primary key of the above table.
2. Values in hrsperweek column can be determined from staffNo and branchNo.
3. There is a partial dependency between staffNo, name and position.
4. A table is in 2NF when it is in 1NF, and there are no partial dependencies.

staffNo	branchNo	branchAddress	name	position	hrsPerWeek
S4555	B002	30, Symonds st, Auckland 1010	Jane Edwards	Admin Support	8
S4555	B004	10, Queens st, Auckland 1010	Jane Edwards	Admin Support	16
S4612	B002	30, Symonds st, Auckland 1010	Anton Layman	Admin Support	12
S4612	B004	10, Queens st, Auckland 1010	Anton Layman	Admin Support	10

Figure 6.15 Data table related to Task 2.3.

Task 2.3 was developed to help the participants who struggled with reading and identifying the dependencies given in the data table. This task aimed to determine whether they could identify simpler dependencies than required by the main task from the table and fill in the blanks accordingly. Even though the participants were required to work with knowledge of composite primary keys, functional dependencies, and normal forms, the data table was easier to analyse compared to the main task, with only two potential candidates for the primary key. By filling in the blanks with simpler dependencies, participants were required to demonstrate their understanding of the relationships within the table and their ability to apply basic concepts. Hence, the participants were only required to operate on a lower cognitive level, *Understand*.

Task 3 – From your 3NF relations, draw an Entity–Relationship (ER) model and include all attributes, primary keys, foreign keys, cardinality, and participation.

This final task required the participants to develop an ER diagram based on the normalised relations in which they were required to use their knowledge of concepts such as entities, attributes, relationships, and cardinalities. Again, this was an *Apply* level task where the participants were required to develop the model based on the identified normalised relations. The model answer is provided in Figure 6.16. Further, in this task, the participants were asked to develop the ER diagram based on the crow's foot notation by identifying the appropriate participation and cardinality notations.

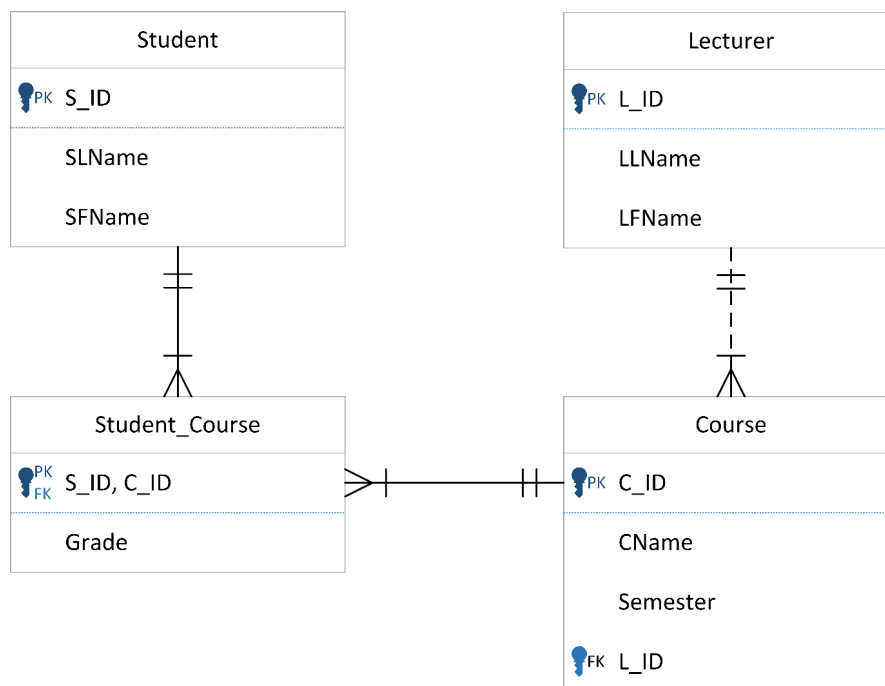


Figure 6.16 Model answer for Task 3.

6.4 Summary

This chapter has presented the novice database modelling tasks, which formed the research instrument for this research. As explained earlier, the initial plan of using a task difficulty predictor did not yield the expected results. Hence, a pragmatic decision was taken to use the well-tested and adopted questions used in the class and textbook, which cover the top-down and bottom-up approaches in database design to design the research instrument. Several tasks were developed under the database model development problem and normalisation problem, which were designed to represent the top-down and bottom-up approaches, respectively. Several precautions were implemented to reduce the overall cognitive difficulty of the tasks. One of these precautions involved the use of redirected tasks, which served multiple purposes in the research instrument. Firstly, they were designed to help participants build the necessary knowledge and skills that could be applied to solve the main tasks. Secondly, redirected tasks served as a source to identify the participants' difficulty by isolating the concepts required to answer the task. By focusing on the number of concepts, the research instrument indirectly provides insights into how novices navigate different levels of complexity in database modelling.

As discussed in the chapter, the timing of the sessions was carefully planned to align with the participants' exposure to the concepts covered in their lectures and laboratory classes. This approach aimed to ensure that the participants had sufficient familiarity with the relevant concepts before engaging in the research instrument tasks. However, it is important to acknowledge that the study was limited by the duration of a single semester. While the timing of the sessions and the opportunity for

consolidation were optimised within this timeframe, it may not have provided sufficient time for full mastery of the subject matter.

The next two chapters present the results of the think-aloud and retrospective interviews which were conducted based on this research instrument. The two chapters utilised the same data from two different perspectives to understand novice database modellers as a group and as individuals.

Chapter 7 Think-aloud: Reflexive Thematic Analysis

7.1 Introduction

This chapter presents the reflexive thematic analysis (Braun & Clarke, 2022) conducted as a part of this research, which explored the difficulties faced by novice database modellers, the mechanisms students used to solve such problems and the tactics used to overcome any difficulties they encountered. Despite several studies on students' challenges in learning database modelling (Batra & Antony, 1994; Batra, 2005, 2007; Currim et al., 2014; Hussain, 2016), there seems to be a lack of research on the common tactics that novice learners use to overcome these difficulties. Understanding tactics used by students providing effective learning interventions should be designed. It is worth noting that most of the existing studies used conceptual models created by students as their primary research data. While conceptual modelling is a fundamental aspect of database modelling, it is equally important to examine the entire process of developing a model that can be implemented in a database. Understanding how students transfer their knowledge across different steps in the modelling process is vital to gain a comprehensive understanding of the learning of novice database modellers. Hence this thematic analysis was conducted to address this gap. The following research questions are addressed to understand the difficulties faced by novices, the thinking process of the learners at various levels, and the learning activities they engage in to complete the tasks.⁵

RQ3.1: What difficulties do students face when attempting novice database modelling tasks?

RQ3.2: What common tactics do students apply when attempting novice database modelling tasks?

The chapter presents a detailed description of the participants, the think-aloud method, including the data collection and transcription protocols, and finally, a thematic analysis of the data. The researcher's reflexivity and subjectivity are also explored as an integral aspect of the interpretation of this analysis.

⁵ It should be noted that the think-aloud data collected was used in two different analyses. Firstly, using reflexive thematic analysis, common themes were extracted from the data as reported in this chapter to explore student difficulties and tactics with respect to database modelling. Secondly, the data from each participant was used to build a Learning Trajectory/Progression (LT/P) for each participant to build a picture of their learning progression through undertaking the research tasks which is presented in Chapter 8 .

7.2 Research Participants

As discussed in section 3.4, the main data collection phase was in the 2nd semester of 2021. Initially, 21 students indicated an interest in participating in this research and gave informed consent. However, only seven students ended up attending the scheduled think-aloud sessions. These seven students completed at least one think-aloud (attempting Tasks 1 and 2 given in section 6.3.2) and are henceforth referred to as participants. Of these seven participants, two dropped out from the study after the 1st think-aloud without providing a reason. For some participants, the 2nd think-aloud (related to Task 3 given in section 6.3.3) of the study was delayed by one week due to the transition from in-person to online think-aloud sessions. There was another participant who, despite the best efforts of the interviewer, found it difficult to focus on the task at hand when thinking aloud, even in the in-person setup. The interviewer decided to work with the participant despite being unable to collect useful research data. But this participant removed themselves from the study after the second think-aloud. It is likely that the shift to online think-alouds due to the nationwide COVID-19 pandemic lockdown made participation difficult for those participants who dropped out from the study. The remaining four participants completed the study and thus provided a complete set of data for the reflexive thematic analysis presented in this chapter. The analysis reported in that chapter comprises think-alouds from two cohorts and a total of six participants, including these four participants and two from the pilot study conducted during the 1st semester of 2021. As discussed earlier, the participants were recruited from two different courses. However, in both cases, the courses were the first encountered in their undergraduate degree that covered databases and database modelling. Combining these data meant that more data points for the reflexive thematic analysis were available and were considered reasonable, given that all participants were novices to database modelling. The only advantage the level 6 students may have is their maturity or emotional stability since they have already completed one year of study. Nonetheless, all the participants were fresh to database modelling. Altogether, 18 think-aloud sessions were analysed to conduct this thematic analysis.

In line with this chapter and Chapter 8 the participants are addressed using gender-neutral aliases (as given in Table 7.1) to remove any unintentional reinforcing of gender stereotypes. Hence, in these two chapters, gender-neutral pronouns (i.e., they, them, and theirs) have been used instead of gender-specific pronouns. The background of the selected participants is presented in Table 7.1.

Table 7.1 Background information of the think-aloud participants.

Participant	Enrolled course	English as a Second Language (ESL) student?	Completed the INFS600 (Data Process and Modelling) course?	Prior experience in UML modelling?	Prior experience in database modelling?
Sam	INFS601	No	Enrolled in the course on the same semester as INFS601.	Current knowledge from the INFS600 course.	None
Jackie	INFS601	Yes	No	No	As a part of a formal secondary education qualification.
Casey	COMP508	No	No	None	None
Morgan	COMP508	No	No	None	- As a part of a formal secondary education (IGCSE ICT) qualification. - Some database implementation experience in MS Access.
Eli	COMP508	No	Yes	Yes	None
Blake	COMP508	No	Enrolled in the course on the same semester as COMP508.	Current knowledge from the INFS600 course.	None

7.3 Think-aloud Data Collection Method

In this study, a data collection mechanism was needed in a way to understand the cognitive thinking of novice database modellers. Several existing mechanisms utilised in the literature, like ethnography or phenomenography, were considered. Observation methods like these can provide rich data since they involve understanding particular individuals or groups through close observation and interaction. However, it was not feasible to extensively observe the participants in their natural settings due to limitations such as the inability to follow the students' home or be continuously present when working on their modelling assessments and tasks. Ultimately, the data in this phase was collected using think-aloud interviews.

The think-aloud method involved the verbalisation of thought processes while performing a task and was initially developed to study human problem-solving strategies in psychology (Newell & Simon, 1972). The work of Ericsson and Simon (1993) on verbal protocol analysis is considered to be a good guide for researchers collecting data using the think-aloud method. The technique mainly involves recording participants as they solve a task while simultaneously speaking their thoughts out loud. According to Ericsson and Simon, the underlying principle is that verbalisations provide a sub-set of

the thoughts in working memory that occur when performing a task. This method can be adopted to determine what information is used during problem-solving and how that forms knowledge that contributes to framing a solution to a problem. This information can then be used to make inferences about how participants reasoned and made decisions during problem-solving.

One major assumption in gathering data using think-alouds is that the process does not interfere with the participant's thought process (Ericsson, 2017). However, gathering this kind of real-time data can be limited by the cognitive load imposed when trying to problem-solve and verbalise thinking simultaneously (Branch, 2000). Therefore, taking necessary precautions to minimise the cognitive effort during verbalisation is important to allow the participants to articulate their thinking process with the least additional burden. Despite this acknowledged limitation, think-aloud is generally accepted to be one of the better means of gaining insight into problem-solving processes (Ericsson, 2017; Rosenthal et al., 2020; Someren et al., 1994). Moreover, in this study, the impact of thinking while problem-solving was reduced through a practice task conducted prior to starting the first think-aloud session.

The think-aloud method is used widely in educational research to gain insight into the cognitive processes employed by students, often novices, as they complete a learning task. It is used to capture students' strategies, intentions, observations and emotions (Someren et al., 1994). Think-alouds have been used to study undergraduate computer science students in many different contexts, including database modelling and design (Batra & Davis, 1992; Rosenthal et al., 2020), software testing (Rojas et al., 2015) and novice programming (Loksa & Ko, 2016; Teague et al., 2013; J. Whalley et al., 2021; Whalley & Kasto, 2014b). The think-aloud method was chosen to collect the data in this study since it's been used successfully in CSER, and, due to its popularity, there was an established set of guidelines that could be adopted easily.

Several guidelines are provided for carrying out think-alouds successfully, ensuring the participants can verbalise their thoughts freely (Ericsson & Simon, 1993; Someren et al., 1994). These guidelines, along with a data collection procedure used by Rosenthal et al. (2020) in their recent study, were utilised in developing this study's think-aloud data collection protocol.

One of the most important facts in conducting think-aloud is to create a place where the participants feel at ease. Often, sessions can take a long time to complete, so "*the room should be quiet, a glass of water should be at hand, the chair should be comfortable*" (Someren et al., 1994, p. 41). Considering these guidelines, a suitable meeting room was allocated for conducting the sessions. The room had a large desk that provided enough room for the participants to work and for the interviewer to observe them without interrupting. The interview room was situated on a floor occupied by the post-graduate researchers where there were minimal disturbances.

In this study, consent was taken during the recruitment process. Hence no time was initially allocated for this in the data collection protocol. However, the students who wanted to join after the recruitment was completed at the beginning of the first session were asked to fill out the consent form based on the guidelines of Rosenthal et al. (2020).

Someren et al. (1994) suggest writing down the instructions beforehand and reading them to the participants instead of using phrases like: ‘tell me what you think’, which could lead to different interpretations of their own. At the same time, they suggested keeping the instructions short to avoid any unnecessary interpretations. Further, they have suggested giving participants reminders to think aloud after a predetermined time of silence. Rosenthal et al. (2020) have suggested using the precise wording “Please keep talking”, and these instructions were adapted as deemed useful in this study’s data collection protocol.

Using a warm-up task prior to the actual session was suggested in the guidelines as a way for the participants to become familiar with the verbalising process. In their study, Rosenthal et al. (2020) provided the participants with a relatively simple task (i.e., construct a conceptual data model comprising two entity types and one relationship type) compared to the main modelling task based on a different domain. This idea was adapted in developing the practice session.

7.3.1 In-person data collection protocol

This section discusses the developed data collection protocol in detail. The step-by-step protocol is presented in Figure 7.1.

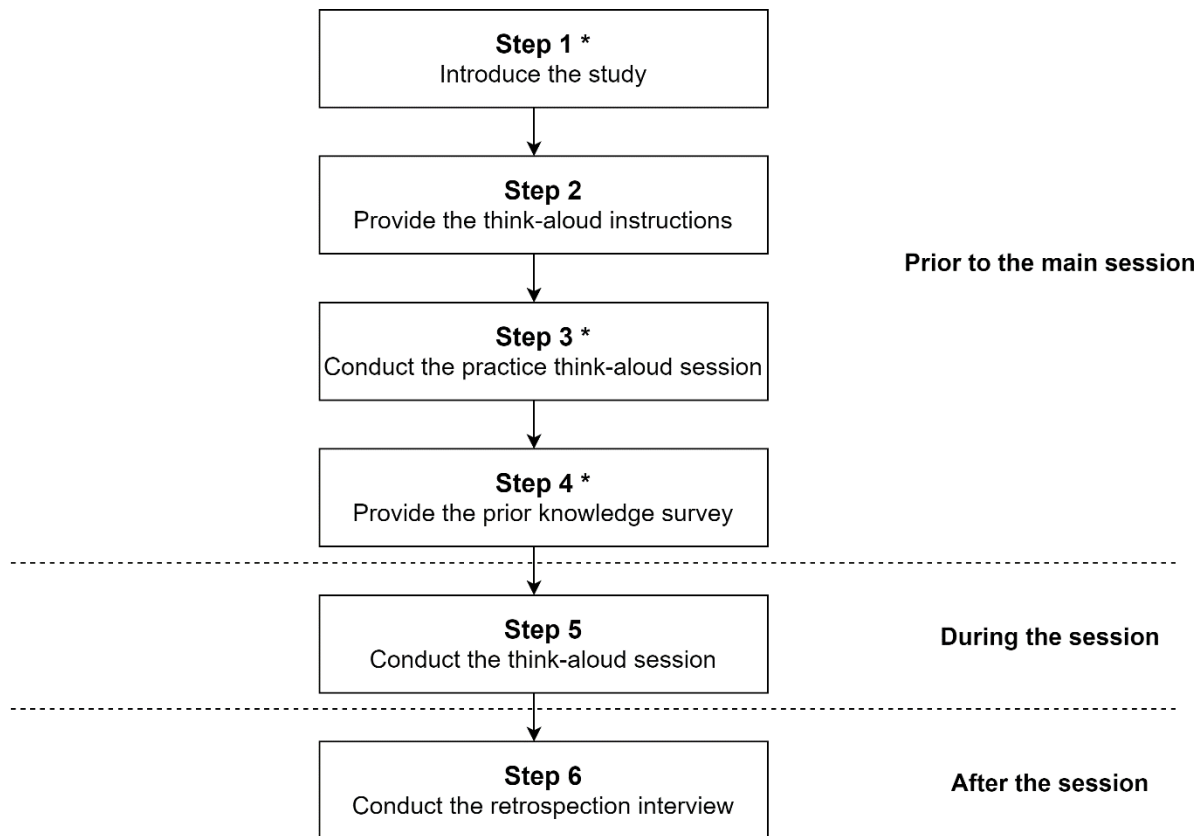


Figure 7.1 Think-aloud data collection protocol.

Note. The steps indicated in * were only followed during the first session, while other steps were followed in the same order during all the sessions.

Step 1: Introduce the study

At the beginning of the first session, building a rapport with the participants was crucial to ease them into what to expect. Hence, prior to the first session, some time was utilised to explain the purpose of the research and data confidentiality. All this information was provided to participants during the recruitment phase. However, the reminder was a good icebreaker to start off each session. The interviewer emphasised that they were only interested in their problem-solving process rather than critiquing or judging the participant's performance.

After introducing the study, their consent was double-checked. If in case they had not completed a consent form, one was provided, and the participants were given the time and instructions to complete it.

Step 2: Provide the think-aloud instructions

As the next step, the participants were provided with think-aloud instructions at the start of each session. A think-aloud protocol script was developed to assist the interviewer in conducting the sessions (see

Appendix L). Clear instructions were provided to the participants at the beginning of each session, reminding them of the importance of talking aloud while attempting the task. They were also asked if they were reading something to verbalise as they read. But this was not strictly enforced. If a participant appeared to be reading silently, they were asked to indicate what they were reading. The primary goal of these sessions was problem-solving, while verbalisation was secondary, attempting a task in the wild – as they would do for an assessment or a project. Hence, they were instructed that the interviewer would, up to a certain level, play the role of observer rather than being active. However, participants were reminded of the importance of continuing to talk while undertaking their task, and if they kept silent for more than one minute, they would be reminded to keep talking. If it appeared a participant was finding it hard to think and speak or was struggling to progress on a task, the interviewer asked specific questions intended to trigger progress or continue the think-aloud process. A specific protocol for interviewer interventions is provided in section 7.4. Further, they were informed that any other issues related to the task or the subject could be discussed during the retrospective interview upon completing the task.

Step 3: Conduct the practice think-aloud session

Several researchers who used think-aloud protocols in CSER have discussed using a warm-up session prior to the actual session, where they discussed the importance of training the participants beforehand, which leads to acquiring more quality data (Kasto, 2016; Teague, 2015). Therefore, in this research, based on the guidelines of Rosenthal et al. (2020), participants were asked to complete a simple practice exercise in the first think-aloud session to familiarise them with the protocol. In this exercise, participants were asked to identify the relevant entities, attributes, and relationships from a simple case study where the case study domain was different to the case study utilised during the sessions. Further, all these sessions were scheduled ahead of time, with each participant using a Doodle poll (<https://doodle.com/en/>), and email invitations were sent out to confirm the interview time and location.

Step 4: Provide the prior knowledge survey

Next, prior to starting the session, the participants were requested to fill in a prior knowledge survey with several closed-ended and open-ended questions. This information was collected to get an idea about the participants' familiarity with developing database models before the course, which might have an impact on their learning. Please see Appendix B for the prior knowledge survey, which was the same one used during the collection of naturally occurring data.

Step 5: Conduct the think-aloud session

Next, the participants were provided with the main task of that session. An alias was assigned to each of the participants, and these were used throughout the recordings to retain anonymity. At the beginning of each session, the participants were asked to write the alias and date of the interview.

The participants attempted the task on pen and paper instead of using a data modelling tool to reduce an additional cognitive load imposed by the tool on the novices. The workings of the students on tasks were recorded using a Livescribe™ Echo 2 smartpen (Livescribe Inc., 2015b) and Livescribe™ smart paper. Figure 7.2 provides an overview of the in-person setup. The participants' interactions with the paper (i.e., writing, annotations, doodles) were recorded along with their verbalisations. The Livescribe notes and audio records were saved as Livescribe PDF files. These files can be played and reviewed using synchronised visual and audio output using the Echo desktop software (Livescribe Inc., 2015a). The participants were asked to inform the interviewer once the task was finished. While conducting the think-alouds, if the participants struggled to progress, the interviewer intervened accordingly based on the interviewer intervention protocol described in section 7.4.

Step 6: Conduct the retrospection interview

End of each session, a retrospection interview was conducted to discuss the difficulties and challenges faced by the participants while attempting task/s. Further, the interviewer managed observational notes recording any critical events, periods of silence or lack of progress as the participants worked on the task. The interviewer added reflective notes about the session based on their perspective at the end of each session. In scenarios where the participants provided incorrect answers or could not provide an answer, the retrospection interview session was turned into a post-task tutoring phase where, in some cases, the interviewer modelled how to perform the task and walked the participants through the model answer.

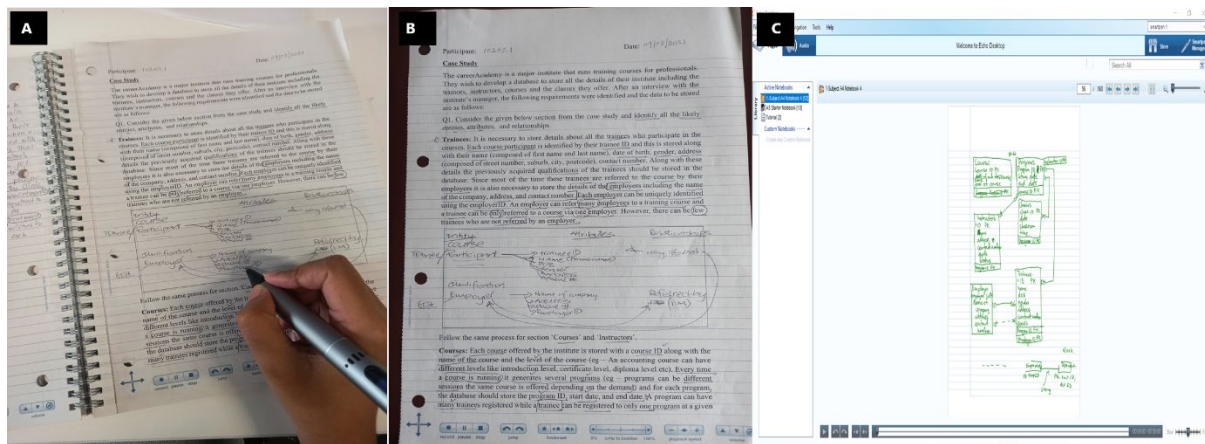


Figure 7.2 Overview of the ‘in-person’ think-aloud setup from the participant and interviewer perspective. A – Livescribe pen and paper used by the participant, B – A close-up of the task provided to the participant using Livescribe paper, C – Echo desktop used by the interviewer to play and view the recording of the participant’s think-aloud session.

Note. From “Reflections on Conducting Online Think-Alouds” by A. Imbulpitiya, J. Whalley, and M. Senapathi, 2023, ACM Inroads. <https://doi.org/10.1145/3596917>. Copyright 2023 held by the owner/author(s). Author’s own work.

7.3.2 The pilot study and protocol/instrument refinement

A pilot study was conducted to ensure the reliability of the think-aloud data collection protocol to capture data adequately. An initial group of 11 students studying the INFS601 course volunteered to participate in the pilot study. However, only four students scheduled an appointment to participate, and all four were selected to participate in the study. One participant withdrew from the study after the first session, anticipating they would not have enough time to commit to the study. Another participant dropped out after the second session without any explanation. Their data was removed from the analysis since the sessions were built on top of one another, and full participation was needed to be able to evaluate the protocol. The data from this pilot study were also included in the thematic analysis presented later in this chapter.

Even though the pilot study intended to test the developed protocol and refine it if necessary, no significant changes were required to the protocol. But, some minor technical trouble occurred with one of the recordings due to an issue with the Echo desktop. This was retrieved without causing any damage to the file, but it showed how crucial it is to have a backup copy of the audio. Hence it was decided to record the audio separately to maintain backups apart from the audio captured via the smartpen. However, during the pilot study, the interviewer encountered issues with the research instrument. For example, there were common concepts the participants struggled with (e.g., the use of composite

primary key), and some of the redirected tasks developed were not predominantly focused on these areas. Hence, the provided tasks were revisited and arranged to address this issue.

Another major concern at the time was the uncertain situation of the COVID-19 pandemic. The country had already experienced a lockdown, and there was still the potential for further pandemic-related lockdowns. The pilot study was able to be conducted and completed in person, however the full study was not able to be conducted fully in person as Auckland entered a lengthy lockdown shortly after the first interview sessions. As a result, a remote think-aloud protocol was developed in order to progress the research. This is discussed in detail in the next section.

7.3.3 Transitioning to a remote protocol

Following the first think-aloud session of the main data collection phase, the country went into a second lockdown with the highest restrictions so far encountered during the second wave of the pandemic. This necessitated a shift to online think-aloud interviews. The first step to making this transition was to investigate the existing work on conducting remote interviews, which could be adopted in this study.

Several researchers have published articles about their experience conducting one-on-one virtual interviews using various tools like Zoom and Skype (Gray et al., 2020; Lo Iacono et al., 2016). However, the only relevant work specifically on think-alouds was reported by Trate et al. (2020). Trate et al.'s work was situated in the field of chemistry education, and they described a protocol for online semi-structured think-alouds. The participants in their study were asked to describe their problem-solving process as they responded to some general chemistry questions. Their protocol's main components included the use of Doodle polls (<https://doodle.com/en/>) for scheduling interviews, in-person meetings with the participants to distribute the interview artefacts, and obtaining consent. They highlighted the importance of minimal intervention during the think-aloud sessions, the use of the participant's and interviewer's webcams to record the session conducted in Zoom, and the use of a second webcam to capture the student's interaction with the materials like worksheets and calculators. The authors mentioned the advantages they saw in their online approach:

Remote interviews allow data collection from a more representative, multi-institution sample, allow a single researcher to conduct interviews consistently across the population, are more cost- and time-effective as there is no time lost to travel, and provide flexibility for the researcher to continue other professional duties. (Trate et al., 2020, p. 2422).

The existing protocol described by Trate et al. (2020) could not be implemented directly in this study since, in their study, they used on-site personnel to meet the participant to obtain the signed consent,

provide and collect the artefacts which were sent to the researcher and to set up the audio and video recording equipment. However, the protocol developed for this research was inspired by their work.

Because the sessions had already started, the main concern was navigating the think-aloud sessions remotely. The most critical requirement in this transition was to find an alternative to replace the Livescribe™ smart pen and paper for data collection. Several video platforms were considered, and the participants were consulted about the tools that would suit them best. After much deliberation, Microsoft (MS) Teams was selected as the video conferencing platform since MS Teams was freely available for all the students via the university. Another major reason for this decision was that the recordings could be held on the university's servers to avoid any data breaches and ensure the data could be managed within the conditions of institutional ethics consent.

Another main requirement was to identify the hardware and software the participants had available to them at home that could be utilised in the data collection protocol. Initially, the use of a tablet was considered to avoid using many different tools that could have added to the cognitive load experienced during the think-alouds and thus might hinder the participant's ability to think aloud. Also, using a tablet with a smartpen had the advantage of preserving the nature of the performance as closely as possible to the original in-person usage of digital pen and paper. However, not all the participants could access a tablet, and due to the country's lockdown restrictions, the researcher could not access the resources on campus (i.e., tablets) and courier them to the participants. Finally, it was decided to allow the participants who owned a tablet to use them, and for the others, the community edition of the visual paradigm tool (Visual Paradigm, 2022) was used for modelling purposes along with simple word processors as required. While this added to the variables in the study, it also provided an opportunity for the researchers to observe the participants in two different setups and evaluate which was a better mechanism for data collection. For example, did having a tablet and a pen mean participants tended to underline and annotate key aspects of the task problem description and doodle to support their problem-solving process more than those without a tablet? Were these processes more easily captured by the researchers if the participant used a tablet? Essentially, the tasks and the way data was transcribed and analysed remained the same regardless of how the data was collected.

The participants shared their screens and microphones while working on their machines at home. MS Teams recorded the shared screen as well as the audio. In accordance with the ethics consent criteria, the participants themselves were not recorded. It was assumed that being able to see the participants in the recorded data would not alter the analysis of the data since the investigation was focused on student problem-solving and modelling processes rather than socio-emotional aspects associated with the learning. However, as the sessions went on, the interviewer observed the impact of this inability when compared to in-person think-aloud sessions.

Given that the participants were either continuing with a task or working on a new task that was based on an earlier problem, the circulation of the artefacts was another issue that needed to be addressed. Hence, the interviewer circulated each participant's previous session workings and/or solutions via email ahead of the session. However, the task specification was not emailed as the researcher needed the participants to encounter the problem at the session for the first time. This was mainly done to avoid allowing the participants to think about the problem without the interviewer capturing their thoughts. Finally, it was decided to share the interviewer's screen to display the task sheet during the initial brief of the session and also at the time the task sheet was provided to the participants via the MS Teams meeting chat channel. The rest of the protocol largely remained the same, and once the briefing was done, the interviewer instructed the participants to share their screens and open the tools they would utilise in their work. In order to maintain the relationship between the participant and the interviewer, the verbalisations during reading were somewhat enforced by the interviewer in comparison to the in-person sessions. Finally, after the recording was stopped, the interviewer instructed the participants to email their answers and any additional notes they had made while working on the task. The recorded sessions were uploaded to the university's cloud storage via MS Teams, and these were downloaded to secure storage, and the versions on the servers were deleted. Figure 7.3 indicates the online aspects of the think-aloud data collection process.

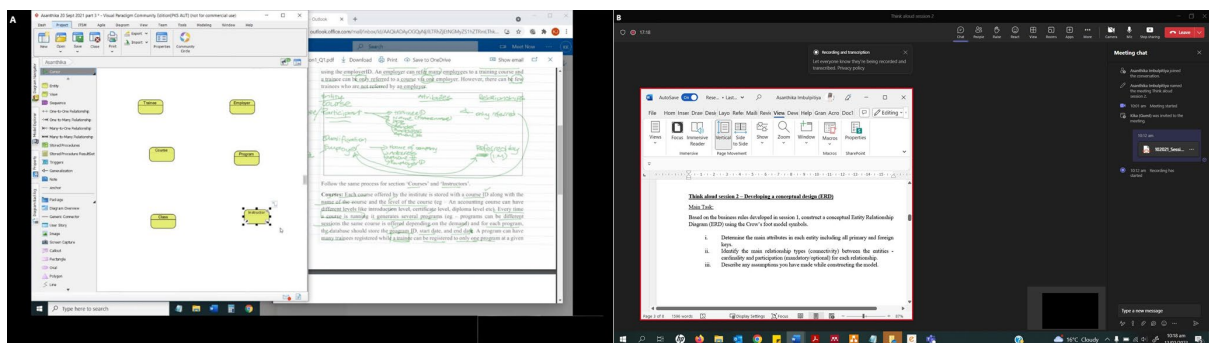


Figure 7.3 Shared screens of the participant and interviewer. A – Participant's shared screen while attempting a task, B – Interviewer's shared screen while introducing a task.

Note. From “Reflections on Conducting Online Think-Alouds” by A. Imbulpitija, J. Whalley, and M. Senapathi, 2023, ACM Inroads. <https://doi.org/10.1145/3596917>. Copyright 2023 held by the owner/author(s). Author's own work.

7.4 Interviewer Intervention Protocol

One main objective of this research was to use think-alouds to gather data closely related to the mental processes used by the participants while attempting database modelling tasks. The interventions provided by the interviewer offered an opportunity for the participants to progress their learning and to complete a task whenever they had challenges like being confused with a task, unsure about the

concepts/syntax to be used, not knowing how to progress any further or was unaware about an existing issue. The scaffoldings or interventions used in this research were adopted from the intervention model introduced by Kasto (2016) in studying novice programmers and originated from the work of Perkins & Martin (1985) and Soloway (1986).

Adopting these interventions can lead to identifying the progress of novice modellers with the Zone of Proximal Development (ZPD), a theory of learning developed by Vygotsky (1978). Vygotsky defined ZPD as *“the distance between the actual developmental level as determined by independent problem solving and the level of potential development as determined through problem-solving under adult guidance or in collaboration with more capable peers”* (Vygotsky, 1978, p. 86). Researchers have applied this theory to identify the ZPD of novice programmers in CSER (Teague, 2015; Whalley & Kasto, 2014b). Hence, it seemed appropriate to adopt the intervention model to discuss the ZPD of the novice modellers to comprehend their learning progression.

During the think-aloud sessions, the interviewer kept interaction with the participant as low as possible. However, the interviewer intervened in situations where a participant failed to progress or appeared to be confused about the next step. Sometimes, the interviewer also intervened when the participant requested assistance. These interventions were provided in a hierarchical manner, beginning with high levels of assistance and progressing to more direct and specific low-level interventions only if the previous approaches were unsuccessful. The high level of assistance involved providing general guidance, clarifications, or prompts to help participants approach and understand the tasks. This level of intervention allows participants to exercise and explore their own problem-solving skills. If the highest or indirect level of assistance did not lead to a satisfactory outcome, the interviewer moved on to more direct and low-level assistance, like providing hints or simpler tasks that aimed to steer participants towards the correct solution without explicitly giving it away—these interventions aimed to scaffold the participant’s learning process, providing support to reach their ZPD. The interviewer yielded to provide the exact solution intervention as the last option if the other approaches failed to elicit the desired outcome. Detailed information on the interviewer intervention protocol is provided in Table 7.2.

Table 7.2 Interviewer intervention protocol.

	Intervention	Purpose	Example	When to intervene
1	Clarification	Used as a high-level intervention to explain the task requirements to the participants.	Clarifying terms used in the case study or confirming the purpose of a task.	If a participant struggled with understanding the task. OR If requested by the participant.
2	General prompt	Used as a high-level intervention to get the participants to realise their mistakes and fix the issue without any support from the interviewer.	Asking the participants to re-examine their solution or to explain the meaning of a term involved in the problem description to imply that there might be an issue.	If “intervention 1” didn’t provide the expected result. OR If a participant made a mistake and didn’t realise it. OR If requested by the participant.
3	Hint	Used as a low-level intervention related to the task to let participants progress.	Suggesting a certain modelling construct to be used or indicating an error in the used syntax or a rule.	If “interventions 1 and 2” didn’t provide the expected result. OR If a participant required more direct guidance. OR If requested by the participant.
4	Redirected task	Used as a low-level intervention and redirect the participant to simpler or more scaffolded task to identify the source of the issue.	Redirecting the participants to read and answer simple questions from a partial ER diagram when struggling in developing an ER diagram.	If “interventions 1 to 3” didn’t provide the expected result. OR If a participant didn’t have the mental schema to answer the task. OR If requested by the participant.
5	Exact solution	Used as the final intervention if other interventions didn’t work.	Providing the participants with the solution and discussing it with them.	If “interventions 1 to 4” didn’t provide the expected result. OR If a participant gives up. OR If requested by the participant.

7.5 Transcription Protocol

Once the data was collected, the next step was transcribing the think-aloud data from audio to a written and readable format. The transcribing happened during three main stages, as presented in Figure 7.4.

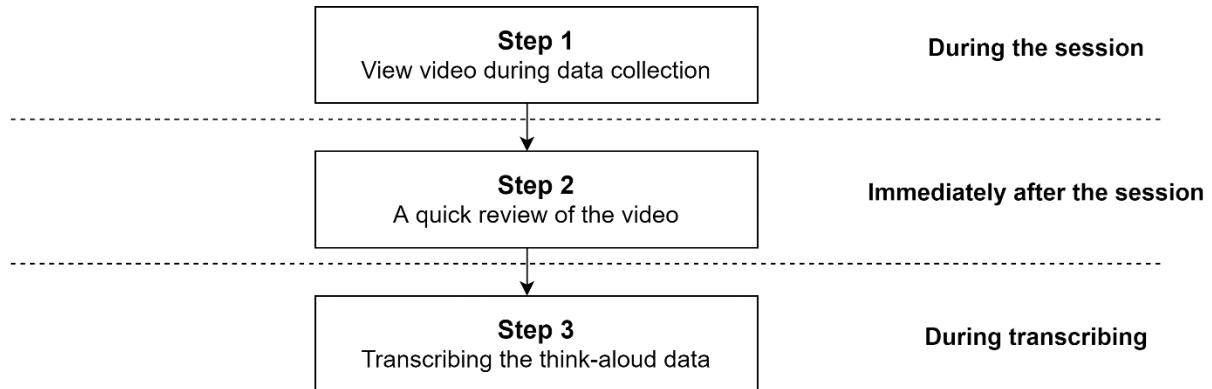


Figure 7.4 Transcription protocol.

Step 1: View the video during data collection

A preliminary step in the transcription was to view the Livescribe video during the collection of the data. This approach was quite useful since the researcher herself conducted all the interviews. In situations where she made a note on a particular observation along with the time stamp, viewing the video made it easy for her to revisit the time-stamped event during the retrospection interviews.

Step 2: A quick review of the video

As the next step, the interviewer reviewed the collected video as soon as possible after each session. During this review, the researcher also compared the audio with the notes she made in her observational diary. This helped the researcher to log the main events and the time stamps. For example, if the researcher intervened at a certain point, it was logged with a quick reason. This was helpful in scenarios where the researcher intervened due to an observation that might not have been recorded on the audio. This approach was particularly helpful during the development of the participants' LT/P, as reported in Chapter 8.

Step 3: Transcribing of think-aloud data

The audio files from each session were transcribed prior to analysis. The transcription guidelines adopted for this research are provided in Table 7.3, with examples from the actual transcription process where required. These guidelines were adopted from those presented in published literature. The references from which the practices were derived are given in Table 7.3.

The transcribing from audio to written text was undertaken by the researcher instead of making use of a third-party transcriber. This has the advantage that the researcher, as part of this process, revisited the think-aloud sessions, becoming familiar with the raw data at least twice before the analysis was undertaken as the transcriptions were verified. Another reason the researcher decided to do the transcription was due to issues that may arise from using third-party transcribers, including the need to train the third party, the unfamiliarity of the third party with the research context, and the lack of a researcher's perspective during the transcribing (Johnson, 2011; Jordan & Henderson, 1995; MacLean et al., 2004).

The initial transcription of the audio was performed using two different software tools. When sessions were in-person, the data had to be transcribed in a separate step using a transcribing tool, Descript (Descript, 2021). However, with the transition to the remote think-aloud protocol, the data were transcribed using MS Teams transcription facility. Following this initial transcribing, the interviewer reviewed the transcript manually, following the guidelines provided in Table 7.3. This phase allowed for the automatically transcribed transcription to be validated and corrected as necessary. While both the tools provided reasonably accurate transcriptions, Descript was preferred over the MS Teams transcription facility due to certain features like its ability to assign speaker roles in distinct colours, which made the transcription more readable and easier to correct where necessary.

In the case of the data gathered in person, some utterances were marked as “inaudible”, which the researcher did manually. In order to try and complete the data, the researcher also re-examined the Livescribe video to compare the audio with what the participant was currently writing. The retrospective interviews were also transcribed following the same guidelines as the think-alouds.

Table 7.3 Adopted transcription guidelines.

Adopted guideline	Source	Example
Use of preambles (i.e., alias of the participant and interviewer) to clearly distinguish between the utterances of the participant and the interviewer. A unique colour for each student alias throughout all the sessions.	(Jenks, 2011)	[00:00:06] Interviewer: Maybe you can write entities here and list them down and attributes just list them down. Any way you like. [00:00:22] 102021: Okay. Entity and attributes. And do I need to worry about the relationships at this moment? [00:02:34] 112021: Um, only sort relationships I did was like, um, relating primary keys and foreign keys and Access and stuff. And like, we would, we would have to like draw them out.
Add references made by the participant or the interviewer to a previous session or made gestures not recorded in the audio file (e.g., point to something on the page in square brackets “[]”).	(Somerén et al., 1994)	[00:13:39] 102021: Each course offered by the institute is stored with a course ID, so course [write under entity list]. So where is it [refer back to the case study], course ID., course name, level of course [write under attribute list]. What else was there? Every time a course
Mark hesitations in speech. E.g., using “...” for a quick hesitation or a break in utterance up to 5 seconds and “[pause]” for longer breaks over 5 seconds.	(Ericsson & Simon, 1993)	[00:13:39] 102021: What else was there? Every time a course is running [box it] ... it generates several programs. Course is [00:15:00] running it will generate, generate several [00:11:29] 102021: Ahhh courseID [pause] [00:11:48] Interviewer: Yeah, what are you trying to do at the moment?
Mark any utterances that didn't make sense during the initial transcription as “[inaudible]”.	(Teague, 2015)	[00:02:56] 112021: Then then. Multiple. Please. It's got links. And. [inaudible]
Non-task-related verbalisations were marked separately. e.g., Modelling tool, reading of the case study	(Rosenthal & Strecker, 2019)	Participant 12021: Ok the careeracademy is a major institute that runs training courses for professionals. They wish to develop a database to store all the details of their institute including the trainees, instructors, courses and the classes they offer. So I think in this sentence every noun is an entity, because oh I'll continue reading. After an interview with

7.6 Researcher Subjectivity and Reflexivity

Braun and Clarke (2013) explain that research is a subjective process where the researchers bring their own assumptions, experiences, and perspectives into the research –researchers’ subjectivity or the position they hold in the research. They further mention that reflective thematic analysis is more about the researcher’s reflective and thoughtful engagement of the data and the analytic process than merely following a set of steps (Braun & Clarke, 2019). Subjectivity and reflexivity have become integral components in many qualitative studies. Reflection in the reflexive thematic analysis involves more than simply outlining the researcher’s position on how they relate to the population being studied – it is more about the role of the researcher as an active participant throughout the entire process. This involves how they relate to the target population as well as how their experiences and beliefs inform their interpretations. Hence, reflection is important to all the phases in the analytic process. In this

section hereafter, the writing format is changed to first person view to present the researcher's reflection and subjectivity since it is a personal encounter.

7.6.1 Researcher subjectivity

The research I presented in this chapter required me to get involved with the participants as an interviewer during the data collection phase. This section discusses how I positioned myself within the study and acknowledges my subjectivity throughout the data collection and analysis.

Firstly, I should acknowledge my familiarity with the thematic analysis approach. Being a novice to qualitative research, this was my first experience undertaking a qualitative approach to research, let alone reflexive thematic analysis. There was a learning curve in understanding the overall approaches to thematic analysis, particularly the reflexive thematic analysis, which was quite time-consuming. However, prior to my PhD studies, I was an academic with seven years of experience and have taught undergraduate software and database modelling courses in a tertiary institute in Sri Lanka, where some of the courses belonged to offshore programs affiliated with two institutes in the United Kingdom and Australia. This gave me a wide spectrum of understanding about the introductory level modelling-related curriculum in a few countries. Further, as a teaching assistant, I have taught laboratory classes in the courses used to collect data for this study in earlier cohorts.

This context with familiarity provided me with some benefits, such as a basic knowledge of the domain, better access to the participants, and ease in developing a rapport with them. However, it was also a challenge for me at certain times not to interpret the data beyond what was provided by the participants during the analysis. This experience, as an "insider" (Holmes, 2020), is the lens through which I viewed and interpreted the participant's think-aloud data in this study.

As I mentioned earlier, as a novice to qualitative research, my supervisors played an active role as mentors and co-researchers to a certain extent in all phases of the research process. Their involvement was in fact, a collaborative effort where I could reflect on what I did and consider new possibilities rather than getting their approval or agreement. Braun and Clarke discussed the same as follows:

From a quality point of view, the aim of such review and reflection is not to determine whether you've got it 'right' or your mentor or supervisor 'agrees' with your analysis. Rather its an opportunity to: explain and clarify your thinking; be questioned; explore alternative ways of making sense of and interpreting the dataset, coding and patternings that you haven't considered; and reflect on whether your particular standpoints or experiences are (problematically) limiting and constraining how you engage with the data. (Braun & Clarke, 2022, pp.273-274)

The supervisors' positions and standpoints in consideration of this study are stated below.

The first supervisor is an experienced CS and SE education researcher with over 20 years of experience. She mainly works with novice software developers to understand their challenges and research how research-informed teaching approaches may be applied to enhance student success. Her research uses the lenses of sociology and psychology to improve student outcomes in computer science programs. Considering the first supervisor's research engagements, she can be considered an 'outsider' (Holmes, 2020) who didn't have any prior knowledge of the group being involved in the study.

The other supervisor is a researcher mainly active in IS, CS, and SE education research. While her being the lecturer-in-charge of the courses where the participants were enrolled, she mainly looked at the data from an 'insider' perspective. Several precautions were taken to mitigate any power imbalances brought up due to this relationship throughout the study. Her insights as a domain expert who was teaching introductory level database modelling courses for over 25 years were particularly useful during the data analysis phase.

7.6.2 Personal reflexivity

As mentioned in the earlier section 7.6.1, being a Teaching Assistant for the earlier cohorts of the courses where the data was collected, it would be impossible to be involved in the study without being influenced by past experience and having preconceptions about the participants' behaviour and performance. It's not possible to fully relate to the participants and their experience while attempting a task however, being someone who taught students in their labs made me relate to them, at least in part, to some of the challenges they came across. Before the think-aloud studies, I reflect on my experience in teaching so that I can use my experience as someone who taught similar courses to build empathy and rapport towards the participants to make them ease into the think-aloud sessions.

During the recruitment, I emphasised the fact that the sessions are to understand how they learn rather than criticising their attempt. Also, during the sessions, I was careful not to judge the participants and intervene with them before they requested help or couldn't proceed. Also, I did my best to respond quickly whenever the participants had questions or felt overwhelmed with the sessions. Considering that the second phase of the sessions was conducted amidst the lockdowns, I spent some extra time with the participants and talked to them, which they appreciated. I was interested in the way the participants attempted to develop their models, the challenges they faced and the tactics they used in developing the models. Despite looking for these commonalities among the participants using thematic analysis, I also wanted to explore how each participant approached their tasks differently despite being taught the same processes in the class.

Further, I maintained a handwritten journal throughout the study to write down my thoughts about the participants' sessions. This journal mainly included field notes taken during the think-aloud data

collection – about the participant’s reactions, emotions, observations, thoughts and judgements while conducting the sessions. The following is an example extract, degendered, from my field notes made immediately after a session:

They seemed to struggle in the beginning, even to start the task. I had to ease them into it and explain the task in detail compared to others to get them to start the task. However, after this, they managed to start and continue.

They took more time than others and struggled with basic concepts like adding cardinalities and deciding on the primary keys. Along with their fragile knowledge of concepts using the tool also seemed to be more challenging. But with the help I provided, they managed to progress, and I felt what they were going through since I’ve come across similar students during my time as a lecturer.

During the retrospect interview, they opened up further and explained that it’s harder for them to grasp the concepts properly despite attending all the classes. I discussed my experience as a lecturer with them and told them I’d met similar students, but with time they could be better at developing a model. The reassurance seemed to make them more interested in attending the next session. I’ve realised how important to have this quick discussion at the end of a session to build a rapport with the participants. Maybe that will ease them into the process more.

Also, being a first-timer to qualitative research and conducting think-alouds, I had a practice session with one of my supervisors acting as a participant. Further, I debriefed my supervisors about the sessions frequently and documented their comments and feedback, which immensely helped me to reflect on what I’ve done during the sessions.

All these notes I’ve made in my journal helped me to articulate my thoughts during the data collection, analysis, and writing phase. Further, based on these observations I made during a session, sometimes I made slight changes to the interventions I developed to help the participants better. For example, I didn’t develop any interventions to help the participants with their knowledge of composite attributes. However, after a session or two, since the majority seemed to need extra help to understand this concept, I added a simple task that could help them if needed. This reflection I did throughout the process helped me to complete the data collection and analysis successfully, particularly as a novice to qualitative research.

7.7 Analysis Protocol

The think-aloud data were analysed using the qualitative method of reflexive thematic analysis (Braun & Clarke, 2022). This method is typically used to analyse the think-aloud data as a group to identify common patterns or themes in the data. In this research, an analysis method that could identify common themes was required to understand novice modellers’ difficulties, challenges, and common tactics they

use when attempting a modelling task. Hence, using a thematic analysis (TA) method was deemed suitable for analysing the data.

Within the spectrum of TA approaches, the *coding reliability* approach is considered to fit into the smallQ paradigm (involving aspects of both quantitative and qualitative methods) underpinned by a positivist philosophy and a more structured approach focusing on the accuracy and reliability of coding. In contrast, *reflexive TA* is considered purely qualitative within BigQ paradigm and looked at through an interpretivist lens that emphasises an organic and flexible approach to coding. In contrast, the *codebook TA* approach lies somewhere between *coding reliability TA* and *reflexive TA* approaches. Codebook approaches (e.g., template analysis, framework analysis) combine the structured approaches of coding reliability TA with the underlying qualitative philosophy of reflexive TA. Reflexive TA is considered to be highly flexible and organic compared to the other two approaches. Codes and themes evolve throughout the analytical process (Braun & Clarke, 2019). Further, the researcher's subjectivity is not criticised in this approach but rather considered as adding value to the analysis process. Reflexive thematic analysis is a "theoretically flexible method for developing, analysing and interpreting patterns across a qualitative dataset" (Braun & Clarke, 2022, p.4). The objective is to provide a persuasive and coherent interpretation of the data, grounded in the data.

The data at this phase was analysed using the reflexive thematic analysis of Braun & Clarke (2022). As they outlined, the main reason for selecting reflexive TA was that it allowed the researcher to determine the outcome of the work without being bound by a specific theoretical framework. Considering that the question the researcher wanted to explore was somewhat an understudied area in CSER, an approach with flexibility was a must – Braun & Clarke (2006) define their approach as an 'exceptionally useful' method to examine understudied areas. Another reason for selecting the reflexive TA was that it allows the researcher to be reflexive and critical about their assumptions and biases. Considering the position of the researcher as an insider, the flexibility of the method allowed the researcher to collect and analyse the data, expressing the participant's opinions, experiences and attitudes as effectively as possible while taking the researcher's interpretation into account. Also, since the coding was mainly done by only one researcher, the reflexive TA approach was much more suitable compared to the other two approaches where multiple coders are involved. The supervisors were involved in coding and theme generation however, this was a collaborative process rather than trying to achieve consensus among each other.

A predominantly inductive approach was taken to analysis, working with the data from the bottom-up (Braun & Clarke, 2012). In this study, data were not coded to fit any pre-existing coding framework but coded openly. The analysis was grounded on the data to understand how novice database modellers learn, their perspectives, experiences and challenges faced during the learning process. Even though the approach was predominantly inductive, an underlying deductive approach was employed to ensure that

the open coding produced relevant and meaningful themes to the research questions. The analysis was conducted from a critical standpoint, as the aim was to understand the experience of participants while attempting the database modelling tasks, along with factors that shaped their understanding.

Throughout the analysis, the researcher sought to identify the patterns across the data to tell a story of how novice database modellers learn to model. The six phases of Braun and Clarke's (2022) reflexive thematic analysis were used through an iterative process described below.

Phase 1 – Data familiarisation

The researcher was able to become closely familiar with the data by managing all aspects of data collection, including creating think-aloud tasks, running the think-aloud sessions, and transcribing them. Further, the think-aloud sessions allowed the researcher to actively engage with the participants, get to know them and comprehend their learning before discussing what transpired in the session during the retrospective interviews. Through this process, the researcher could fully immerse in the data before actively engaging in the analysis. First, the audio was transcribed using a tool, as mentioned in section 7.5, which was automatic transcribing. After this, approximately 19 hours worth of transcriptions were updated and corrected by viewing the video, and despite being a slow process, this process enabled the researcher to understand the data thoroughly as it was done while listening to the audio and watching the video of what the participants did. Braun and Clarke (2022) suggest asking certain questions to critically engage with the data during the familiarisation phase, such as “How does the person make sense of whatever it is they are discussing? What assumptions do they make in describing the world? What kind of world is revealed through their account?” (p.44). The researcher managed an observational diary during data collection, which was utilised during transcription to note interesting observations, including personal reflections on what the researcher noticed about the participant when attempting a task and how they used different tactics in developing the solution (see Figure 7.5).

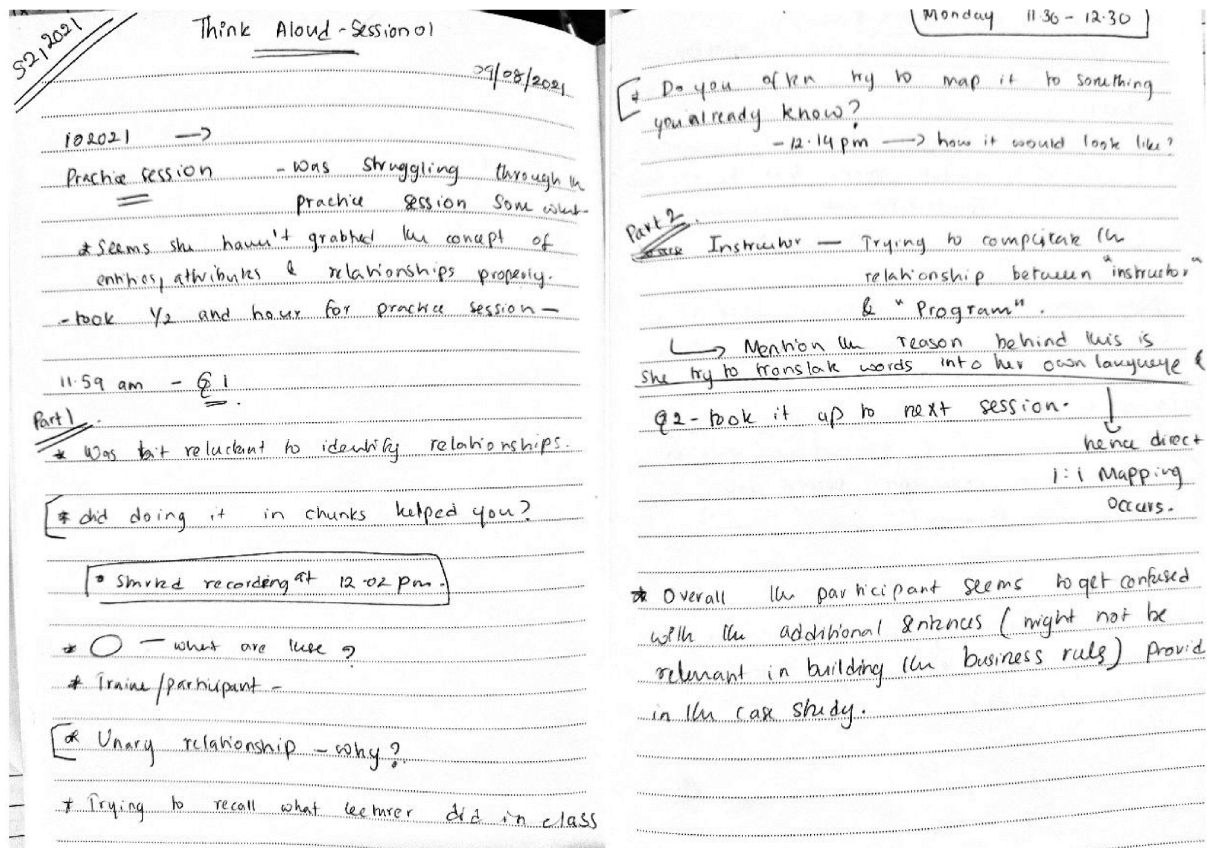


Figure 7.5 An extraction from the observational diary.

Phase 2 – Data coding

Following the transcription phase, the coding phase commenced. In this phase, verbalisations that provided meaningful snippets related to understanding novice database modellers were identified, guided by research questions 3.1 and 3.2. The think-aloud transcripts were coded using the comment feature in Microsoft Word, where relevant excerpts were highlighted and commented on with a suitable name. Initially, qualitative data analysis tools like Nvivo were considered for the coding of data. One major reason to deviate from using a tool was that using such tools can take the researcher away from the total emersion of the data compared to the manual coding, where the researcher needs to re-engage with the data regularly. Braun and Clarke highlighted their experience with manual coding as follows:

This approach worked for each of us – because we have found we think and engage with data differently when we read it as hard copy compared to electronically. We also liked to leave the screen behind, and to code in all sorts of different spaces – which can bring subtly different things to the process. (Braun & Clarke, 2022, p.66)

Since this is the researchers' first experience in conducting a thematic analysis, it was decided to go ahead with the manual coding to avoid the additional difficulties related to electronic coding. The

DocTools ExtractData word add-in (Fredborg, 2021) was used to collate all the codes, and the excerpts, page number, line number and date were then saved into a separate Word document.

Initially, alongside the researcher, the supervisor also coded the same transcription independently. This was conducted mainly as a point for reflection in which the researcher later cross-referenced the two perspectives in order to refine and review her codes. The coding was an iterative process that progressed slowly, and the researcher had to step away several times in order to return to the same data with a fresh frame of mind. Initially, the codes derived tended to be more semantic, such as ‘proposing assumptions’ or ‘annotating a case study’, primarily focused on identifying and categorising the explicit or surface-level elements of the data closer to participant language. This was expected since the coding process was more inductive and exploratory. However, as the coding progressed, some possible latent codes were identified, like ‘readability of model’, ‘assumptions lead to incorrect solutions’ and ‘real-world knowledge lead to incorrect solutions’. Latent codes delve into underlying meanings or concepts that may not be immediately apparent from the data, which is sometimes quite abstract from the explicit content of the data (Braun et al., 2019). For example, the codes related to using assumptions and real-world knowledge suggest a relationship between participants’ assumptions/real-world knowledge and the incorrect solutions they arrive at. It implies that participants’ preconceived notions or beliefs may influence their decision-making and lead them to arrive at ultimately incorrect or flawed solutions. In this given context, the decision was made to keep the latent codes in generating the themes despite the initial suitability of semantic codes. This decision was driven by the recognition that latent codes reveal underlying meanings related to the decisions made by novices, providing valuable insights into their thought processes.

The researcher reviewed the dataset several times through the coding process, and the codes were renamed, merged or removed as they evolved. Further, during this process, the researcher's discussions with the supervisors helped refine and reform the codes. As coding progressed, the researcher identified several patterns within the data that could be collated together, and mind maps were developed that could be later used in theme development (Figure 7.6).

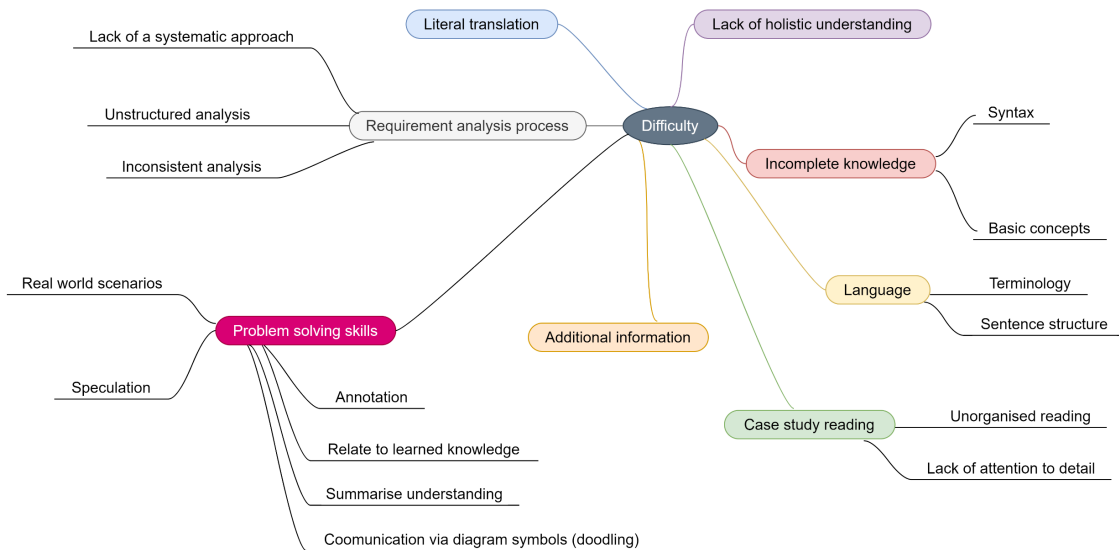


Figure 7.6 An initial mind map developed during the coding process.

Phase 3 – Initial theme generation

Braun and Clarke (2022) note that this phase is about finding, losing, and then finding the way again. This doctoral researcher experienced the path, revisiting and questioning themes through several iterations.

As discussed earlier, the researcher identified patterns from the data set while coding. Starting with the raw codes, the researcher in this phase began the process of collating and organising the codes. Starting initially with the common patterns that emerged and were noted during coding. This was quite time-consuming as the researcher was new to qualitative analysis. The process was strictly inductive. The research consciously attempted to avoid framing the themes based on the ideas and definitions from previous research. However, since the researcher had knowledge of prior research, there were themes in the researcher's mind that came from similar research. This knowledge, at certain times, tended to introduce potential biases during data interpretation and resulted in the researcher questioning their data-driven codes. For example, initially, a code named 'lack of holistic understanding' was identified from the analysed data. However, after much consideration, it appeared that it could not be grouped under any of the identified themes at that point. Hence, the researcher took a step back and reexamined all the data extracts coded under this. The code 'lack of holistic understanding' came from the work of Batra and Davis (1992), where they discussed how the experts are focused on developing a holistic understanding of the problem before developing a model. However, when you actually looked at the data through the lens of the participants in context, it appeared that the issues participants had were related to other difficulties (e.g., in one instance, a participant struggled with their lack of attention to detail in reading the case study) rather than the lack of a holistic understanding. Hence, the researcher

found it necessary to repeatedly return to the raw and transcribed data to ensure that the themes and codes were grounded in the data during this phase.

After initial theme generation, the researcher discussed the outcome with the supervisors only to realise that some identified themes were topic summaries rather than themes. Braun and Clarke (2021) explained topic summaries as “summaries or overviews of the most frequent things participants said in relation to a particular topic or data collection question” (p. 39). Unlike a theme which is defined as patterns in the data underpinned by a central concept, a topic summary doesn’t have anything that unifies what participants said about this topic. It simply summarises the participants’ responses to a question that Braun and Clarke referred to as a shared topic but not a shared meaning. For example, one initial theme, *reasons for difficulty in modelling*, was identified as a topic summary rather than a theme which included quite different data or even contradictory data (e.g., lack of knowledge, language barriers, or struggle with tools). Such a topic is useful but fails to provide the shared meanings that are required from a reflexive thematic analysis. Hence, themes subsequently deemed to be topics were removed. Further, looking at the data again, the researcher reflected on the identified themes and realised that some codes collated under the same theme were contradictory or didn’t make much sense. For example, the code ‘Terminology’ was grouped under the theme *Language* however reflecting on it, confusion came up about whether it should belong to the theme *Incomplete knowledge* since the coded data extracts were examples of scenarios where the participants struggled with their lack of knowledge in the terminology they learnt related to the database modelling domain. At this point, the researcher decided to start over the theme generation again.

With the newly acquired knowledge, the researcher attempted to generate themes again from the raw codes rather than evolve from the original themes. Instead of using Microsoft Word (e.g., using tables and lists) to collate the codes into themes, a more hands-on approach was used this time. The researcher used sticky notes to write down the codes so they could be easily moved into different groupings as part of identifying themes. This approach allowed the researcher to conceive themes and move codes between them as the themes evolved into patterns of shared meaning. Figure 7.7 shows this first evolution of theme development.

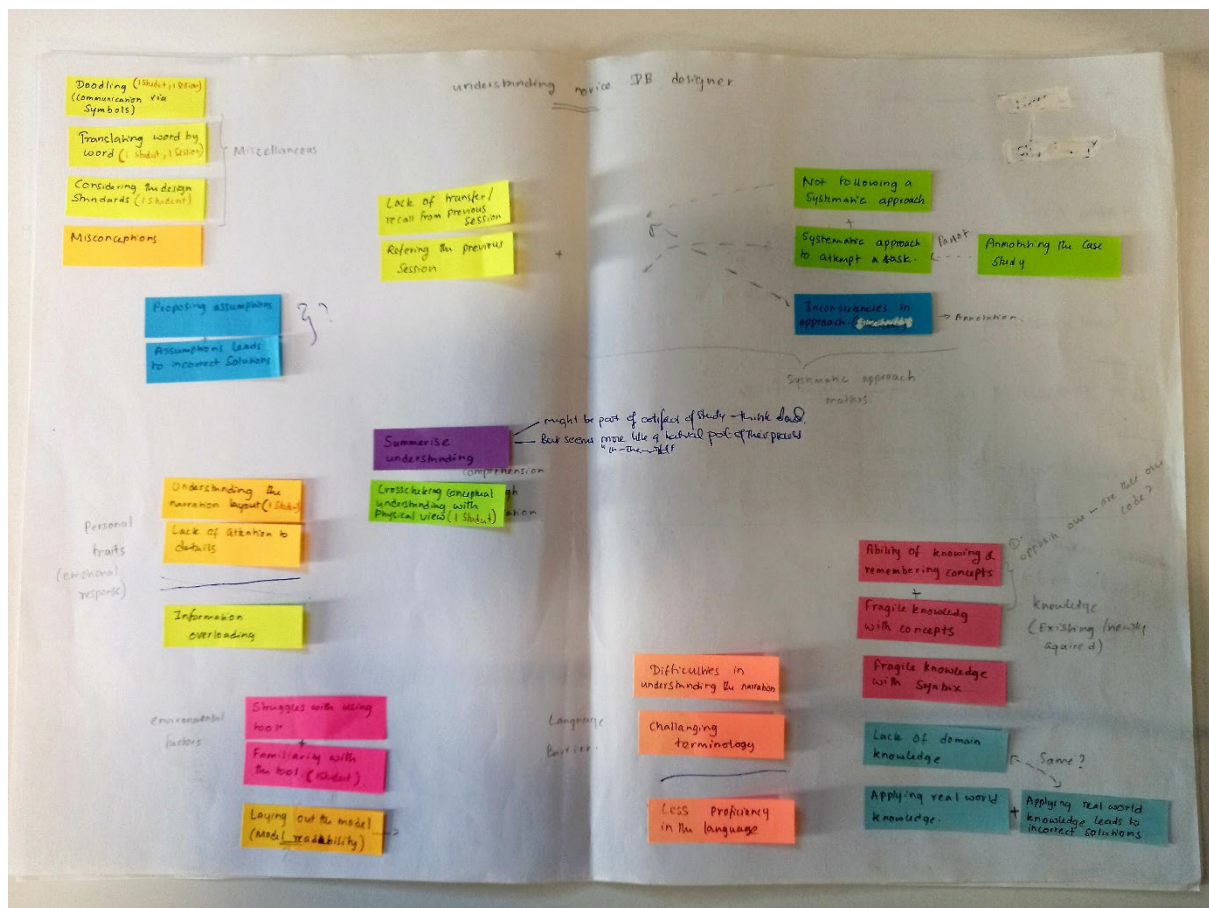


Figure 7.7 Initial attempt at theme development.

During the development of themes however, there were scenarios where the researcher doubted her decisions and lost her way, uncertain about whether to move on to the next phase or not. But Braun & Clarke explain that the absence of a simple, easy-to-apply rule for reflexive thematic analysis can lead to these kinds of uncertainties and doubts (Braun & Clarke, 2022). After spending some more time on this phase, the researcher realised that she arrived at the same outcome even after revisiting the data. Hence, at this point, the researcher realised that it was time to move on to the next phase.

Phase 4 – Theme development and review

This phase was started by reviewing the candidate themes against all the extracted data from the think-aloud sessions. At this point, the researcher collated each code, the associated excerpts, the participant id, and the source session into a single sheet in Microsoft Excel. The researcher created physical cards for each code with sample excerpts to make the theme development efficient and realistic, as suggested by one of the supervisors. This card sorting exercise helped the development of the themes as it was easy to move codes manually along with the thinking process. Braun and Clarke describe this process as recursive, where the researcher has to move backwards and forward between the data and the analysis

(Braun & Clarke, 2022). In fact, during this phase, several versions of the themes were created, and the supervisors were involved in giving feedback and drawing on their domain expertise.

In the first attempt, seven themes were identified, and these were cross-referenced with the initial candidate themes developed by the researcher. Although the current themes were somewhat similar to the candidate themes, the number of themes dropped from nine to six by merging some and including a new sub-theme. When moving from the candidate themes, it was realised that the *comprehension through verification* and *steps in process* themes could be, in fact, merged together. *Comprehension through verification* theme consisted of two codes: ‘summarise understanding’ and ‘cross-checking conceptual understanding with physical views’. Even though these two codes were put under a separate theme, after reflecting on it, they seemed more like a part of the participants’ process or approach to attempting the tasks. Hence the two themes were merged and under the theme *steps in process*. Further, among the candidate themes, a theme named *miscellaneous* consisted of the codes ‘lack of recall (transfer) from previous session’ and ‘referring from/to the previous session’ were merged into the existing theme *cognitive process*. Initially, the researcher named this theme as *miscellaneous* as she wasn’t sure where it would fit, but upon reflecting on the decision and a discussion with the supervisor, it was decided to merge the two codes with the *cognitive process* theme since the data the codes were related to have the ability to transfer knowledge learnt on one task to the next. Similarly, two other codes, ‘proposing assumptions’ and ‘assumptions lead to incorrect solutions’, were also kept as a separate theme where the researcher was unsure about the theme name or where the codes should belong to. Tentatively, it was decided to merge these codes into the cognitive process theme as that appeared to be the closest theme that could relate to the codes. Both codes were about making assumptions as part of the modelling process, with some assumptions leading to success while others led to errors. Finally, after looking through the theme *steps in process*, it appeared that three codes (‘systematic approach to attempt a task’, ‘not following a systematic approach’, and ‘inconsistencies in the approach’) within the theme were related to the actual overall process the participants followed rather than being related to a step in the process. Hence these three codes were taken out and placed under a new theme named *process and strategies* where *steps in process* theme became a sub-theme of it. Version 1 of Figure 7.8 indicates the initially developed themes with the card sorting activity. The card sorting exercise was beneficial because it allowed the researcher to shift the codes around as their thoughts developed.

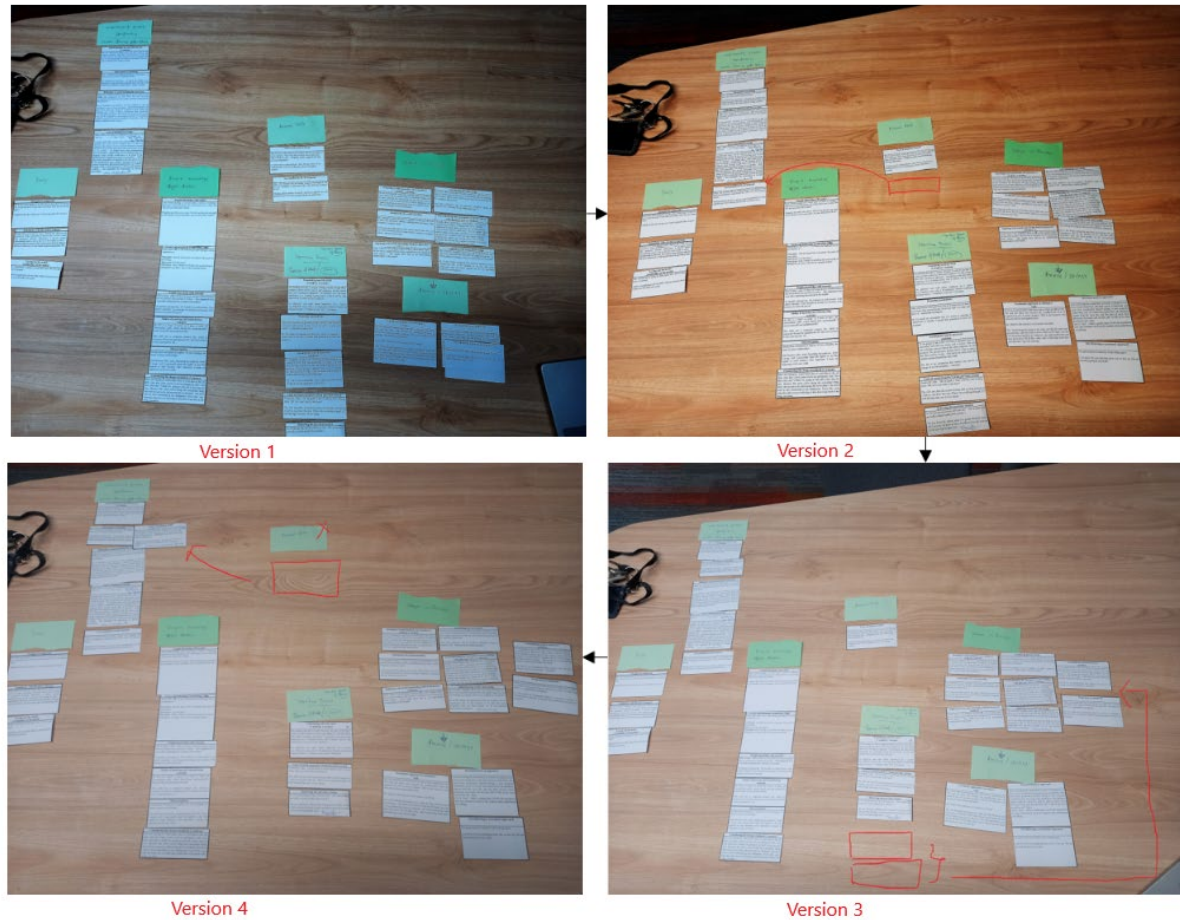


Figure 7.8 How the themes evolved during the review process.

After identifying the themes in version 1 of Figure 7.8, the researcher reflected on them and the supervisor's feedback and decided to refine the themes further. Doubts emerged about the theme, *personal traits*. Under this theme, there were only two codes collated, 'lack of attention to detail' and 'less proficiency in the language'. The main issue came with the latter: the researcher felt 'less proficiency in the language' directly related to understanding the case study or the theme *understanding problem specification*. For such participants, despite their need to follow a particular process in completing the task, it becomes difficult since they can't understand the problem due to their lower proficiency in the language. Even though it's a personal trait related to their prior knowledge and skills, the artefact from that is such participants move into developing models without understanding the problem specification. With this, the code 'less proficiency in the language' was moved under the theme of *understanding problem specification*. Another question raised at this point was whether there was a need for the theme of *personal trait* altogether. Considering that there was no research question related to this, the researcher examined the coded data again and realised the theme of *personal trait* was generated out of interest rather than in relation to the research questions. However, the researcher decided to keep the theme for the time being as a placeholder to be revisited in the next round of

refinement. The discussed theme development as indicated in version 2 of Figure 7.8, is illustrated in Figure 7.9.

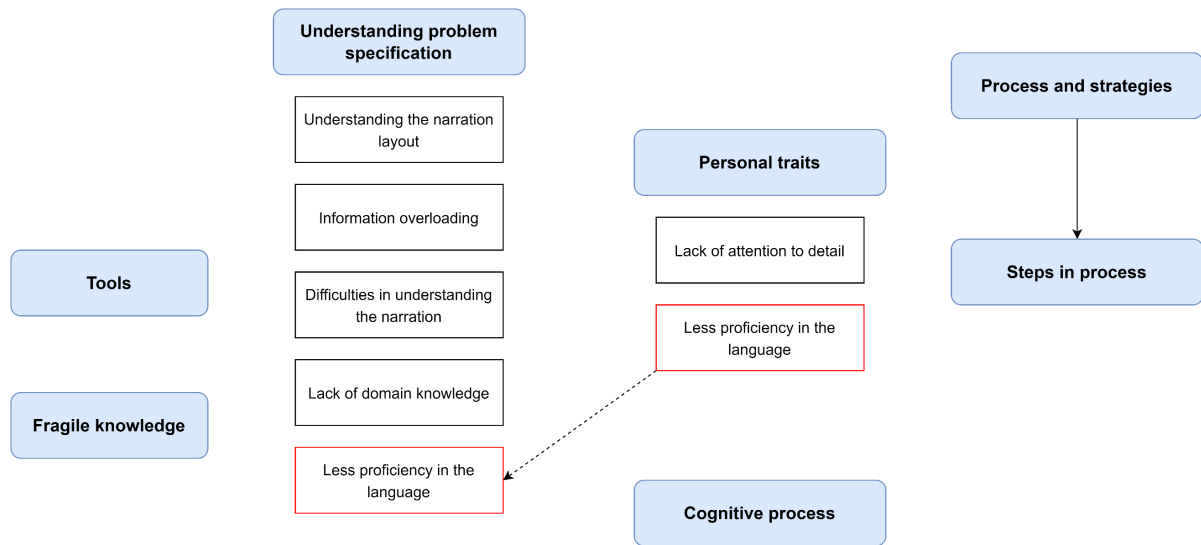


Figure 7.9 Evolving of the themes – version 2.

The theme of *cognitive processes* also required rethinking. Initially, the codes ‘proposing assumptions’ and ‘proposing assumptions lead to incorrect solutions’ were categorised under this theme. However, after the discussion with the supervisor, the researcher realised that these codes were simply a step in the process of developing a model that is taught to the students. The researcher realised these assumptions were necessary to solve the problem if the participants determined that necessary information was missing from the case study. This was confirmed after looking back at the data. For example, during the first task, Eli mentioned: “*It doesn't say anything about the relationship of a program to a course, but can do it with an assumption...*”. Due to this, the codes ‘proposing assumptions’ and ‘proposing assumptions lead to incorrect solutions’ were moved from *cognitive processes* theme to *steps in process*. How the themes evolved from version 2 version is illustrated in Figure 7.10.

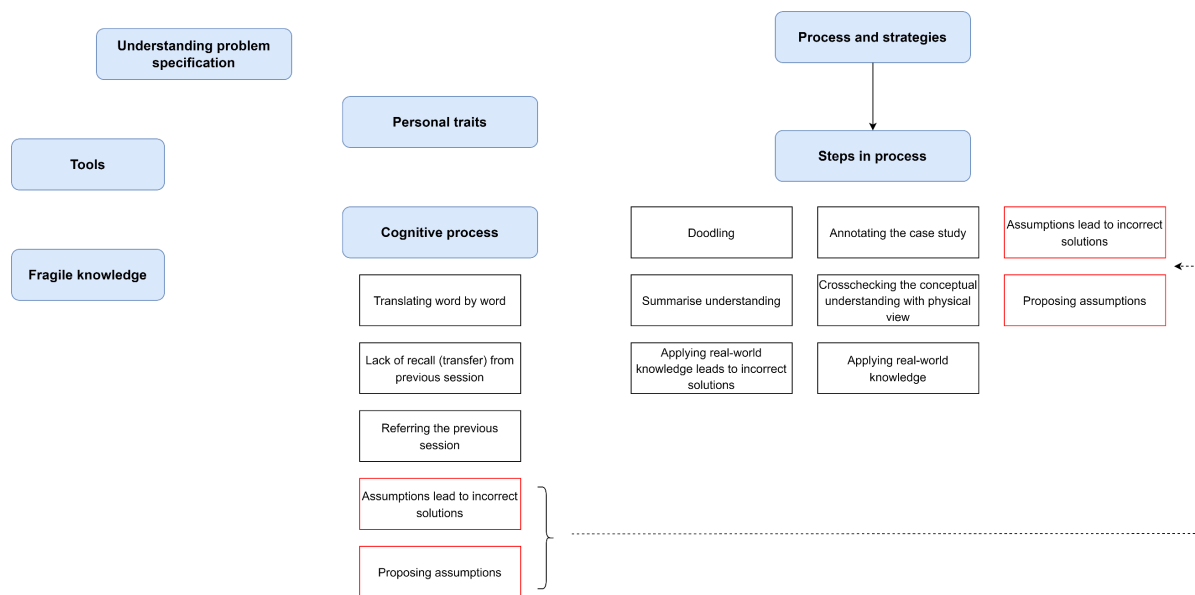


Figure 7.10 Evolving of the themes – version 3

After this version, the researcher was satisfied with the generated themes. However, there was only one code under *personal trait*, and the question “Are there enough data to evidence this theme” (Braun & Clarke, 2022a, p. 99) was asked to review the theme. The answer was no, and hence, the researcher decided to remove the theme of *personal trait* as decided initially. This raised the issue of what should be done about the code ‘lack of attention to detail’. The researcher went back to the data to understand whether the lack of attention led the participants to miss something from the problem specification. If that’s not the case, then there was probably an issue when the problem specification was translated into the model. With the extracted data, the researcher realised it was more about misunderstanding the problem specification due to the lack of attention in working with the case study. Hence, the code ‘lack of attention to detail’ was placed under the theme of *understanding the problem specification* (see Figure 7.11 for the updated themes mentioned in version 4 of Figure 7.8). The theme development and review happened while going back and forth between the data, and the initial themes evolved with the reviews. Figure 7.8 indicates how the themes progressed during this process.

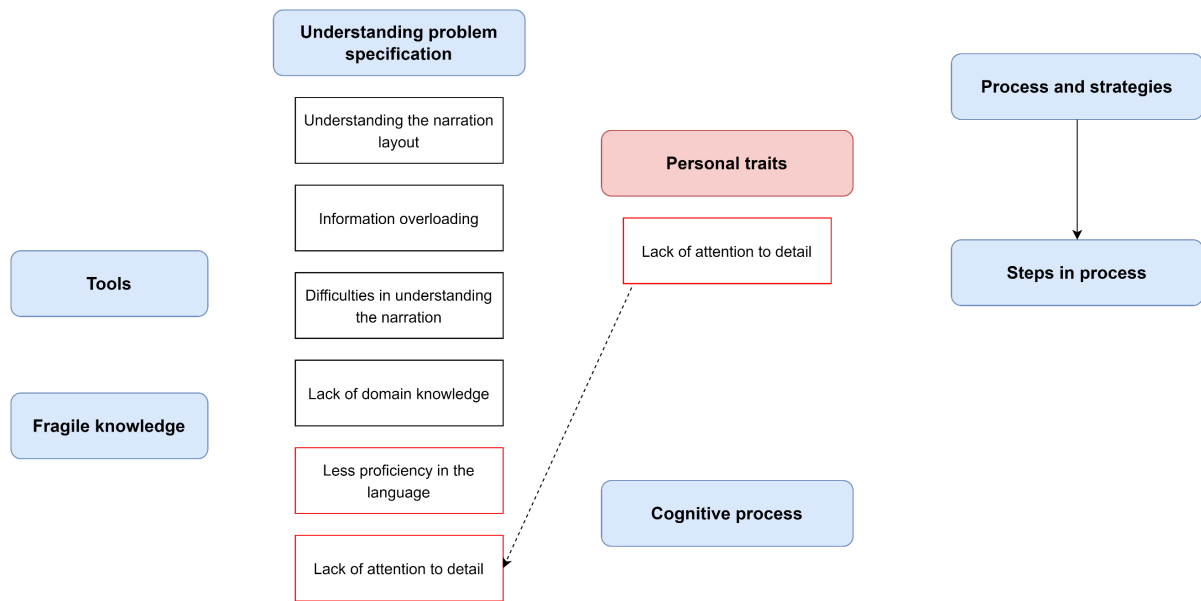


Figure 7.11 Evolving of the themes – version 4.

By the end of the reviewing phase, six main themes were identified *tools*, *understanding the problem specification*, *fragile knowledge*, *cognitive process*, *steps in process*, and *process and strategy*. The theme *process and strategy* and its subtheme *steps in process* did not properly present the idea of the code grouped under them, hence it was decided to reconsider the name of the themes. These refinements are discussed in detail in the following section. See Table 7.4 for an overview of the identified themes and the theme definition. Alongside the theme development, the researcher developed thematic maps, which were updated as the thinking progressed. As per Braun and Clarke, thematic maps are mainly used to capture the figurative representation of the potential themes and their relationships (Braun & Clarke, 2022). Figure 7.12 illustrates the refined thematic map the researcher developed at the end of this phase.

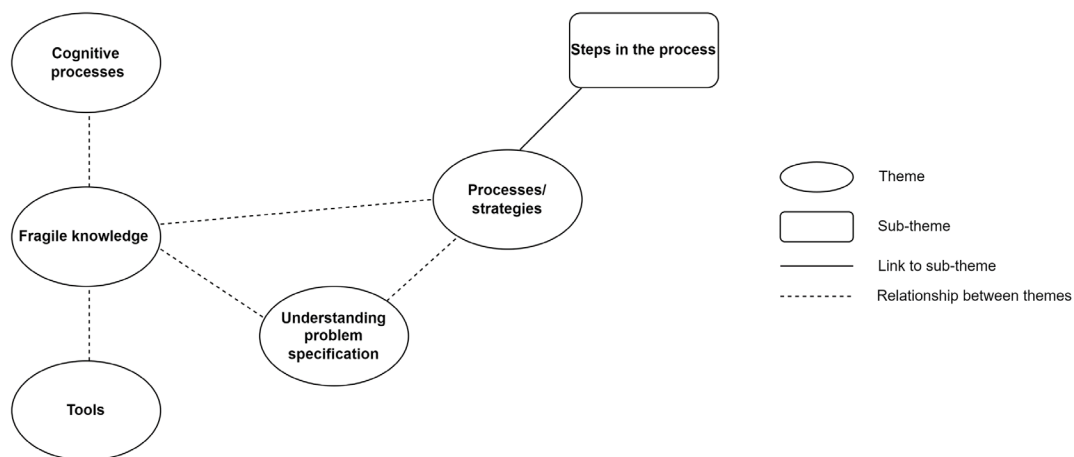


Figure 7.12 Refined thematic map.

Phase 5 – Theme refining, defining and naming

Braun and Clarke describe this phase as a phase that “involves further development around your themes, as well as more precise analytic work refining your analysis” (Braun & Clarke, 2022a, p.108). At this stage, the researcher articulated the identified themes to both supervisors, one of whom is a domain expert in databases who brought more clarity to the identified themes. This led to further refinement of codes being moved over different themes to precisely fit the related ideas.

Upon revision, it came to light that several codes under the *steps in process* and *process and strategy* themes contained several codes that explicitly describe the tactics used by the participants during the interpretation of the case study, which is the most important stage in developing the model. Hence it was decided to form those codes under one theme called *interpreting the case study*. Further, the codes that described the results that occurred due to using the identified tactics incorrectly (i.e., code – ‘assumptions lead to incorrect solutions’) were also placed under this theme. Also, the codes ‘information overloading’ and ‘lack of attention to detail’ placed under the theme *understanding problem specification* were transferred to this new theme since the data is related to the incidents that occurred while interpreting the case study.

Further, two codes (e.g., codes – ‘summarise the understanding’ and ‘crosschecking understanding with physical view’) placed under the theme *steps in process* appeared to be specifically used to verify the process despite being a sort of tactic. Therefore, a new theme was introduced, named *verification of process*, and the two codes were moved under the new theme. Interestingly, this was a theme that was let go of in an earlier phase of the analysis, but upon reflection on the data again, having them under a new theme seemed to explain the data better.

When revising *understanding the problem specification* theme, the researcher realised that some of the codes are clearly related to how the problem is presented and some about the domain. Due to this clear distinction, the theme was divided into two to better explain the data as *problem presentation* and *understanding the problem domain*. The theme *tools* didn’t go through any changes since the data appeared to be mapping with the themes correctly. However, upon revising the *cognitive process* theme and checking against the existing data, the researcher decided to remove it as it didn’t fit the research question. But a couple of codes under it were about recalling what the students did prior to the session, and these were placed under a new theme, *problem recall*, along with another code that described the student’s ability to remember and recall what they were taught which was earlier placed under the *fragile knowledge* theme. The remaining code, ‘translating word by word’, under the *cognitive process* theme, was checked along with the data again and moved under the *interpreting case study* theme as it discussed how some participants used it as a tactic to interpret the case study.

Throughout this revision phase, two codes ('considering the design standards' under the *fragile knowledge* theme and 'not following a systematic approach' under the *processes and strategies* theme) couldn't be placed under an existing theme. Upon further inquiry with the data, it was realised they didn't warrant a new theme, and it was decided to drop the two codes from the analysis. The discussed changes to the themes and codes during this refining stage are presented in Figure 7.13.

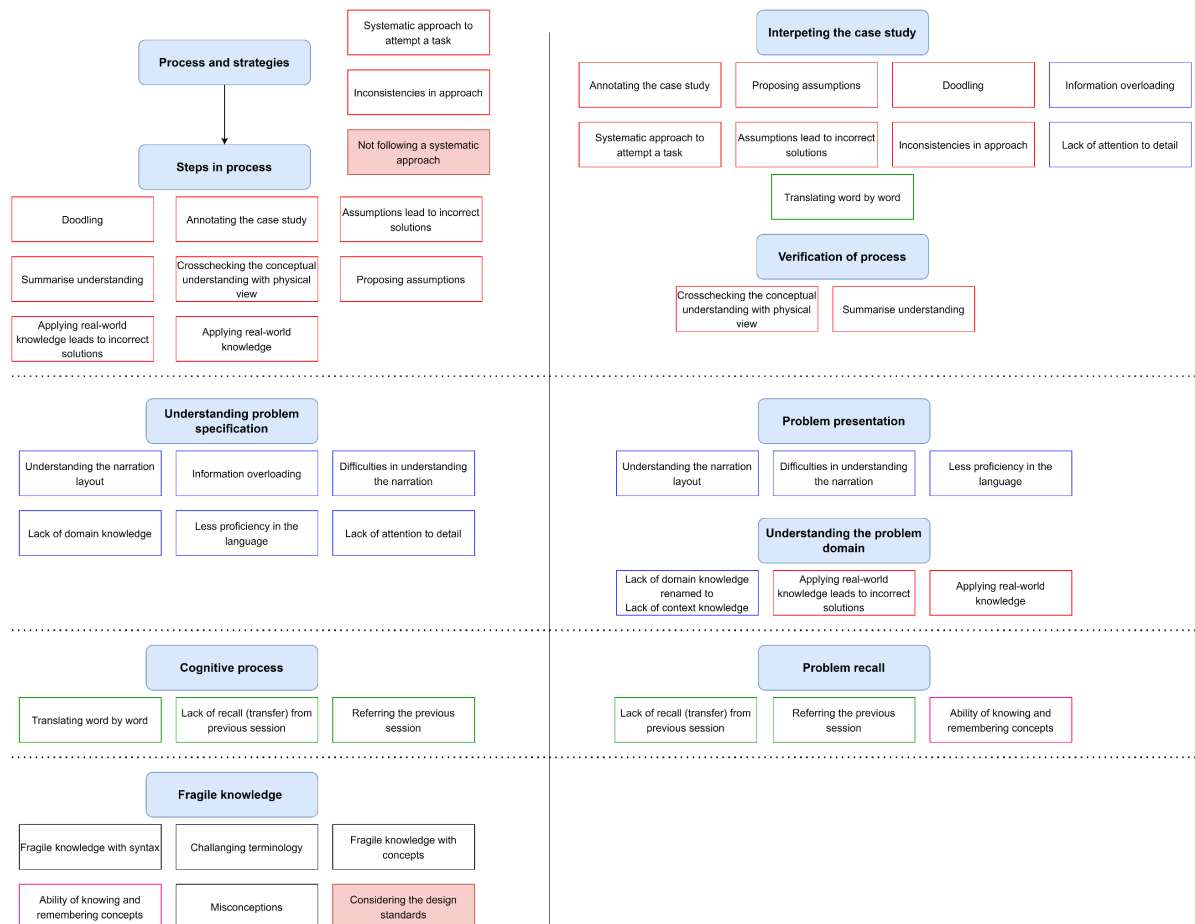


Figure 7.13 Refined themes.

Note. The right side column indicates the refined themes, where rounded rectangles represent the themes while the rectangles represent the codes under each theme. The outline of the rectangles is colour-coded to indicate the theme the code is transferred from, while the dropped-out codes are indicated using red-coloured rectangles.

Once there was a noticeable distinction between the related ideas, the theme names were finalised. The finalised thematic map is provided in Figure 7.14. To maintain a close link between the themes and the data, a theme definition was written for each theme, backed up with participant quotes (See Table 7.4).

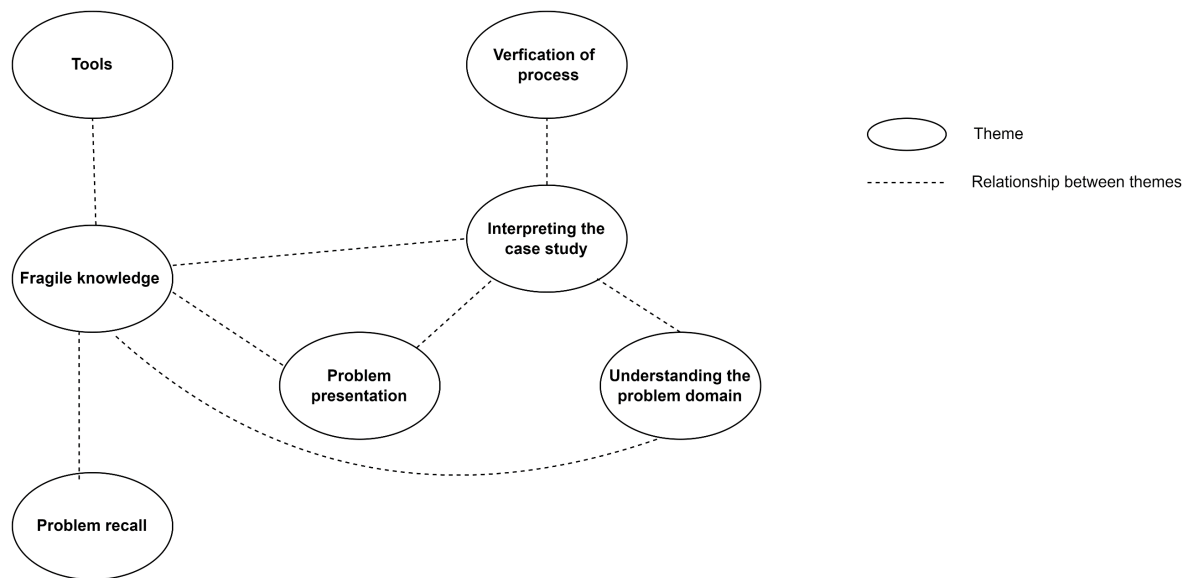


Figure 7.14 Finalised thematic map.

Phase 6 – Analytic Narrative

This final phase, which Braun and Clarke call “writing the report”, provides a written discussion for each theme. Themes are illustrated through participant quotes. Even though this was the final phase of the process, the researcher still revisited the earlier phases and data to ensure the ideas being presented closely represented the data while addressing the relevant research questions. The analytic narrative is presented in the following section, 7.8.

7.8 Thematic Analysis

Writing up each theme brought further clarity while articulating the ideas backed by the participant quotes. This was yet an iterative process while revisiting earlier phases to ensure that themes were consistently presented to address the research objectives. As per Byrne (2022), an important task in this phase is deciding the order in which the themes are presented. The themes must be connected logically and meaningfully to explain the story. Where relevant, each theme should be built upon the previous theme while reporting their own individual narrative. It was decided to start presenting the themes with the theme, *interpreting the case study* first since it was clear what the participants did to interpret the case study set the foundation for how their model was developed.

Further, this theme was also influenced by most of the codes categorised under this theme. Hence, this was considered a theme that could provide an understanding of the narrative that will be followed. Discussing *problem presentation* and *understanding the problem domain* appeared to be sensible after the first theme since these two themes introduced positive and negative practices of the participants

related to the problem while interpreting the case study. The *verification of process* theme was presented next, which explained some of the higher cognition levels of participants that emerged during the interpretation of the case study. A theme related to the participant's fragile knowledge was presented next, which was closely related to a few themes: *problem presentation*, *understanding the problem domain*, and *interpreting the case study*. This theme was then followed by the themes 'problem recall' and 'tool'. At this stage, the links were made to the existing literature to further explain the patterns identified in the data.

The overview of the identified themes and example data extracts from the participants are presented in Table 7.4.

Table 7.4 Overview of themes.

Theme name	Theme Definition	Example data extracts
Interpreting the case study	This theme explores various tactics applied by novices to analyse the case study, including the scenarios where these tactics were not used consistently, leading to incorrect solutions.	<i>"So I've made my assumption like if a course is running, it should have it should, it should have generated at least one program."</i> (Jackie) <i>"However, there can be few trainees [<u>underline and then circle</u>], sorry, I'm circling the process or relationship that I'm thinking, few trainees who are not referred, not referred [<u>underline</u>] by an employer [<u>underline</u>]." (Casey)</i>
Problem presentation	This theme examines how novices understood the problem/task specification and the difficulties they faced during the process.	<i>"Okay. Uh, because my mother tongue is not English. I'm kind of struggling to understand what this sentence means."</i> (Jackie) <i>"Then this is where it starts to play games on me... where it says while some programs can have more than one instructor assigned to them based on the number of trainees allocated for the program."</i> (Casey)
Understanding the problem domain	This theme explores the instances in which novices applied their existing knowledge to understanding the case study and the difficulties faced while doing it.	<i>"So in this, a program may have no trainees, it's an assumption like in the domain or in real life some programs may not have people, so it's an optional participation program."</i> (Sam)

		<i>"I tend to think about entities like programming wise and qualification name can be changed later on and if it happens, how can I say that many records should be fixed if participant has qualification name as it's attribute." (Jackie)</i>
Verification of design decisions	This theme captures how novices approach verifying their process and/or design decisions.	<i>"Usually when I think about a relationship, between two entities or two tables, I imagine like extra records registered in those tables and just try to imagine how each record relate to the other. So in program, table and trainee table there should be one program like p1 and p1 can have three trainees, t1, t2, t3. It is also possible that t1 can be in charge of not only p1 but also p3." (Jackie)</i>
Fragile knowledge	This theme explores the difficulties faced by novices due to their fragile or incomplete knowledge in three areas: terminology, conceptual knowledge and syntactic knowledge.	<i>"One to many, where we have to create the new relation and get the two primary keys and put it in there." (Casey)</i> <i>"Is it a composite attribute or a composite primary key? It's so similar and some of the terminology and stuff gets lost in translation." (Eli)</i>
Problem recall	This theme describes the novices' ability to transfer their existing knowledge and the difficulties they faced during problem-solving.	<i>"So this is many to many, I have to have an intermediate table, which should have traineeID and qualificationID." (Jackie)</i> <i>"TraineeID, uhm... Oh my gosh, I can't read my own writing, name. OK, let's read what is this here." (Casey)</i>
Tool	This theme explores the novices' experience in using a CASE tool to develop the models.	<i>"I might do this the wrong way. I can't remember how to do it." (Morgan)</i>

7.8.1 Interpreting the case study

A participant's approach to interpreting the case study, their tactics, and how they used them are encompassed in the theme of *interpreting the case study*.

When examining the participants' efforts in interpreting the case study, several tactics they used to help build an understanding of the presented problem were identified. It is important to differentiate between

tactics and strategies before discussing these observations since these terms are often interconnected but possess distinct meanings. Tactics can be referred to as specific actions or techniques learners use to achieve an immediate outcome as a part of a problem. For example, tactics in interpreting the case study could include actions such as annotating the case study, doodling, performing noun-verb analysis, etc. On the other hand, a strategy can be defined as broader plans or approaches that guide the overall problem-solving process. For example, in database modelling, applying a top-down approach to understand the problem holistically can be considered as a specific strategy that can be applied to achieve the overall outcome of a problem.

Several tactics were identified under this theme: annotating the case study, doodling while reading, making assumptions, and translating the case study word by word into modelling constructs. While some of the participants applied these approaches successfully, others did not. Thus, this theme includes both effective and ineffective applications of these tactics. Some of the tactics observed were specifically taught in class, while some might have been developed independently by the learners to cope with the task.

Annotating the case study while reading the case description was a tactic modelled by instructors in class when conducting noun-verb analysis. This form of analysis is a tactic commonly taught to novices to help them break down the modelling process into manageable chunks while also building an understanding of a case study. During class, the instructor worked through simple case descriptions, underlining verbs that map to relationships and colours highlighting nouns that might represent entities or attributes. Even though there were no explicit guidelines given on which annotation should be used for a given construct, it was emphasised that each construct needs to be annotated distinctively, making it easier to develop business rules later in the process. All except Blake applied this technique when reading the case study, albeit in their own way. The participants adopted a number of different annotations, such as underlining, rectangles, brackets, and circles, to differentiate between different constructs. Some participants, for example, Sam, Morgan and Eli constructed their own annotation systems, which they used consistently throughout the task, commenting on their systems out loud. The annotations they developed and related verbalisations are provided in Table 7.5.

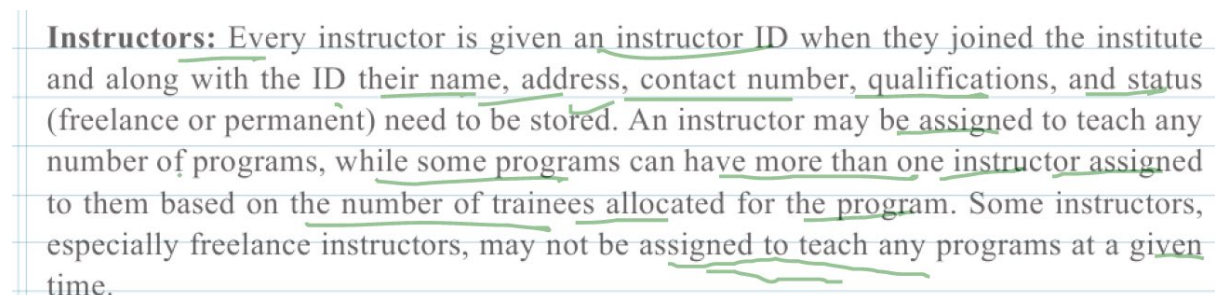
None of the students mentioned the term noun-verb analysis to explain what they were doing. However, one participant, Eli explained why the process was followed:

this is sort of like something to refer back to and helping with, I guess, framing those business rules and stuff and understanding relationships between things. [Eli]

The participants did not explicitly discuss how they distinguished which nouns represented entities and which represented attributes. However, Sam mentioned: “So I think in this sentence every noun is an

entity, because oh I'll continue reading.” As Sam identified the nouns as entities while reading the preamble to the task, they were correct in this instance, but they didn't give any reasoning for this decision they made. However, this was an indication of their knowledge that nouns could be potential entities.

Two of the participants, Casey and Jackie, did not annotate consistently. They started mixing up the annotations they initially defined as they proceeded with the case study. This introduced inconsistencies that led to confusion. Casey was the most inconsistent with their annotation, who was the lowest-performing participant based on their performance in the think-aloud study. After a certain point, there was no distinction in what Casey was highlighting, underlining all potential attributes, entities and relationships (see Figure 7.15). This meant they could not extract the correct information from the case study to develop business rules. This poor use of the tactic also impacted their ability in later tasks, where they attempted to create a conceptual model for the same case study.



Instructors: Every instructor is given an instructor ID when they joined the institute and along with the ID their name, address, contact number, qualifications, and status (freelance or permanent) need to be stored. An instructor may be assigned to teach any number of programs, while some programs can have more than one instructor assigned to them based on the number of trainees allocated for the program. Some instructors, especially freelance instructors, may not be assigned to teach any programs at a given time.

Figure 7.15 Incorrect use of annotations.

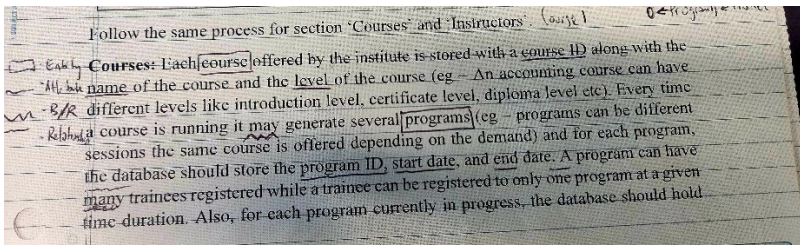
Interestingly, Blake, one of the middle-level performed participants in the study, did not make any annotations. However, they made a couple of errors in their solution, which could arguably have been avoided if the tactic had been used. For example, they identified *carrerAcademy* as an entity which was the system or used both the nouns *Trainee* and *Employee* as entities. These errors could have been identified if a proper noun-verb analysis had been conducted since, in a noun-verb analysis, they were instructed to remove the nouns that represent the entire scope of the system and to remove duplicates. During the retrospection, Blake agreed that they could have avoided the errors if they had annotated the constructs in the case study description. But also mentioned that they didn't feel the need to annotate since:

I didn't bother to, you know underline the things because the case study was sort of familiar and it was easy to identify the entities and attributes the way it was given. But as you explained, it would have been easy to figure out my issues you know where I used trainee and employee both if I followed the approach. [Blake]

The lack of consistent and systematic annotations hindered participant progress. In Casey's case, they were unable to identify the entities, attributes and relationships correctly, and the lack of a system of annotations hindered their ability to progress through the tasks in the study and led to misunderstandings of the problem domain. For example, while writing in business rules in Task 2, they found it difficult to identify the relationship between Employer and Trainee from their solution in Task 1.

Employer may... no, that doesn't sound right, it should be trainee may. Hmm... employer... I think employer only referred... hang on a minute. It's hard to get it from here [pointing the mouse to the solution of Task 1]. I'm going to have to read this whole thing again. [Casey]

Table 7.5 Annotation systems developed by the participants.

Examples from the participant's workings	Participant's verbalisation
 Follow the same process for section 'Courses' and 'Instructors'. (Casey) <u>Courses</u>: Each course offered by the institute is stored with a <u>course ID</u> along with the <u>name</u> of the course and the <u>level</u> of the course (eg - An accounting course can have different levels like introduction level, certificate level, diploma level etc). Every time a course is running it may generate several <u>programs</u> (eg - programs can be different sessions the same course is offered depending on the demand) and for each program, the database should store the <u>program ID</u>, <u>start date</u>, and <u>end date</u>. A program can have many trainees registered while a trainee can be registered to only one program at a given time duration. Also, for each program currently in progress, the database should hold	<i>So the square boxes will be an entity, and then the underlined will be an attribute. [Sam]</i>
<u>Trainees</u>: It is necessary to store details about all the trainees who participate in the courses. Each course participant is identified by their <u>trainee ID</u> and this is stored along with their name (composed of first name and last name), date of birth, gender, address (composed of street number, suburb, city, postcode), contact number. Along with these details the previously acquired qualifications of the trainees should be stored in the database. Since most of the time these trainees are referred to the course by their employers it is also necessary to store the details of the <u>employers</u> including the name of the company, address, and contact number. Each employer can be uniquely identified using the <u>employerID</u>. An employer can refer many employees to a training course and a trainee can be only referred to a course via one employer. However, there can be few trainees who are not referred by an employer.	<i>Okay. Uh, employer can refer many employees for a training course, and a trainee can be only referred to a course via one employer. So that's a relationship, I'll indicate it using R.. [Morgan]</i>
<u>Trainees</u>: It is necessary to store details about all the trainees who participate in the courses. Each course participant is identified by their <u>trainee ID</u> and this is stored along with their <u>name</u> (composed of first name and last name), <u>date of birth</u>, <u>gender</u>, <u>address</u> (composed of street number, suburb, city, postcode), <u>contact number</u>. Along with these details the <u>previously acquired qualifications</u> of the trainees should be stored in the database. Since most of the time these trainees are referred to the course by their <u>employers</u> it is also necessary to store the details of the <u>employers</u> including the <u>name of the</u>	<i>So I guess I'll put like an asterisk to denote some sort of a key sort of traineeID is the key. [Eli]</i>

Another tactic some participants (i.e., Jackie, Morgan and Eli) used was expressing their ideas using doodles or symbols while interpreting the case study. The symbols they used here were the model elements and parts of the notations taught for developing ERDs. One of the three participants used a slightly different set of symbols, more like UML class diagram elements. Later, that participant revealed that they had completed the object-oriented analysis and design course that covered UML in the previous semester. This influenced their doodling style.

Morgan, Jackie and Eli all three doodled while analysing the case study without being required from the task or being prompted by the interviewer. Also, it's worth noting that this was not a tactic utilised by the instructors. Jackie further used doodling as a mechanism to summarise their actions. They doodled while summarising each paragraph of the case study.

So just to summarize what I got, so participant, participant is an entity and the primary key is traineeID? TraineeID is not a primary key because, if the primary key of participant is there traineeID that will be duplicated. So, I guess participants should have their own primary key. Like a surrogate key or something. So I just give participantID or traineeID.. Okay. traineeID is something like participantID. So traineeID is a primary key, and then there are attributes, date of birth, gender and address, contact number... [Jackie]

Jackie's step-by-step doodling of their verbalisation is given in Figure 7.16. Jackie, an ESL student, in particular, found it easier to draw ERD elements when they found it difficult to express an idea in English. Either way, it was observed that participants tended to draw in order to visualise their thinking when interpreting the case. However, the practice of doodling became less noticeable once the think-aloud sessions transitioned from in-person to online mode. This shift can be attributed to the challenges posed by the virtual setup, which may have made it more difficult for participants to engage in doodling, unlike in a pen-and-paper environment.

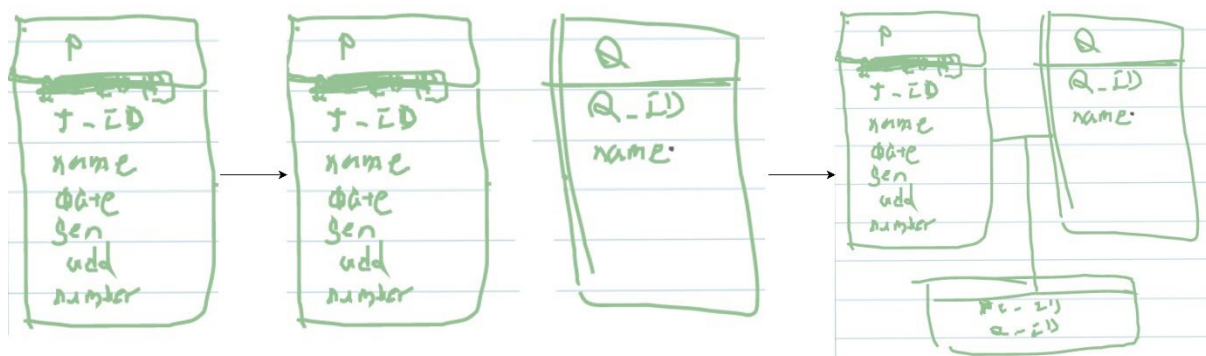


Figure 7.16 Step-by-step doodling of Jackie.

The participants also tended to make assumptions when interpreting the case study. This strategy was explained in class as a useful way of coping with missing case study information. Students were instructed to document any assumptions made. Such assumptions need to be made carefully and appropriately. When considering the case study, all but Blake managed to apply this tactic to arrive at a sensible interpretation of the case study:

Uhm, each course must generate several, so that's a one we don't even know what several means. So let's go with one to many and that's it. [Casey]

Each program currently in progress can have many allocated classes. And each class should belong to one program only. So a class can only belong to one program. Each program currently in progress can have many allocated classes. So I assume a program has to have at least one class and they can even have many classes, so one to many. [Morgan]

However, in some cases, the participants, Jackie and Morgan, made assumptions that led to incorrect interpretations of the case study. During the retrospective interviews, several reasons behind making incorrect assumptions emerged. The main reason appeared to be a lack of attention to detail in the case study that made them go beyond what was stated in the case study. Both participants stated below admitted during the retrospective interviews that they missed the lines explaining the cardinalities in the case study and hence, tended to assume based on their existing knowledge of the domain.

I think there can be a trainee who doesn't have any program because they just started. They just start working for a company or something. So a trainee may have one program. [Jackie]

So I'm going to say, at the very least, a program must be associated with that one course and a course. You can have multiple courses that do the same program. So I'm going to, I'm just going to assume that it's one-to-many, but that could be wrong. [Morgan]

Jackie made an incorrect assumption about the cardinality between *trainee* and *program*, where they assumed that there could be trainees without allocating to a program. But in this instance, they admitted that it was rather due to not understanding the sentence correctly, being an ESL student. During the retrospective interview, Jackie explained how they would use an online dictionary to either search synonyms for the terms or translate them into the mother tongue if it's difficult to understand. However, in this scenario, Jackie explained they tended to translate the sentence in their mind to make sense of it, but this could have led to the incorrect assumption.

I think I missed it or rather I wasn't sure what it meant. Yeah, like language is the main, like problem for me to understand. [Jackie]

Interestingly, both the participants who made incorrect assumptions were the ones who performed well in the study. It seemed that maybe they were too confident in their ability to make decisions, which could have led them to this issue.

Another tactic applied by the participants while interpreting the case study related to how they translated the case study into database modelling constructs. Casey was seen using this tactic repeatedly while interpreting the case study. However, Casey tended to translate word-by-word without attempting to build an overall meaning of the interconnected parts. Batra and Antony (1994) explained this observation as a 'literal translation', where novices literally translate the textual description of a problem

into the constructs of an ER model without reflecting on the overall model or alternative solutions. In the following example, Casey struggled to interpret the relationship between the entities (*employer*, *employee*, and *course*) and appeared unsure about how the three entities needed to be connected, failing to grasp the overall context.

An employer can refer many employees to a training course and a trainee can be only referred to a course via one employer. I'm thinking that this is tongue twisting, as only it's got like two entities in there which is course and employers. What about employee? So that's kinda like catching me off guard. [Casey]

In contrast to Casey, Jackie tended to translate sentence by sentence. After reading each sentence, Jackie processed and analysed the information to transform it into constructs before moving to the next sentence in the case study. Each time Jackie identified the constructs, they questioned how that related to the overall case study. For example, the following extract indicates how Jackie questions the relationship of the *Course* entity to the overall case study when analysing the trainee section of the case study, where the course was only mentioned as a part of a sentence.

Also now I'm thinking whether a training course can be an entity. But there is no information related to it. Yeah. So I'm not sure, if a training course can be an entity or not or how it fits to the case study. [Jackie]

Casey and Jackie both used different approaches in translating the natural language case study into database modelling constructs⁶.

Despite their level of performance, three participants showed a lack of attention to detail when trying to interpret the case study. In some cases, participants ignored the requirements when not understanding the related text and, rather than trying to understand the information, skipped over it, deeming it not important. Casey opted to ignore requirements, deeming the information unimportant, as exemplified by their utterance:

⁶ As an aside, an interesting interpretation of these observations can be made using the SOLO taxonomy (Biggs & Collis, 1982). Casey could only deal with one element of the problem at a time interestingly, this maps to the SOLO taxonomy's unistructural level while Jackie was working at SOLO's relational level. The ability to abstract and holistically relate information is clearly important to understanding the case study.

Then however, there can be few trainees who are not referred by an employer. Well, they don't need to worry about that. Okay I think I'm done. [Casey]

Blake questioned whether *Employee* and *Trainee* were the same entity but decided not to explore this further, skipping over the necessary intentional analysis phase of noun-verb analysis, stating:

Are employees the same entity as trainees not sure, Yeah. Ah never mind. [Blake]

Through the retrospective interviews, it emerged that Casey, Blake and Morgan simply missed entire sentences in the case study, leading to incorrect interpretations. This suggests a lack of attention was given to the case study. Moreover, this might have been overcome if they had adopted a systematic process or tactic to verify their understanding of the case study. Only Jackie and Sam verified their work before moving from one section to the next in the case study description. These participants, who cross-checked their solutions, completed the tasks with the fewest number of errors.

It appears that information overload played a role in the poor performance of Casey, who struggled on multiple occasions when trying to understand and interpret the case study. From the researcher's observations and the retrospective interview, the cognitive load for this participant was likely due to inconsistent tactics and fragile knowledge. Casey struggled while going back and forth between Tasks 1 and 2. Task 2 was dependent on the knowledge built during Task 1. Because Casey failed to annotate the case study in a meaningful and systematic manner, any insights gained were not recorded, leading to confusion when they needed to refer back to them. For example, Casey struggled with figuring out what a session was when developing the conceptual model in Task 2. However, it was established that session was a noun that was not either an entity or an attribute during Task 1. However, Casey's inconsistent approach to presenting information led them to read through the case study again instead of referring to the knowledge gained from Task 1, which probably could have led to possible information overload as Casey couldn't figure out what the term session was for a long time.

That's different sessions what's different sessions. Where does that come from? [Casey]

The other main reason for information overloading was their fragile knowledge, and as they progressed through the modelling tasks, it overwhelmed the participants with too much information and concepts they could not comprehend. However, this was a rare occurrence, particularly when the participant performed poorly during the study.

It's all so overwhelming, you know, just trying to, you know, get your head around all that. [Casey]

7.8.2 Problem presentation

Undoubtedly, interpreting the case study is the most important step in designing a database, as it lays the foundation for any future modelling efforts. This theme presents how the participants understood the problem presentation while working on the case study text. The main difficulty participants encountered related to problem presentation was the struggle to understand the narration. It was evident that Casey, Jackie and Sam had some difficulties with how the sentences were structured, particularly when they were too long or connected using conjunctions. Interestingly, both Casey and Sam, whose first language was English, had difficulty understanding the problem presentation.

...every instructor is given an instructor ID when they joined the Institute and along with the ID, their name, what? Along with the ID, their name ... that makes no sense. [Casey]

So an instructor can teach or can be assigned, or may be assigned to one or more program. And some programs can have more than one instructor assigned to them. What I'm thinking now is like, if there's a program there should be corresponding instructor, but if there is an instructor, he or she may not be assigned to any program because an instructor maybe assigned to teach any number of programs. Uh, it is hard to understand this sentence. [Jackie]

Um, because there's so many sentences that describes, uh, one relationship it's hard to like combine it and then find out like their participation then, um, identify cardinalities. [Sam]

However, both Eli and Blake mentioned being able to see a pattern in how the case study was presented, which made it easier for them to understand the narrative. During the retrospective interview, Eli specifically mentioned this and explained how they noticed the way the information was presented in the case study, which helped them to understand the flow of the case study clearly:

Trainee, after that, would be the related attributes. And that's also because of the syntax and how the stuff is laid out. Because it's actually laid out in a very particular way that makes it relatively easy to identify these things, I think. [Eli]

Eli also commented on the case study presentation, stating, “This case study is already being processed somewhat so that it has information laid out in a particular way very useful for the learning process”.

As already mentioned, Jackie, an ESL student, sometimes struggled to understand the problem presentation due to their level of English proficiency.

Okay. Uh, because like my mother tongue is not English I'm kind of struggling to understand like what this sentence means. [Jackie]

Also, it was evident during the retrospective interview that Jackie tended to translate the English sentences into their mother tongue to understand them better, and this two-stage translation process could have made it difficult to comprehend how the problem was presented.

Jackie: *I think I misinterpret about like given time duration.*

Interviewer: *Okay. So it's related to that particular term, given there "at a given time duration"?*

Jackie: *I think I mistranslate it.*

7.8.3 Understanding the problem domain

Sometimes, participants had difficulty understanding the problem domain even though the case study was intentionally developed to avoid this additional complication, and some attempted to apply their real-world knowledge to make decisions when translating the case study into a conceptual model. While applying real-world knowledge can be an effective tactic for decision-making, if not used correctly, it can lead novices to incorrect solutions (Batra & Wishart, 2004). Morgan, Eli and Sam all applied their existing real-world knowledge to clarify the case study.

I'm currently thinking about, um, say programming one, there was comp500 and then ENC the engineering one. So I'm thinking of examples where you could have a programme. [Morgan]

Also each program currently in progress can have many allocated classes and each class should belong to one program only. Okay. So I think the programs refer to streams, and then classes refereed to the actual sort of like subjects. [Eli]

Some even used existing knowledge to make assumptions when interpreting the case study, as stated below:

So in this thing a programme may have no trainees, it's an assumption like in the domain or in real life some programmes may not have people so it's an optional participation, programme. [Sam]

However, not everyone managed to apply this strategy effectively. As an example, while Morgan was attempting to design the relationship between the *Course* and the *Program* entities, it was evident that they wanted to rely on their existing knowledge of a similar scenario.

Yeah, I was. I was thinking it kind of like a like a paper right? So you have a paper and you have multiple courses doing the same paper. So I think that's where I got that from, but it could be wrong in this context at least. [Morgan]

At least in this particular instance, it seems the participant gravitated towards familiar domain knowledge rather than the information provided in the case study. Morgan even commented on the same but did not change their decision. Another participant made a similar error to Morgan and explained

how relating the scenario to real-life knowledge made them extract incorrect information from the case study.

This is I couldn't... How can I say that? Extract the information correctly. I mean, it is not about like misinterpreting or miss translating. This is what I couldn't understand and I thought of a real life situation and there maybe trainee who doesn't have any programme, but this possibly, like this made me not picking the information. Okay. Yeah. [Jackie]

There was another instance where a participant struggled with not having enough exposure to the domain despite it being supposed to be familiar. The following example came up during the retrospection interview with Jackie on how they justified designing *Qualification* as an entity instead of an attribute.

I tend to think about entities like programming wise and qualification name can be changed later on and if it happens, like, I mean, how can I say that many records should be fixed if participant has qualification name as it's attribute. [Jackie]

In this scenario, Jackie thinks a qualification can be changed, and if it happens, many records would have to be updated if the participant has a qualification name as an attribute. While this seems reasonable, it is unlikely to be the case as even if the qualification name changes, all people prior to the change would have the original qualification (e.g., if BCIS changes to BSc in 2022, then prior to 2022, people still hold a BCIS). This highlights the importance of domain in the database modelling process.

7.8.4 Verification of design decisions

Verification or checking of design decisions was not explicitly taught as a process to the participants. However, it was a tactic the more successful participants were observed using.

Sam and Jackie demonstrated a tendency to summarise what they had just done when thinking aloud. Based on the interviewer's observation, it appeared to be an artefact of both the think-aloud process and the way the case study was presented in sections. Sam tended to summarise their findings at the end of each case study section by reading the relevant section of the case study to cross-check their answers, while Jackie summarised by doodling and coming up with a sketch.

OK so lets just summarise what I've got. So there is instructor entity and ID is the primary key and name, address, qualification and status are attributes. Okay Program is another entity. [Sam]

Another interesting answer verification tactic observed was when Jackie crosschecked their conceptual understanding of the problem by using instances of data occurrences to verify the cardinality between two entities. As indicated in Figure 7.17, Jackie was attempting a visual representation of data instances

to validate the relationships and cardinalities identified during the conceptual design phase. By sketching the physical database design, Jackie visually depicts the tables, their attributes and the relationships between them to create a concrete representation of how these constructs will be organised in the actual database implementation.

Usually when I think about relationship , between two entities or two tables, I imagine like extra records registered in those tables and just try to imagine how each records relate to each other. So in program, table and in trainee table there should be one program like program one ,program one can have three trainees , trainee one, trainee two, trainee three, but it is also possible that trainee one can be in charge of not only program one, but also like program three. [Jackie]

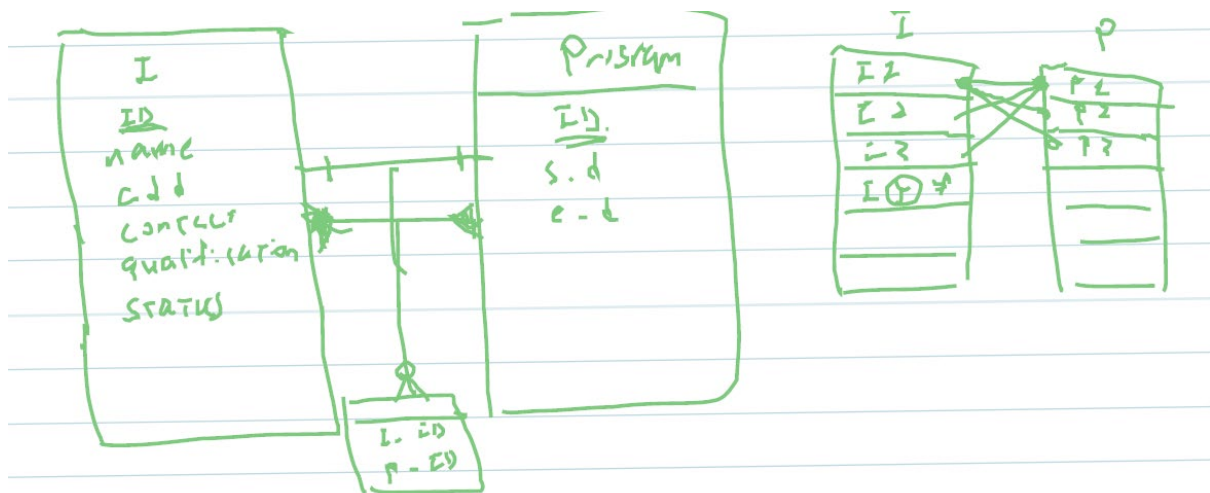


Figure 7.17 Jackie’s attempt to verify the conceptual design by sketching the database structure.

7.8.5 Fragile knowledge

Throughout the study, all the participants struggled with fragile or incomplete knowledge despite their differing degrees of success in both the final course outcomes and this study. This fragility was particularly evident in three main areas: terminology, conceptual, and syntactic knowledge. The terminology refers to the specific vocabulary and terms describing the elements and processes in the subject area that ensure effective communication. Conceptual knowledge refers to a deeper understanding of the fundamental concepts and principles within a subject. Syntactic knowledge often refers to understanding the rules and structures that govern the syntax or grammar of a language or domain.

Terminology

Fragile knowledge of terminology can be defined as where students struggle with understanding and correctly using the specific terminology associated with database modelling. Sometimes, they had a

limited vocabulary or encountered unfamiliar terms, which hindered their comprehension. This lack of familiarity with key terms can make it challenging for students to communicate effectively and grasp the underlying concepts.

Novice database modellers involved in our sessions struggled with the specific terminology used in the database modelling domain. In some situations, these challenging terms seem to be a cause of their fragile knowledge. As an example, some had difficulties with the terminology used in database modelling, in some cases confusing similar terms like composite attribute and composite primary key. All six participants had trouble recalling the exact terminology at least once, even if they managed to explain the underlying concept to a certain level.

Is it a composite attribute or a composite primary key? It's so similar and some of the terminology and stuff gets lost in translation. [Eli]

I am not sure what it's called. What I'm trying to do is, where you write the columns and then you do the arrows. [Casey]

Particularly, Casey, who had a relatively weak performance, was overwhelmed by hearing many new terms at the same time.

Casey: And the logical the conceptual. My gosh it's just trying...

Interviewer: I think, is it the terms?

Casey: Yeah, I think it's all about the name trying to distinguish, you know, like it's the meaning behind the terms. It's confusing.

The following is an example where a participant struggled to communicate effectively due to the confusion of terminologies. Eli initially mentioned that *qualification* was a composite attribute, but with the interviewer's prompt, they immediately realised they had used an incorrect term and went on to explain that it was a multivalued attribute and how to resolve it correctly.

Uh, I think also trainees previous qualification could be composite as well, and same with an instructor composite course qualification because... [Eli]

Conceptual knowledge

Fragile conceptual knowledge refers to a weak understanding of the fundamental principles and concepts in database modelling. The difficulty of comprehending the core concepts or theories can lead to misconceptions or shallow understanding that could affect a student's ability to apply knowledge in a given context to make informed design decisions and efficient database modelling.

Almost all the participants encountered difficulties with basic database modelling concepts. Sometimes, they could not simply recall what had been taught in class despite attending.

To identify a foreign key. Uh, I think it is in the lecture. I just didn't remember. I just thought the foreign key becomes the primary key from the parent. [Blake]

I don't think yeah no. No matter how long I sit here thinking about it I don't think I'm not going to pull it out of thin air really. [Morgan]

In the retrospective interview with Morgan, it was clear that they couldn't recall the mapping rules taught in the class to transform a conceptual model into a logical one apart from the one related to many-to-many relationships. Upon further discussion, it was revealed that Morgan's incomplete knowledge in this area may have been due to a lack of practice with similar exercises that required the application of all the mapping rules. Interestingly, this same issue was observed in the responses from the other three participants from Morgan's level 5 cohort. They could only recall the mapping rule related to the many-to-many relationships, which led to incomplete solutions, as illustrated in Blake's solution given in Figure 7.18. Blake, also studying the level 5 course, had an incorrect solution due to the unresolved multivalued attribute –qualifications and composite attribute –address as highlighted with a red box in Figure 7.18.

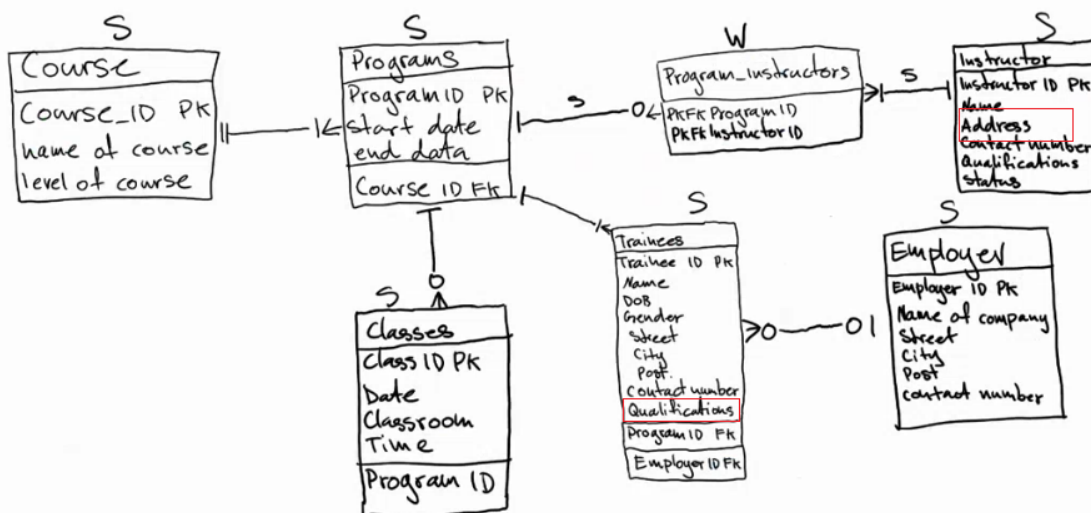


Figure 7.18 Incorrect solution due to fragile knowledge. The errors are marked in red colour.

Several other participants were confused when trying to differentiate between different concepts and terminology they had learnt. Some participants provided incorrect explanations for determining certain concepts, and others used incorrect terminology while explaining a concept. From the retrospective interviews, it was evident that many participants struggled with the abstract nature of some concepts. The following extracts portray the difficulties participants had due to fragile knowledge, particularly related to composite and multi-valued attributes:

If there's multivalued attributes such as address, which we even have like city, street, and country in one column in one cell, I personally think. [Jackie]

This extract demonstrates that Jackie incorrectly used the term multivalued attribute to refer to the attribute *address*, which was a composite attribute consisting of multiple components (i.e., street, city and country). Upon the interviewer's query, Jackie indicated: *"I mix up like multivalued attribute and composite."*

I get a feeling that there should be a composite attribute for course level because it talks about in brackets different levels like introduction level, diploma level. [Casey]

In this extract, Casey incorrectly associates course level as a composite attribute. They mistakenly believe that a composite attribute should represent the different levels within a course, such as the introduction and diploma levels. This indicates a fragile understanding of composite attributes and their usage correctly. It is important to note that Casey does not realise that composite attributes represent a simple attribute composed of multiple components rather than different levels within a course. The confusion demonstrated by Casey suggests a need for further clarification to understand the concept of composite attributes.

One of the participants, Casey, who didn't perform well, struggled throughout the sessions with their fragile knowledge. But, their struggles were most evident during the normalisation problem (discussed in section 6.3.5) when Casey attempted to transform the developed conceptual model into a logical model. From the beginning of the task, Casey showed incomplete knowledge of the concepts needed to complete the task, as evidenced by the below extract, where Casey was confused about the task itself.

That's right, so in order to transform them into a logical design, I have to start off with those primary keys and then break them down, like think I can remember. [Casey]

As the session progressed, it was evident how fragile their knowledge was, and it seemed the cognitive load placed upon their short-term memory did not allow Casey to grasp the concepts properly. The following is an extract where Casey was explaining how to resolve one-to-many relationships but instead discussed the mapping rule for many-to-many relationships.

Casey: OK, so that is going to be, I'm going to get the primary key from these two and create a new relation.

Interviewer: OK. Before you do that, look at what the relationship is. Is it one-to-many or many-to-many?

Despite the interviewer's intervention, Casey kept on repeating the same mistake *"One to many, where we have to create the new relation and get the two primary keys and put it in there."*

Misconceptions related to database modelling were a rare occurrence among the participants. However, it was mainly observed when they attempted to transform the conceptual model into a logical one, where several participants believed that transformation only required changing the many-to-many relationships. Some considered that the associative entity used to resolve the many-to-many relationship in this transformation is only required since it connects the entities as a bridge.

Going from conceptual to logical, it's just changing the many to many relationships.
[Blake]

It's only job is to connect entities, so it's simply references entities. [Morgan]

Jackie seemed to think it was unimportant when the interviewer questioned them about not applying the developed business rules in creating the conceptual model. Specifically, Jackie and Casey seemed to consider this step as only a component of the question rather than incorporating it into their own process.

I mean, well, I personally take business rules lightly or not regard or treat business rules important, I mean as important as the case study. [Jackie]

However, Jackie, one of the best-performing participants during the think-aloud sessions, did not experience significant difficulties by not considering the business rules to develop their conceptual model. On the other hand, Casey faced challenges in associating business rules with the conceptual model. Casey struggled to recall the knowledge developed during Task 2, where they were required to identify and document the business rules based on the given case study. Instead of relying on their acquired knowledge, Casey often had to refer back to the case study for guidance. Casey's difficulties indicate a fragility in their knowledge regarding the application of business rules in the context of conceptual modelling.

Syntactic knowledge

In the context of database modelling, syntactic knowledge involves understanding the correct symbols or elements of models. This includes proper placement of the symbols and adhering to specific rules for constructing the database models. For example, a dashed line should be used to differentiate strong relationships from weak ones in developing a logical ERD. Further, in database modelling, syntax can also be referred to as the grammar taught to express the business rules. This syntax governs the specific structure and format that should be followed when expressing the business rules using natural language.

The participants occasionally had trouble with the syntax needed for database modelling. In contrast to the participants who used the modelling tool, the visual paradigm, the participants' fragile knowledge of syntax was more apparent when they used pen and paper. This might be because when the

participants use the tool, they have an advantage since they can work around it to recognise the notations even if they can not remember them. Also, the tool does some of them automatically, for example, when creating a 1:M relationship, it applies the mapping by creating a foreign key in the parent entity.

And I think looking at it every time a course is running it generates several programs. So I think that's one to many. But I'm not sure about the symbols. [Blake]

Program must have one course. So let's say that just one dash or two dashes. I can't remember. Are you allowed to tell me? [Eli]

However, all the participants who used the tool struggled with using and handling it at certain times, which could have also contributed to their cognitive load.

During the think-alouds, it was observed that some participants faced difficulties in recalling or effectively using the taught grammar to express the business rules in database modelling (i.e., correctly defining participation and cardinality constraints using a relationship verb phrase syntax in formulating the business rules). These struggles indicate challenges with syntactic knowledge and its application in practice.

I forget when to use may and when to use must. Can you explain it? [Blake]

Trainees may not be referred by employer [business rule written by Casey]. Well, how do I do this one? It seems something is missing. But I can't remember what... [Casey]

7.8.6 Problem recall

Developing a well-structured database model requires the students to transfer knowledge effectively from one phase to another, from interpreting the specification, developing business rules, and creating the conceptual model to transforming it into a logical model. This theme discusses how the participants handled this process and the difficulties they faced while recalling the progress made on the modelling problem in a previous think-aloud session. Morgan, Eli and Sam often tended to refer to the solutions they had developed in the previous session and attempt to build on that knowledge to progress on the current task at hand:

So I'm just going to use what I did in previous session in here. [Morgan]

So this is where I look at the business rules. [Eli]

And then in my old notes of, from the previous, um, exercise. I had a small sort of diagram going on which I'm going to use. [Sam]

I'm just basically taking what I've gotten already down well at least the basic entities, attributes from the session two into visual paradigm here. [Eli]

However, Casey mentioned an inability to refer back to their previous solution and transfer that knowledge effectively. One of the main causes of their inability to refer to their own solution was the ad-hoc manner in which they approached the previous task. For example, Casey had no clear system for annotating their noun-verb analysis and mentioned:

OK qualification. Why do I have qualification there and the employer underneath?
[Casey]

Then later, Casey attempted to refer back to their answer to Task 2:

The one from the second session that we had, because if I look at, you know this one. Where I'm scrolling through, If I look through that one, it's too messy. [Casey]

Other participants, at least once, did not refer back to the findings of their previous session. This led to incorrect solutions on the task at hand despite their answer to the precursor task being correct. For example, both Jackie and Casey had instances where they had difficulties in developing the conceptual model (Task 3) even though the required information was already identified in their previous solutions to Tasks 1 and 2, but they did not seem the need to refer back. For example, in this extract, Jackie was uncertain about whether the *qualification* is an attribute to the *Trainee* entity even though in Task 1, they decided that it was an attribute and could even be a multi-valued one.

Qualification of the trainee, um, should be an attribute of trainee or not? Because I don't know the relationship between trainee and the qualification. Should I check in the previous answer ah... never mind. [Jackie]

7.8.7 Tools

The way participants utilised the modelling tool was observed due to the change from in-person sessions where the participants used pen and paper to online sessions. Only three participants: Morgan, Casey and Eli, used the tool during the sessions, and all of them faced issues. The main reason appeared to be their lack of practice in handling it despite it being the course's prescribed tool which was used in tutorials and assessments.

Most problems arose when trying to add relationships correctly and adding the cardinality on the wrong side of the relationship and/or not knowing how to edit a relationship and/or cardinalities that had already been added:

So it says each employer, no, the crow is not where I want it to be. I want the each. I want that each thing. How do you change that? [Casey]

I can't exactly remember which way this goes, the relationship. Isn't there a zero to one relationship in visual paradigm? [Morgan]

When resolving a many-to-many relationship in the logical model, Casey added a new row in the bridging entity to add the composite primary key. Casey was not aware that once the relationships are added, the tool would create the primary key accordingly, “*Okay, so I've got there uh instructor and program. So many-to-many create a new relation with the primary keys ...perfect. I'm going to insert a column here. Yep.*”.

Despite some initial struggles, Morgan and Eli handled the tool efficiently. For example, they inserted the primary keys as soon as the entities were created. As a result, when the entities were linked together via a relationship, the foreign keys were generated correctly. Also, all three of them considered how the model would look once designed. Initially, it appeared that the participants were concerned about the readability of the model. However, it became clear through additional discussion and observation that the consideration of model layout was caused by the lack of knowledge or unfamiliarity with the tool. Given how user-friendly the visual paradigm tool is, it stands to reason that if they had used it properly, the following doubts might not have surfaced:

Oh, I think this is going to look messy now. So let's just move it around. [Casey]

Is there enough space for my links? Maybe I need to move it down so the links can reach. [Casey]

7.9 Discussion

The following sections, 7.9.1 and 7.9.2, are used to connect the identified findings of the thematic analysis to the existing literature in database modelling or CSER in general. The discussion is mainly divided into two sections: common tactics identified by the participants and the difficulties and challenges they faced.

7.9.1 Common tactics applied by the participants

The participants applied different tactics when attempting the tasks, and these tactics were split between taught processes and the ones they developed by themselves. As expected, most of these tactics were applied to interpret the case study as it laid the foundation for the model being developed. Annotating the case study is a taught process on par with the taught noun-verb analysis. This is a tactic designed to guide learners in systematically analysing and understanding the requirements in a provided problem description to identify the modelling elements. However, it was noted that some participants struggled with this taught process due to their unsystematic use. This suggests that while the participants were

aware of the instructed tactics, they may have found it difficult to apply them consistently. While there has been literature discussing the difficulties faced by novice database modellers, to the best of the researcher's knowledge, there was none that discussed the use of noun-verb analysis or annotating as a tactic used by novice modellers. However, several database-related textbooks introduce noun-verb analysis as a tactic or a process used in analysing the requirements, including Hoffer et al.'s (2016) textbook used in the courses in which the participants were enrolled.

Making relevant assumptions was another taught technique to the novices to interpret a case study precisely. However, we observed that some participants made wrong assumptions that led to issues in the model, which led to incorrect, missing, or redundant modelling constructs. This observation maps with the findings reported by Hussain (2016). Hussain, in their analysis, identified frequent ER modelling errors made by students. Then, they had gone in to identify the causes of these errors using Ishikawa's fishbone diagram (Ishikawa, 1976), where they concluded that making a wrong assumption or missing necessary assumptions were one of the main reason for the majority of errors made by the students which led them to have incorrect models. Further, they have categorised these two reasons as having a 'severe' impact on student generated models since such assumptions can lead to major changes in the relational model.

The participants in the study demonstrated a tendency to rely on their existing knowledge and real-world experience during the case study analysis. While this tactic could be advantageous if applied properly and in a relevant manner to gain valuable insights and help participants make informed decisions, it was observed that some participants struggled to utilise their real-world experience effectively. In some instances, participants made decisions based on their experience or what they knew rather than focusing on the information provided in the case study, which led to incomplete or inaccurate database models. In a study conducted by Batra and Wishart (2004) with novice database designers, they found instances where the designers misinterpreted the problem and deviated from the expected design, which was similar to the above-explained observation from this study. Further, they stated: *"that perceptions, memories, thoughts, and actions have a tendency to err in the direction of the familiar and expected."* (P. 401).

Doodling was a tactic adopted by several participants, mainly during the in-person sessions, while using pen and paper to attempt the tasks. Even though doodling is not a tactic that emerged in the literature associated with learning database modelling, ample researchers have discussed doodling in novice programming (Clear et al., 2009; Khomokhoana, 2022; Lister et al., 2007). Doodling in programming can be seen as a form of visualising the thought process to explore and experiment with ideas and potential solutions. Unlike tracing, doodling may not be a completed or formalised process that often stops after going through a couple of iterations once a decision is made on the coding solution.

Among the novice database modellers of this study, it was observed they used doodling as a spontaneous tactic. While not explicitly prompted to do so, they used the taught notations as a means to visually check their understanding of the identified design elements rather than relying on written explanations. In this doctoral study, the doodling done by participants as a technique to note visually key aspects of the case study often produced partial diagrams. These participants performed better on the tasks. These doodles can be considered to be a form of visualisation. Such visualisations are believed to lower the cognitive load by reducing extraneous cognitive processing, leading to better comprehension (Schwamborn et al., 2011). Indeed, Katz (2018) showed that visualisation designed to support the conception of database modelling schema resulted in learners making fewer errors and producing higher quality solutions than the control group, who were not supported with visualisations. They hypothesised that the visualisations reduced the cognitive load imposed by such tasks.

In this study, only three participants who were among the higher-performing students attempted doodling as a part of their problem-solving process. This observation suggests that the use of doodling could be a more personal preference or an individual learning style rather than a commonly adopted tactic among novice database modellers. However, the fact that higher-performing participants were more inclined to use doodling could indicate that they saw the benefits of visualising their thinking process as a way to reduce the complexity of the problem. On the contrary, it's worth noting that among the other three participants who didn't use doodling, there was only one who struggled and performed weakly, whereas the other two performed moderately well, which could suggest they may have employed other cognitive tactics or preferred different modes of thinking.

Some participants were observed using the tactic of literal translation of the case study in identifying the modelling elements. Using literal translation is often described as an error-prone strategy utilised by novices to reduce the cognitive strain of working on database modelling tasks (Batra & Antony, 1994). In Batra's study, they found that novices tend to translate the domain semantics literally into database structures, causing literal translation errors. This same issue was observed among this study's lowest-performing participants, who tended to translate the domain semantics erroneously.

Researchers have suggested that literal translation errors among novices are due to novices struggling to relate the different parts of a problem, and as a result, novices tend to opt for a literal translation, where they attempt to directly match the problem's elements with their solution without fully understanding the underlying concepts. Further, they explained how experts approach the problem from a top-down perspective, utilising their experience to develop a holistic view of the problem domain (Dinesh Batra & Davis, 1992; Venable, 1996). However, in a recent study, Rosenthal et al. (2022) claimed that in their study, which was conducted with a range of modellers from novices to experts, the majority of modellers read and annotated the entire modelling task before starting to model, suggesting

that they all used a literal word by word translation. As they observed, none of the medium-experienced and expert modellers in their study adopted a modelling style indicating a more holistic processing of the case description. It should be noted that in Rosenthal's study, all the expert participants were postgraduate students and not industry professionals. Thus, their experts could be considered to be closer to novice students than an expert who is an industry professional who has been working in the area for several years. Thus Rosenthal's results of not seeing an holistic approach to analysing the problem description among the expert modellers is problematic. In Batra's study four of the five experts studied were working professionals. Batra's position is that experts start analysing a problem description using a top-down approach rather than a bottom-up literal translation. Thus it may be reasonable to assume if our participants analyse the problem description top-down that, they are closer to an approach adopted by an expert. However, no further studies on the nature of expertise was available and it remains unclear what a more expert approach to a database design is.

Only two of the participants even attempted to verify their solutions, and a failure to crosscheck or verify answers to tasks meant errors were propagated onto subsequent tasks. Similar issues were noted by Venable (1996). In his work, he also noted the ability to judge the quality of models adequately was a key challenge for novices. He elaborates on this further, explaining that novices fail to apply even basic quality control heuristics, such as examining the data model for completeness. In contrast to Venable's findings, two participants in this study did attempt to crosscheck their answers before proceeding to the next task by summarising their findings before moving forward. This activity helped confirm their current understanding of their partially developed model. One participant, one of the best performing in the study, used a crosschecking in which they visualised the data instances to validate the design decisions they made related to the relationships and cardinalities.

7.9.2 Difficulties and challenges faced by the participants

The difficulties and challenges faced by the participants of this study map to those reported by other researchers (Batra, 2007; Hussain, 2016; Katz & Shmallo, 2015; Sedrakyan et al., 2014; Venable, 1996). Regardless of their performance in the sessions, the study participants exhibited fragile and/or incomplete knowledge. The majority of issues arose as a result of fragile knowledge of database modelling concepts and terms. There was also evidence of a lack of knowledge of model elements/syntax (Rosenthal & Strecker, 2019). The cases analysed in this study support the conjecture made by Hussain (2016) that the students mainly make errors due to insufficient knowledge of either ER concepts or problem domains and not being able to apply the knowledge correctly despite having the required knowledge. As discussed in section 6.3.1, the case study was deliberately chosen to be in a familiar domain for the participants. The intention behind this decision was to minimise the potential issues arising from the problem context or domain, allowing the researcher to explore how the novices

learnt without the added complexity of working in an unfamiliar domain. Hence, only very few participants struggled with the lack of context or domain knowledge related to the provided case study. There were some instances where participants struggled with the domain where they made invalid assumptions which led to incorrect solutions.

Another main struggle faced by these novices was the lack of proficiency in the language, which led to struggles in understanding the narration. Interestingly, in this study, even the participants whose first language was English struggled in certain scenarios. They struggled mainly with the longer sentences and conjunctions used in the sentences in the provided case study, which appeared to add a certain level of cognitive strain on the participants and the challenging terminologies. However, it appears that understanding the case study was particularly problematic for the ESL participant. These observations somewhat agree with the recent work of Mason and Seton (2021) identified during the literature review. As per Mason and Seton, the difficulty faced by NESB students has been linked to learning materials, assessments, and presentations. The argument is that the way such materials are typically presented increases the load on the working memory of NESB students. Mason and Seton give a set of guidelines for presenting assessments that address aspects of cognitive load and show through an empirical evaluation that this type of treatment levels the playing field and benefits all students. However, the authors admit that further investigation is needed to confirm these findings.

Another main issue identified was not having a proper verification process to validate their design decisions. There were instances where the participants started a task from scratch rather than transferring the acquired knowledge to the new task. This was a common problem identified particularly among the mid to lower-performing participants, who probably did not recognise the relation between the tasks. This lack of transfer appeared to occur due to inconsistencies they made while following the taught processes when completing the tasks. For example, one participant struggled to develop the conceptual model since their interpretation of the case study was not consistently done. This affected them like a 'domino effect' as all these sessions were interconnected. However, none of the past literature discusses this factor as a challenge for novice database modellers, and this could most probably be due to having conducted the studies on single tasks. But at least in this study, it was implicated that referring to their own solutions to transfer the knowledge was a difficulty faced by novices, particularly when their attempt at problem-solving was not systematic.

Another difficulty observed among the weaker-performing participants was information overload. Wurman (1989) define information overload as the excessive amount of information an individual might come across, which can cause stress and anxiety. Batra (2005) explained the potential causes (i.e., disorder, novelty, inconsistency, noise, and undifferentiated features) of information overload introduced by Reeves (1996) in relation to database modelling. Some of the observations made with the

participants who struggled with information overloading were in line with the explanations of Batra. For example, a participant struggled with information overload in Task 4 when developing the logical model. Batra defined this as a ‘disorder’ cause and stated, "*attribute-based models like the relational data model can create information overload and lead to low designer performance*". Novelty didn't appear to be an issue at least in this study, since the case study was laid out and presented in a particular manner. However, some indicated they were overwhelmed with all the concepts learnt for the first time. Batra defines the difference between real-world semantics and their data model representations can create representation inconsistency for the students, even though this was not observed among the participants in this study. However, some participants struggled with similar relations having different representations in the relational schema, like the fact that many-to-many relationships require a separate relation while one-to-many relationship does not.

Table 7.6 provides a mapping of the identified difficulties and challenges to the existing literature. The findings identified in this analysis, particularly on fragile knowledge, are well supported in the previous literature. However, certain challenges, such as potential information overload and language-related difficulties encountered during case study analysis, have not received comparable attention within the published literature. Interestingly, an area not discussed in the published literature was the issues around knowledge transfer when designing databases. To the best of the researcher's knowledge, there was only one publication that touched on this by Zyl & Mentz (2022), which was an extended abstract and hence was not included in the synthesis table.

The observations and analysis made in this study also provide evidence supporting the conjecture of other researchers that the challenges faced by the participants are largely caused by the higher cognitive load imposed by the tasks (Batra, 2005, 2007; Katz, 2018a; Mason et al., 2016; Mason & Seton, 2021). This observation will be discussed with the application of CLT in detail in Chapter 9

Table 7.6 Identified difficulties and challenges mapped to the existing literature.

Identified difficulties/challenges	(Batra, 2005, 2007)	(Sedrakyan et al., 2014)	(Hussain, 2017)	(Katz & Shmallo, 2015)	(Venable, 1996)	(Mason & Seton, 2021, 2020)	(Rosenthal & Strecker, 2019)
Fragile knowledge						✓	
• Terminology							
• Conceptual knowledge	✓		✓	✓	✓		✓
• Syntactic knowledge							✓
Invalid or missing assumptions		✓	✓				
Lack of domain-specific or real-world knowledge					✓		
Information overload	✓						
Language issues						✓	
Lack of a process to verify design decisions		✓			✓		
Lack of transfer							

7.10 Summary

This chapter focused on analysing think-aloud data as a group to identify the difficulties and challenges novice database modellers faced while attempting the tasks. The data was analysed using a reflexive thematic analysis to identify the common patterns. The next chapter discusses the analysis of the same think-aloud data from a different perspective, where a group of three participants' data were analysed individually in a narrative style to understand their learning progressions.

Chapter 8 Think-aloud: Narrative Analysis

8.1 Introduction

This chapter presents the second stage of analysis of the think-aloud data. The difficulties faced by the students and the common approaches to database modelling detailed in Chapter 7 are further explored here to understand the individual learning of the participants.

Learning trajectories have been extensively used in mathematics education to inform the development of curricula and interpret the research findings on the teaching and learning of mathematics (Simon, 1995). In the literature, learning trajectories have been defined and interpreted in several ways. As per Lobato and Walters (2017), some researchers have distinguished between teacher-conjectured possible progression and researcher-documented progression of actual learners. Further, Lobato and Walters (2017) mentioned that learning trajectories focus on the emergence and development of student thinking and understanding over time, while learning progressions establish predetermined benchmarks for students' learning at different stages due to the researcher's rational analysis according to existing literature. In their work, Lobato and Walters (2017) used the idea of learning trajectory and progression interchangeably and introduced the term LT/P, which captures different objects of learning where its elements could be cognitive conceptions, forms of discourse, observable strategies and textbook tasks. Similar to Lobato and Walters (2017), the researcher opted to use the term LT/Ps in this context, where LT/Ps is discussed in the scale of the learning of individuals across a single problem presented in tasks where the progress is researcher documented to identify the benchmarks of students' learning of each phase while solving the problem.

Lobato and Walters (2017) have presented a taxonomy of seven approaches: cognitive levels; level of discourse; schemes and operations; hypothetical learning trajectories; collective mathematical practices; disciplinary logic and curricular coherence; and observable strategies and learning performances. These LT/Ps are based on the work of researchers who claimed their work is related to learning trajectory or progression in mathematics education. Based on their approaches, the work of this study resonates more with the approach named "schemes and operations". This approach is focused on modelling students' initial schemes and mental operations and the modifications of these schemes over time. The main objective of this approach is to gain a detailed and in-depth understanding of how an individual's understanding of mathematical concepts develops over time. It's important to note that this approach is rooted in Piaget's theory of cognitive development (Piaget, 2003).

Despite not being used extensively in CS education, unlike in mathematics education, learning trajectories have garnered the attention of computing education researchers in understanding novice

programmers' behaviours and learning progression, which is discussed in detail in section 8.3.1. Hence, the LT/Ps were developed in this study to understand the progression of learning among novice database modellers, primarily by understanding the participants' thinking processes on developing a database model at various levels of learning activities they are engaged in. Further, the LT/Ps were used as a model to identify the impact of learning interventions provided to the novices. Additionally, the researcher used the LT/Ps to identify any barriers or challenges that novice learners faced during their progression that could have hindered their learning. As stated in the original approach by Lobato and Walters (2017), ideally, the research should be carried out over a long period of time to understand how the students' schemes developed over time. A major limitation of this work is that this study was constrained by only one introductory database course in the degree program. Hence, a longer view of progression was not possible beyond the semester of the study. According to Lobato and Walters (2017), a trade-off in their approach was the limited attention given to the teacher's pedagogical actions. However, in this study, efforts were made to address this limitation and include the pedagogical actions used by the teaching staff in conducting the research (Izu et al., 2019a, 2019b; Jiang et al., 2019).

The analysis in this chapter addresses the following research question to gain a better and more in-depth understanding of the novices based on their learning progression through a single problem they attempted in database modelling.

RQ3.3: How does the learning of novice database modellers of different capabilities develop over time?

The chapter first introduces the data analysis method. It should be noted that the participants and data collection details are provided in Chapter 3 and Chapter 7, respectively, since the analysed data are the same as in Chapter 7. Narratives are then presented for each participant as they progress through the modelling process, from analysing the case study requirements to developing an ERD. This narrative is represented as diagrammatic LT/Ps to facilitate the interpretation of the findings.

The final set of tasks related to normalisation is presented as a separate narrative since the normalisation problem was independent. As discussed in section 6.3.5, while the first four tasks focused on developing a database design using a top-down approach, the normalisation uses a bottom-up approach in situations where the data already exists. Moreover, it's a logical modelling technique used to ensure the development of well-structured database designs without any anomalies. Since the novices often found this technique to be rather difficult and challenging, the participants' learning progress of this session is presented separately from the other tasks.

In the context of the narratives, the term section refers to a specific section of the case study (see Appendix H). For example, the information about trainees from which the business rules are built is

presented in the case study of a section head trainee. Moreover, a session refers to a single instance of a think-aloud interview in which more than one or more tasks may have been completed by a participant.

8.2 Research Participants

The analysis presented in this chapter uses the same think-aloud data set used for the analysis in Chapter 7. See section 7.3 for a detailed discussion of the think-aloud data collection protocol used and Chapter 6 for the tasks in the study. As discussed, six participants (two from the pilot study and four from the main study) completed all four think-aloud sessions. However, due to the curriculum change, the data from the pilot study was omitted from this analysis. Further, due to ethics consent conditions, this thesis does not include the entire set of transcriptions, encodings, or analysis of the think-aloud sessions of all participants. LT/Ps were established for three participants who completed the full set of tasks and represented a spread of capabilities in terms of overall performance on the tasks. Their performance level was determined based on how they performed on the think-aloud tasks, not in the course, since it seemed a more reliable measure of ability than their assessment marks because the participants were studying during the COVID-19 pandemic, where the assessments were not controlled. The measures used were the number of interventions provided, the errors in their solutions, and the time taken to complete the tasks (see Table 8.2). The aim was to be able to compare and contrast learner progressions and gain insight into what types of teaching interventions help to progress learning and whether or not existing theories about learning could help inform the findings. It was hoped that the insights gained from this analysis might help educators design better learning activities, tasks and assessments and to be able to initiate informed teaching interventions.

Morgan, the **best-performing** participant, completed the tasks in four sessions within the allocated time with minimal interviewer intervention. The next participant, Blake, completed the tasks within the allocated sessions but needed more direct assistance from the interviewer than Morgan, who was selected as the **middle-level** performing participant. Casey had the **lowest performance level** and took more sessions to complete Tasks 1 and 2 of the database model development problem, whereas the others completed them within the allocated session time. Casey attempted all tasks but failed to complete the tasks given for the normalisation problem entirely. Despite somehow attempting the tasks, Casey required frequent interviewer intervention to progress, and at times, despite multiple attempts by the interviewer, the assistance was not enough to overcome the difficulties they faced.

8.3 Data Analysis Method

Narrative research explores how a subject has acted in various contexts across time, and it provides a way to understand the data by keeping the story intact (Polkinghorne, 1995). A “structural narrative analysis” is used to examine the progressions of the participants as they worked through each session.

This type of narrative analysis focuses on how the narrative is told, including the sequence of events based on the data (Kellam et al., 2015).

Considering the LTs in this context, it is important to note the sequence of activities, and this type of analysis draws out the sequence of activities that happened in a way that the thematic analysis presented in Chapter 7 does not. Hence, narrative analysis is useful as a way of presenting an individual's learning trajectory, which aims to understand and map what participants did in each session while progressing through tasks, which helped build a deeper understanding of the participants' challenges and barriers to progression. Also, the narratives were presented carefully not to create a comparison between the participants since every narrative has its own context, meaning, and significance, and analysing stories in isolation allows researchers to appreciate their distinct qualities and complexities and not violate the principles of respecting the uniqueness of the individual stories (Polkinghorne, 1995; Riessman, 2008).

The narratives are grouped by the participant and are presented from the researcher's perspective, backed by the participants' work. The information included in each narrative is as follows:

- A summary of the encoded think-aloud and retrospective interviews detailing the number of tasks, time to complete the task/s, the level of completion and whether they are partially or fully correct is presented. Further, the interviewer's interventions are summarised based on the scaffolding provided to the participant (see section 7.4). The time to complete the task does not include the time spent on the retrospective interview but rather the time the participant spent on the task, including the time spent on the interventions by the interviewer.
- The transcribed think-aloud data is presented in chronological order, along with the participant's quotes and solutions to support the narratives. While the participants' think-aloud data is the primary source of analysis, interpretations are also made based on the interviewer's observations while conducting the sessions. For example, if there are scenarios in which the participants didn't verbalise a reason behind a certain action, the interviewer's observation notes are considered.
- Then, the transcribed data from the retrospective interview is presented where relevant. In scenarios where the participants were provided with a substantial degree of scaffolding/interventions or where participants could easily solve the problem, retrospective interviews were not conducted.
- Along with the narratives, a visual representation of the construction of the LT/P is presented as a diagrammatic interpretation of the individual's learning progression.

Further, the participants were categorised based on their ability to solve the tasks according to Vygotsky's ZPD notion (discussed in section 7.4). These categories have been used throughout the narration to indicate their ability to solve data modelling problems.

1. What a participant can do individually or independently – Comfort Zone
2. What a participant can achieve with the help of knowledgeable others – Within the ZPD
3. What is beyond the participant's reach even with someone's assistance – Outside the ZPD

8.3.1 Development of the learning trajectory

As mentioned in section 8.1, LTs have been used by computing education researchers to understand the learning progressions of novice programmers. Hence it was decided to use a similar mechanism to understand the progression of novice database modellers over time in the think-aloud sessions. The first step was identifying how researchers presented LT/Ps in the CSER literature.

The limited literature on using LTs related to programming was mostly presented as narratives (Izu et al., 2019a) or various techniques specific to the domain (Worsley & Blikstein, 2013) or domain-specific visualisations (Jiang et al., 2019)).

Izu et al. (2019a) have attempted a learning trajectory by decomposing programming comprehension tasks into the 12 cells of the Block Model table, where they outlined how to sequence tasks based on their difficulty. Worsley and Blikstein (2013) have used a quantitative approach to analysis – an algorithm to study the changes in programming styles among novice programmers. They have looked at how tinkering and planning episodes affected progress through analysis of student source code and changes made to it on a task. The study was conducted with a group of novice Python programmers, analysing the programming activities of four novice programmers over six weeks. Their technique involved collecting data on the programming activities, such as codes and debugging sessions, to map out the learning trajectories over time.

Jiang et al. (2019) attempted a data-driven learning trajectory represented as a trajectory of the code developed by the students. They developed their programming trajectory based on what the students were doing however, this approach misses the time on a task and the thinking of the students while they were attempting the tasks. From their learning trajectory, it was hard to define whether the students were learning or doing random changes to the code until they got it right. Their statistical analysis of the learning trajectories does provide the common mistakes of students and how they are corrected but missing the evidence of how learning happens as provided in this study. Overall, Jiang et al.'s (2019) work can be considered the one study most similar to what was attempted in this study, but it lacked the richness of the data identified from the approach used here.

There was no literature visualising the learning trajectories related to programming, let alone in other areas of CSER. The closest work to address this gap was conducted by Jiang et al. (2019), where they utilised automatically generated code pathways in a syntax tree to identify areas where students encountered difficulties, which was the closest approach aligned to the work presented in this chapter. Recent research has predominantly focused on learning progressions for assessing student progress, using statistics and models to automate assessments and propose new exercises to facilitate learning. However, these approaches have limitations and challenges, particularly in understanding the assessments and determining intervention tasks that are appropriate (Harris et al., 2022). This study aims to address this gap by emphasising the visualisation of learning trajectories using a timeline. The visualised timeline is intended to allow a means of interpreting, summarising, and comparing participant learning trajectories at a glance.

The visualisation of the LT/P was developed from the verified transcription text and the timeline as recorded by automated transcription of the think-alouds. In developing each LT/P, three main components were taken into account:

1. A timeline developed from the transcriptions of the think-aloud interviews.
2. Interventions provided by the interviewer informed by the narrative.
3. Other key events in the timeline such as participant activity or inactivity and interviewer notes/observations of the affective state of the participant at key junctures.

Considering the above-mentioned components, a visualised timeline was developed (see Figure 8.1). The timeline consists of the following elements:

Interviewer interventions

As discussed in section 7.4, the interviewer interventions were divided into two main levels based on the assistance provided: high-level interventions and low-level interventions.

High-level interventions include the assistance provided as general prompts (IP) and clarifications (IC), and low-level interventions are hints (IH), redirected tasks (RT) and exact solution (ES) as the final resort of assistance. These are indicated as colour-coded rectangles in the timeline to identify the type of intervention distinctively (e.g., IC – blue, IP – orange, IH – green, RT – yellow, and ES – red). Prior to every intervention, the reason for providing the assistance is given in the arrowhead (e.g., FK difficulty, uncertainty, syntax difficulty, request help, etc.). Moreover, if the participant triggered the interviewer-provided intervention by requesting help, those interventions are further marked with an asterisk (e.g., IH*).

Key events – Periods of activity and inactivity

The period of activity (marked in light grey in the timeline) indicated the time when the participants actively engaged with the task, whereas the period of inactivity (marked in a darker shade of grey in the timeline) indicated the time when the participants were not doing anything, including thinking aloud.

Other key events

Apart from the marked key events of activity and inactivity in the timeline, events that were important to understand the progression of the participants were identified through interviewer observations and their field notes. For example, an event like self-identifying an error and fixing it didn't trigger the interviewer to provide any assistance but was important in the timeline to understand what the participant did. These are marked in purple in the timeline (see Figure 8.1). Further, the errors made by participants are marked with a crossed circle (as indicated in Figure 8.1) in the timeline bar, along with a description of the error. Further, the arrows in between interventions indicate the reason for providing the assistance. See Table 8.1 for a detailed description of these reasons.

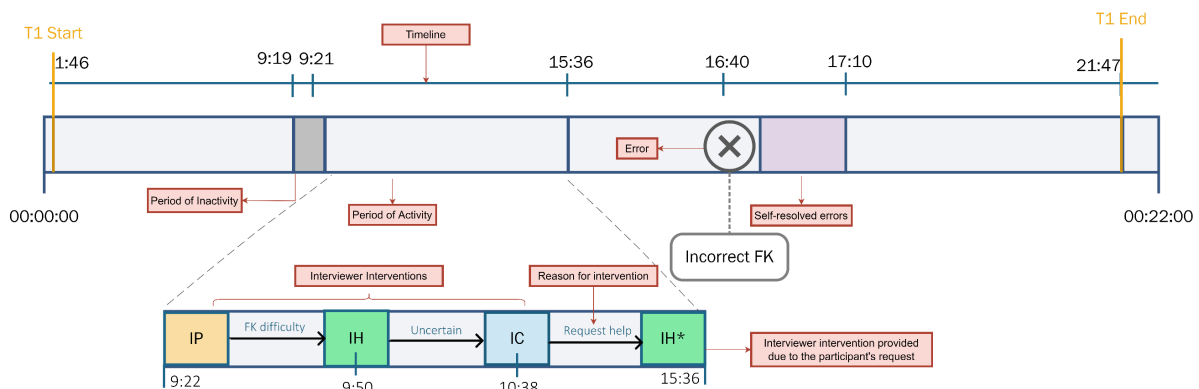


Figure 8.1 Sample LT/P timeline

Also, it's important to note that "T1 start" on the timeline indicates the actual time the participants started working on the task from the point they settled after their introductory chat, but the time was recorded from the beginning of the session, which is time 0:00:00.

Table 8.1 Reasons for providing the assistance.

Reason for Intervention	Definition
Tool difficulty	Instances where participants struggled with handling the Visual Paradigm tool.
Uncertain	Instances where participants struggled or felt unsure about how to proceed.
Incorrect explanation	Instances where participants struggled to explain the scenario using correct terminology.
Relationship difficulty	Instances where participants struggled with naming relationships or indicating the participation (optional/mandatory) correctly.
Relationship type difficulty	Instances where participants struggled with incomplete knowledge related to relationship types (e.g., identifying/non-identifying or strong/weak relationships).
Attribute difficulty	Instances where participants struggled to differentiate between types of attributes (e.g., composite and multivalued attributes).
FK difficulty	Instances where participants struggled with identifying the foreign keys.
Mapping rule difficulty	Instances where participants struggled to explain or apply the mapping rules used to transform a conceptual model into a logical one.
Dependency difficulty	Instances where participants struggled to identify the functional dependencies.
BR grammar difficulty	Instances where participants struggled with the grammar required for writing business rules.
Notation difficulty	Instances where participants struggled with recalling and applying the correct notations in database models.
Request help	Instances where participants requested assistance from the interviewer.
Clarify	Instances where the interviewer decided to provide further clarification after a low-level intervention.
Difficulty referring previous task	Instances where participants struggled to refer to the previous task.
Narration difficulty	Instances where participants struggled with reading and interpreting the given case study.
Lack of attention	Instances where participants struggled with a lack of attention to detail, e.g., interpreting the case study without properly reading it.
No progress	Instances where participants struggled to progress in a task.

8.4 Morgan

8.4.1 Session 1: Analysing the case study and extracting business rules

Morgan, one of the best-performing participants in the think-alouds, appeared to be quite confident in attempting the tasks. They completed both tasks, which involved analysing the case study and extracting business rules within half of the allocated session time. Morgan spent 23 minutes completing Task 1, while they managed to finish Task 2 within 10 minutes. Throughout the session, Morgan appeared to be operating within their *comfort zone* – a zone where a learner could solve problems independently (Vygotsky, 1978a) and didn't require the interviewer's assistance to progress through the tasks.

However, Morgan made a couple of mistakes, one of which they identified and corrected without the interviewer's assistance, which placed them in their comfort zone. But, the other two errors mentioned in Table 8.2 were not identified till the interviewer's queries during the retrospection. This actually made them be in their ZPD (aka zone where a learner can solve tasks with the help of a knowledgeable other) instead of their comfort zone, which will be discussed in the following section. Overall, Morgan followed a quite consistent systematic approach in attempting Task 1 and refereed the extracted information correctly to build the business rules in Task 2, which could have been the reason for completing Task 2 rather quickly.

Task 1: Narrative

Morgan started Task 1 (see section 6.3.2) by reading the preamble, identifying a few potential entities, annotating them with an asterisk, mentioned: *"I'll just leave a little mark here to remind myself"*. Morgan skim-read the case study (see Appendix H) before starting and noted that the case study was divided according to at least one of the main entities. From the beginning, Morgan appeared to be attentive. Morgan started by reading the first section and underlining the term *Trainees*, indicating that it's an entity. This was followed by adding parentheses to annotate the related attributes of the *Trainee* entity. Interestingly, Morgan didn't articulate the reasons behind their choices. They continued underlining *Employer* as an entity and using parentheses to indicate the relevant attributes: (*company*), (*address*), (*contact number*) and (*employerid*). Morgan then went on to identify the existing relationship between the *Employer* and *Trainee* by reading the relevant text in the case study and then annotating that text using the letter *R*. Morgan noted how the text explained the relationship in both directions. Once this initial analysis was done on the section related to trainees, Morgan started to write down this information in the space provided in the task sheet. Even though the task only expected the participants to note down the identified constructs, Morgan seemed to take a different route in expressing the information using symbols. Morgan first added both entities and relevant attributes into an entity diagram element. After reading the sentences related to the relationship, Morgan connected the entities with a line and even identified the relevant cardinalities representing them using crow's foot notation. See Figure 8.2. They even appeared to be thinking about the optional/mandatory participation hinted in the case study, stating that the sentence *"the employer can refer..."* is synonymous to "may" refer to, emphasising the optional nature of the participation.

Trainees: It is necessary to store details about all the trainees who participate in the courses. Each course participant is identified by their trainee ID and this is stored along with their name (composed of first name and last name), date of birth, gender, address (composed of street number, suburb, city, postcode), contact number. Along with these details the previously acquired qualifications of the trainees should be stored in the database. Since most of the time these trainees are referred to the course by their employers it is also necessary to store the details of the employers including the name of the company, address, and contact number. Each employer can be uniquely identified using the employer ID. An employer can refer many employees to a training course and a trainee can be only referred to a course via one employer. However, there can be few trainees who are not referred by an employer.

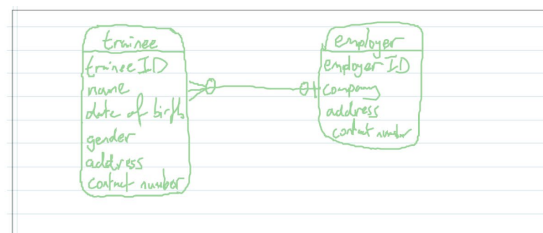


Figure 8.2 Morgan's annotations on the case study and presentation of the extracted info

Then Morgan moved on to the second section of the case study, immediately underlining the *Course*, mentioning it was an entity and identifying the relevant attributes with parenthesis. Following this, Morgan annotated the entity *Program*, its attributes and the relationship between the *Course* and the *Program* using the same annotations as earlier. Further, Morgan identified the relationship between the *Trainees* and the *Program*. Morgan finished that section by identifying the *Class* entity and its attributes. Then, as observed previously in the case of the *Trainee* and *Employer*, they started summarising this information using crow's foot notation to describe the relationship. Morgan first added the entities *Program*, *Course* and cardinality on the *Program* side. And then, they immediately decided that the information related to the cardinality of the *Course* side was missing in the case study stating that:

It doesn't say anything about the relationship of a program to a course but can make an educated assumption. So a program has to have at least one course. I'm currently thinking about programming one, which we did COMP500, but there was an engineering one as well. So I'm trying to find the relationship based on that. So I'm going to say, at the very least, a program must be associated with one course, and you can have multiple courses that do the same program. So I'm just going to assume that it's one to many, but that could be wrong.

The case study expressed the same idea, but Morgan somehow missed it, probably because it was an elaboration of a sentence and presented within brackets. However, they accurately discussed their knowledge of a similar scenario and assumed that it should be a many-to-many relationship based on that knowledge. Even though Morgan applied a relevant tactic here to resolve an issue by making an assumption in this case, it was not required since the case study explained the relevant cardinality. Morgan made an incorrect assumption that was contrary to the case study information. Morgan then moved on to add the relationship between the *Program* and the *Trainee* by adding a one-to-many relationship using the crow's foot notation between the entities and, finally, adding the *Class* entity and a one-to-many relationship with the *Program*.

Finally, Morgan moved on to the last section, underlined the *Instructor* entity, and annotated the attributes in the case study with parentheses. Then, they read the sentence related to the relationship between the *Instructor* and the *Program* and annotated it using an “R”. When working on this, Morgan read each sentence at least twice and, in particular, revisited the ones explaining relationships as if to confirm their interpretation of a sentence’s meaning. This indicates their attention to detail in analysing the case study description. The annotations they used were consistent throughout the case study. Figure 8.3 indicates the learning trajectory of Morgan at the end of Task 1. Morgan immediately moved on to Task 2 in the same think-aloud interview session. Hence the retrospection for both tasks was conducted at the end of the session.

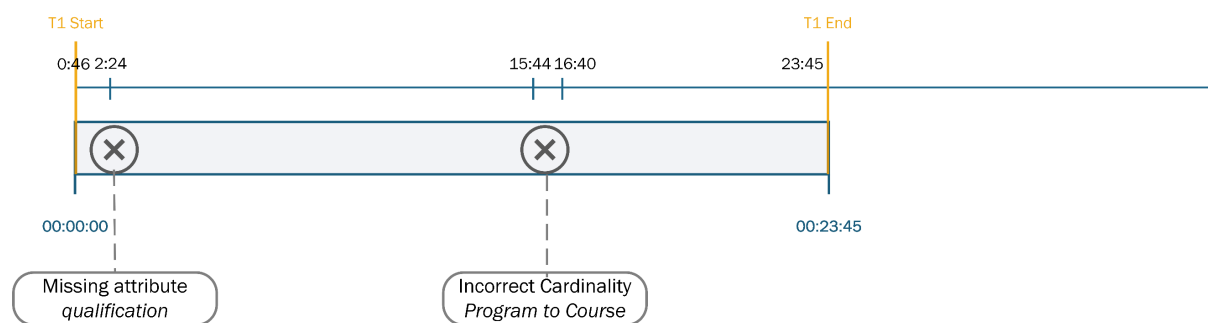


Figure 8.3 Morgan’s LT/P at the end of Task 1.

Task 2: Narrative

Morgan used a systematic approach to developing the business rules founded on their answer to Task 1. Morgan started with the business rule defining the relationship between the *Employer* and *Trainee* entities. They checked the cardinality they had arrived at in Task 1 and defined a business rule for both directions of the relationship. Morgan correctly used the grammar that was taught in class to write the business rule. After writing the first two business rules, as indicated in Figure 8.4, they realised the mistake of not using proper verbs to explain the relationship stating, “*hold on, I forgot it’s not have, it’s referred to...*”. Their systematic approach and early presentation of entities and relations using crow’s foot notation appeared to be helpful in effortlessly developing the business rules.

A trainee may ^{be referred by} have a zero or one employer
 An employer may ^{refer} have ~~no~~ zero ~~or~~ many trainees
^{generate}

Figure 8.4 Morgan’s attempt at writing the business rules showing later corrections.

Morgan continued with the task and used “have” in the next business rule between the *Course* and *Program*. But after their earlier encounter, Morgan immediately realised that proper verbs could be used to explain the relationship instead of using “have” in all the business rules. Without any prompt, Morgan fixed this slight issue to match the grammar taught in class and continued to complete the rest of the business rules without assistance. Throughout the task, Morgan was in their comfort zone and no interventions were needed since they recognised and fixed their mistakes on their own. The only persistent error in Morgan’s solution was propagated and related to the cardinality relationship between *Course* and *Program*. Morgan’s completed LT/P at the end of the session is presented in Figure 8.5.

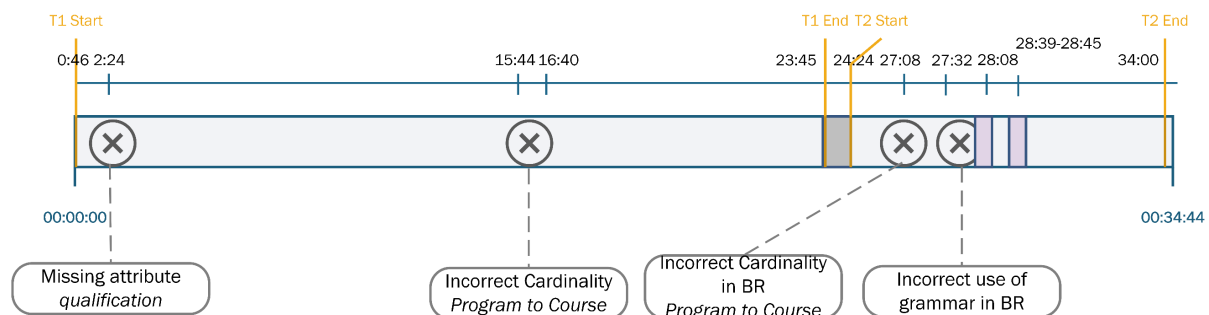


Figure 8.5 Morgan’s LT/P at the end of Task 2.

Tasks 1 and 2: Retrospection

Despite being careful and re-reading the case study sentences, Morgan still overlooked some of the important details in the case. The interviewer had to point out the missing *qualifications* attribute, and Morgan accepted their mistake, claiming they must have missed it since it was on a different line than the other attributes. Despite being consistent, this was an instance where Morgan appeared to have a lack of attention to the case study. Furthermore, the interviewer mentioned their incorrect assumption, leading to an incorrect cardinality between the *Course* and *Program* entities. Morgan agreed that they were unsure about this but didn’t attempt to correct their mistake before ending the session.

8.4.2 Session 2: Developing a conceptual model

Session 2 consisted of only one task (Task 3 discussed in section 6.3.3) involving creating a conceptual model and building on the knowledge gained through the tasks in session one.

As discussed in section 7.3.3, the in-person think-alouds were transitioned into remote ones due to the lockdowns that happened with the pandemic COVID-19 pandemic. Apart from the first session, Morgan did all their sessions virtually, and since Morgan didn't own a tablet device, this session and the remaining sessions were conducted using MS Teams and Visual Paradigm to develop the conceptual model.

Morgan took 22 minutes to complete this session's task and was provided with six interviewer interventions while attempting the task. All but two of these were related to difficulties Morgan had using Visual Paradigm. At this point, all the participants have attempted using Visual Paradigm at least once during their labs, and they already had the tool installed in their machines when they attended the session. Morgan completed session two, sitting comfortably within their ZPD, with the interviewer providing high-level interventions. However, Morgan, while thinking they had successfully completed the task, actually had one error (see Table 8.2), which was one propagated from the first session where they overlooked detail in the case study and made their own incorrect assumption.

Task 3: Narrative

Morgan started Task 3 by looking back at their solution from the previous session and added the *Trainee* and *Employer* entities based on the first business rule they had derived in Task 2. Then, they decided to add the rest of the entities before proceeding further. Morgan then added the attributes referring back to their solution to Task 1, particularly focusing on the doodles they had made using crow's foot notation. They started by adding the attributes for the *Trainee* entity. They missed adding the *qualifications* attribute, which was also missed in Task 1 and was discussed during the retrospection of the last session. The interviewer did not intervene at this stage but instead decided to revisit this attribute's omission in the retrospective interview. Morgan continued adding the attributes for the *Employer* entity. Morgan followed the same process for the *Course*, *Program*, *Class*, and *Instructor* entities, continually referring back to the partial diagrams developed in Task 1. See Figure 8.6 for the initial entities added by Morgan. Once the entities were added with the relevant attributes and primary keys, Morgan decided to add the relationships between the entities.

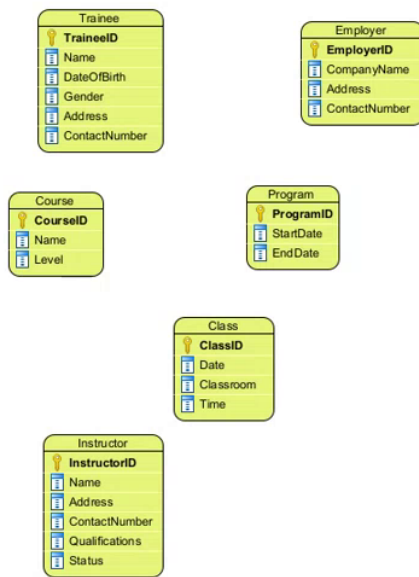


Figure 8.6 Morgan's initial entities with attributes

Morgan referred to the first business rule they developed in Task 2 and attempted to add the relationship between the *Trainee* and the *Employer*. However, Morgan appeared to be hesitant and unsure as they were hovering the mouse pointer over the relationship options in Visual Paradigm but did not add anything for nearly one and a half minutes. As mentioned by Morgan, the exact cardinalities they wanted to add were not among the displayed choices. The interviewer suggested to Morgan that the cardinalities could be changed later, prompting him to progress. With this, Morgan decided to add a relationship that was closest to what they needed and change the cardinalities later. This intervention was not needed due to being unsure of their original solution or how to build the relationship but due to a lack of familiarity with drawing models in Visual Paradigm.

The direction Morgan added the *Employer* $\rightarrow$ *Trainee* relationship, dragging from the Trainee entity to the Employer entity to draw a line. In Visual Paradigm, this results in an incorrect foreign key being automatically added to the *Employer* entity. In addition, while editing the autogenerated cardinalities for the relationship, Morgan ticked the option to make the relationship an identifying one, which was incorrect. After making these changes, Morgan asked the interviewer whether the automatically created foreign key (*traineeid* in the *Employer* entity) was required as they were creating a conceptual model. With this question, the interviewer decided to intervene and reflected back, asking Morgan whether they thought the added foreign key was correct. The interviewer was trying to get Morgan to make a decision rather than remain in a state of uncertainty. Looking back at the solution to Task 1, Morgan mentioned: “*employer has many trainees, so primary key of the employer should be in trainee ... we should know the employer of every trainee. Hmm... I think this is incorrect*”. Morgan realised the issue

but had trouble fixing this error using Visual Paradigm. After over a minute, the interviewer further intervened, mentioning to Morgan how the direction of a relationship was added using the tool could result in an incorrect cardinality. Even after this prompt, Morgan remained unsure of how to fix the issue and stayed silent for nearly a minute. At this point, the interviewer decided to provide low-level intervention instead of the prompts provided so far by asking Morgan to delete the existing relationship and add the relationship in the correct direction. Morgan finally seemed to understand the source of the issue and deleted the existing relationship, adding the relationship correctly, this time by dragging the mouse pointer from the *Employer* to the *Trainee* instead of the other way around. The LT/P of Morgan, including the interventions related to using the Visual Paradigm tool, is indicated in Figure 8.7.

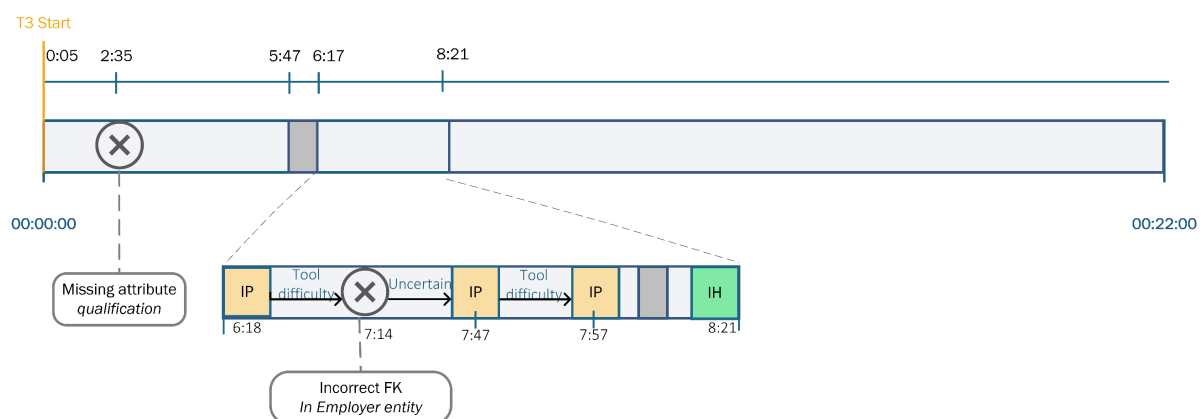


Figure 8.7 Morgan's current LT/P.

However, when Morgan changed the cardinalities, Morgan again ticked the check box to indicate that the relationship was identifying (i.e., strong), repeating an earlier error. Morgan appeared to be hesitant about whether to tick this option or not and mentioned that if selected, the relationship would be displayed as a solid line. To jog Morgan's memory, the interviewer asked whether it was an identifying relationship or not. Initially, Morgan indicated they didn't know what an identifying relationship was. The interviewer asked whether they could explain what a weak entity was instead. Despite not knowing what an identifying relationship was, Morgan could correctly explain a weak entity using an illustrative example. With that same example, the interviewer further clarified what a strong or identifying relationship was at Morgan's request. Morgan was then immediately able to identify their mistake and referred back to the partially developed diagram with *Trainee* and *Employer* in Task 2 and stated:

Okay I get it now, last time I should have make it a dotted line [pointing the mouse over the relationship between Trainee and Employer] because it's a weak relationship since they [entities] are both strong.

With a mix of high and low-level assistance, Morgan finally correctly added the relationship between the *Trainee* and *Employer*. Morgan then proceeded to add the relationship name between the *Employer*

and *Trainee* by referring to the relevant business rules developed in Task 2. See Figure 8.8 for Morgan's solution before and after the interviewer's intervention. At this point, Morgan also decided to spread the entities across the canvas, and it seemed that they were now considering the layout of the diagram.

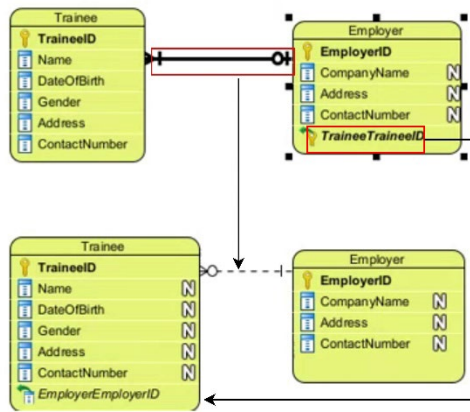


Figure 8.8 Morgan's initial attempt to add a relationship before (top) and after (bottom) the interviewer intervention. Red boxes indicate the errors Morgan made.

Morgan's progress appears to have been hindered largely due to their lack of familiarity with Visual Paradigm, though they also had an issue with the term *identifying relationship*, which was a term introduced to them in the class. However, they have used both the terms strong and identifying relationship during the class, which could contribute to the issue Morgan experienced. Considering all the help on the tool that the interviewer had to provide, it seemed that using the Visual Paradigm added an additional cognitive strain on Morgan despite Morgan having followed a systematic approach to developing the model until that point. The interviewer felt that if Morgan had been working with pen and paper, they would have been able to complete the task more readily.

Next, Morgan added the relationship between the *Trainee* and the *Program* by referring back to the partial diagrams developed in Task 1 and the business rules developed in Task 2. Again, they initially added the relationship in the wrong direction, creating an incorrect foreign key. But, without any prompt, Morgan immediately identified the issue, stating:

"Oh, no it's the other way like earlier. This adds a wrong foreign key. Let me delete it."

Morgan followed the same process to add the relationships between the remaining entities correctly. In doing this, Morgan made one mistake when adding the cardinality between the *Course* and *Program*, adding a many-to-many relationship instead of a one-to-many relationship. This was an error Morgan made in Task 2, which was not fixed and remained in the business rules they were referring back to. The interviewer put a note to discuss this during the retrospection despite having the same discussion

during the retrospection of session 1. Apart from that, Morgan managed to complete the task without any further assistance. See Figure 8.9 for the final conceptual model developed by Morgan. Once finished, Morgan double-checked and verified what they did with the business rules developed in the previous session. Only two out of six participants did this check for correctness once a task was completed. Morgan's completed LT/P after Task 3 is given in Figure 8.10.

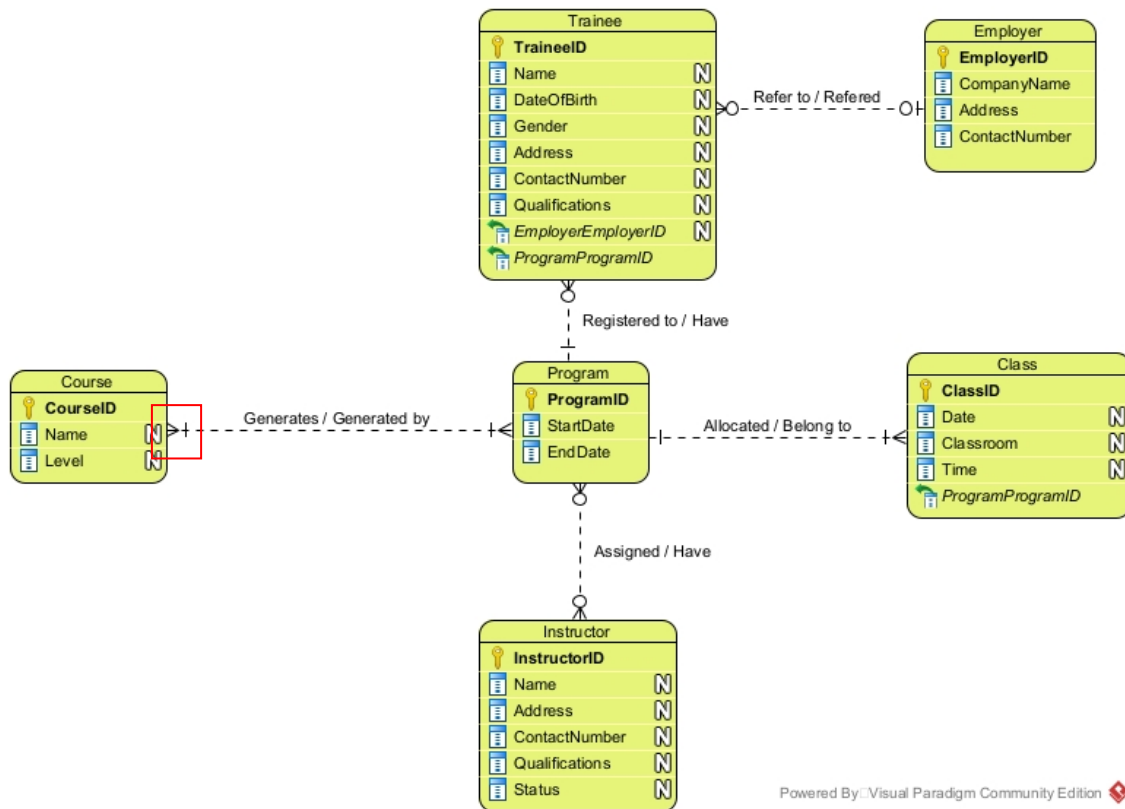


Figure 8.9 Morgan's conceptual model. The red box indicates the one relationship error in the model.

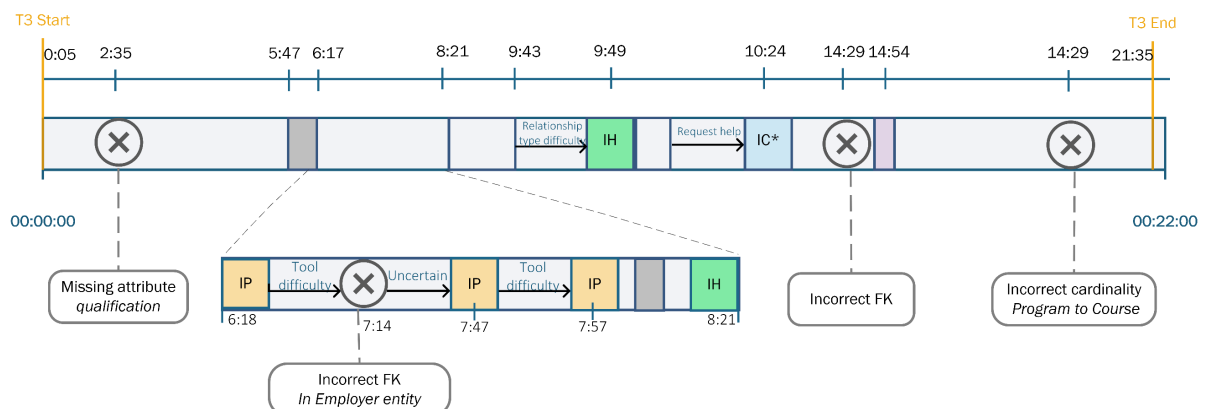


Figure 8.10 Morgan's LT/P at the end of Task 3.

Task 3: Retrospection

A quick retrospective interview after the task was completed allowed the interviewer to explore the two issues: the omission of the *qualifications* attribute and the incorrect relation between the *Program* and *Course*. The interviewer told Morgan that the *qualifications* attribute was missing. At this point, Morgan wondered whether they had also missed it in the previous session. The thinking out loud triggered Morgan to recall the discussion they had with the interviewer at the end of the previous session. Morgan mentioned that they had forgotten to add a note after the previous retrospection. This suggests that Morgan largely relied on the attributes, entities and business rules developed in Session 1 to build the conceptual model using a direct translation process without checking and verifying their original understanding of the business rules developed in Task 2 as part of Task 3 modelling process.

With the interviewer's feedback Morgan, in this retrospective interview, read the related section in the case study and then added the *qualifications* attribute to the *Trainee* entity. It is this model that is shown in Figure 8.9. The interviewer then went on to explore the incorrect cardinality in the model. Morgan recognised that it happened due to the incorrect assumption they made during Task 1 and recalled their discussion with the interviewer in the previous session. This time, Morgan made a note indicating their issue as a reminder.

8.4.3 Session 3: Transforming the conceptual model into a logical model

Session three was comprised of a single task, Task 4 (see section 6.3.4). Morgan used the Visual Paradigm tool to complete the task in 30 minutes. The interviewer needed to intervene seven times, and the majority of the interventions (six out of seven) were high-level interventions. This assistance helped Morgan to remain within their ZPD to complete the task.

The interviewer provided Morgan with their previous solutions from sessions one and two before the session. Morgan started by reading the task displayed on the interviewer's screen and opened the Visual Paradigm file they developed the last time, as the task required them to transform their conceptual model into a logical model.

Task 4: Narrative

After reading the task, Morgan appeared to be unsure about the term 'mapping rules' stating:

Uhm, mapping rules... I'm quite not sure at this moment but I'll probably remember it, but I can't remember specifically calling them mapping rules or anything. But in a logical design you can't, If I remember this correctly, you can't do many to many relationships you have to put something in between them to resolve that problem.

After this, Morgan immediately turned their attention to their conceptual model from Task 3 and unprompted started attempting to resolve the many-to-many relationship between the *Program* and *Instructor* entities. Morgan added a new entity to the model named *Program_Instructor* and explained how the new entity gets connected to *Program* and *Instructor* entities. Then they added a one-to-many relationship from the *Instructor* to the *Program_Instructor* entity but then began questioning this action. Morgan expressed doubt as to the cardinality of the *Instructor* side even though it was added correctly. Morgan started mumbling incoherently, hovering the mouse pointer over the relationship they had added and this went on for around two minutes without any progress. Hence, the interviewer intervened and asked whether Morgan could recall the related mapping rule or a similar example from their lecture materials to try and direct their thinking. With this prompt, Morgan managed to recall a many-to-many relationship where the cardinality for both sides was one-to-many. Morgan appeared to partially recall what the lecturer did to resolve this issue, but their recall was not complete. In this instance, the interviewer took the same example that Morgan recalled, showing them the lecture materials and explaining the reasoning behind the cardinalities. After the interviewer's clarification, Morgan returned to their model to correctly add the cardinalities between the three entities *Program*, *Program_Instructor* and *Instructor*. After adding the cardinalities, Morgan realised that the *Program_Instructor* entity had two foreign keys, which meant it should have a composite primary key consisting of these two foreign keys. Morgan changed them to be primary keys, which automatically changed the lines representing the relationships from dashed to solid lines, at which point they recalled:

Okay now Program_Instructor is a weak entity which means the relationships are strong. I remember this from last session.

It is clear that the interventions provided were sufficient to progress and consolidate Morgan's knowledge.

Once the relationships were added, Morgan mentioned they couldn't remember how to name them and stayed silent for over a minute. The interviewer suggested that they could refer to their previous solutions. Morgan returned and referred to their answers for Task 2 to figure out the proper verbs that could be used as the relationship names. Morgan's LT/P up to the point of resolving the many-to-many relationship names issue is given in Figure 8.11.

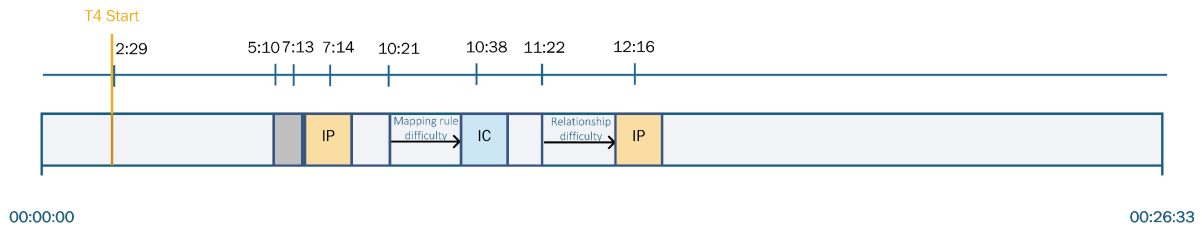


Figure 8.11 Morgan’s LT/P till resolving the many-to-many relationship.

After resolving that relationship, Morgan seemed unsure of what to do next and was silent for over a minute. The interviewer intervened and asked whether any other mappings needed to be done to transform the conceptual model into a logical one. Morgan was sure there were more, however, they couldn’t recall any then.

Ah, I'm sure I'm missing something, but I don't think no matter how long I sit here thinking about it, I don't think I'm going to it pull out of thin air, really.

It seemed that Morgan wasn’t sure what else they could do to proceed. The interviewer decided to intervene with the intention of bringing forth Morgan’s recollection of the mapping rules related to the attributes by asking: *Can you recall any mapping rules related to attributes?* This prompt somewhat triggered Morgan’s memory, and they started explaining a scenario where multiple values cannot be stored for the same information and that such information should be stored separately from the entity. Morgan was, in fact, describing how a multivalued attribute needed to be resolved however, it seemed to the interviewer that they were unable to accurately express the idea as they could not use the correct terminology. Despite performing better, Morgan still had issues grasping the terminology, making it harder for them to express their ideas clearly. However, with their explanation, the interviewer decided to provide a direct hint and guided Morgan towards the *qualifications* attribute of the *Trainee* entity to see whether they could identify that it was a multivalued attribute by asking the following:

OK, let me ask you something. If you go to the trainee entity there's an attribute called qualifications. So what would you store there? What type of attribute is that if you can recall the name? [Interviewer]

After that, Morgan explained that *qualifications* would store multiple values, and in a table, there would be only one allocated row, hence a new table should be introduced to store the many *qualifications* a *Trainee* had acquired. It seemed Morgan was more comfortable by first relating to the data occurrences of the *qualifications* attribute, which later led to their conceptual understanding – they relied on a concrete example to describe the more abstract concepts involved. Taking Morgan’s explanation into consideration, the interviewer then explained that *qualification* was a multivalued attribute that could

store multiple values per *Trainee*, and how it could be resolved and the mapping rule related to that. With this new explanation, Morgan added a separate entity to handle this (see Figure 8.12).

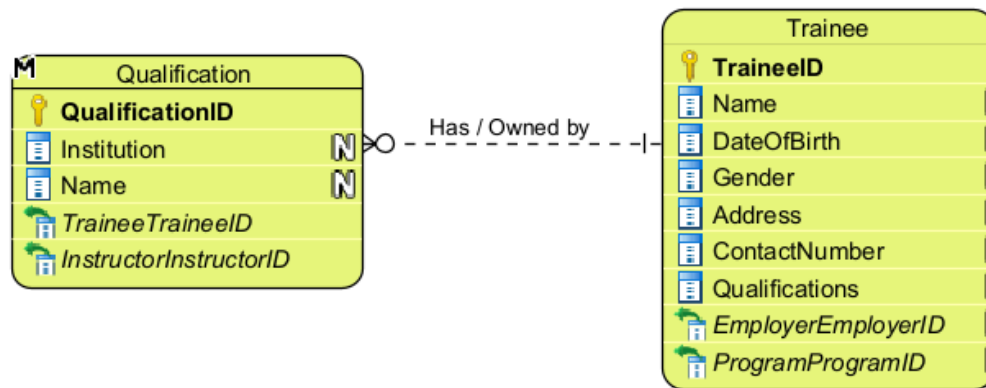


Figure 8.12 Morgan's logical model showing their resolution to the multivalued attribute problem.

Morgan went on to explain that they didn't realise *qualifications* stored multiple values when reading the case study. It is likely that Morgan hadn't paid due attention to understanding the problem statement text when they extracted key information during the first session. When creating the new entity, Morgan made an error by not recognising that the primary key needed to be a composite attribute (i.e., *traineeID, qualificationID*). Morgan resolved the *qualifications* attribute in the *Instructor* entity and indicated the task was completed. See Figure 8.13 for the completed solution and Figure 8.14 for the finalised LT/P.

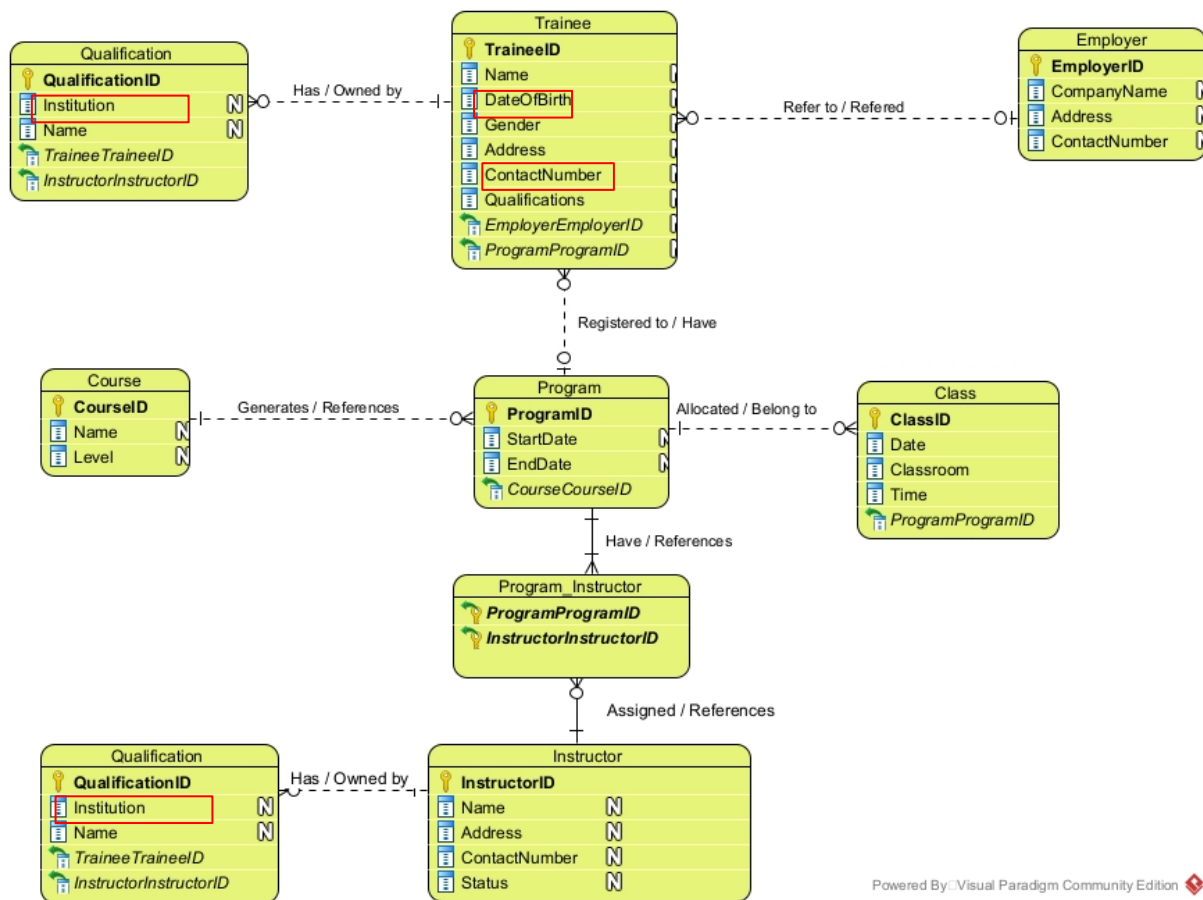


Figure 8.13 Morgan's completed logical ERD with any remaining errors highlighted using a red box.

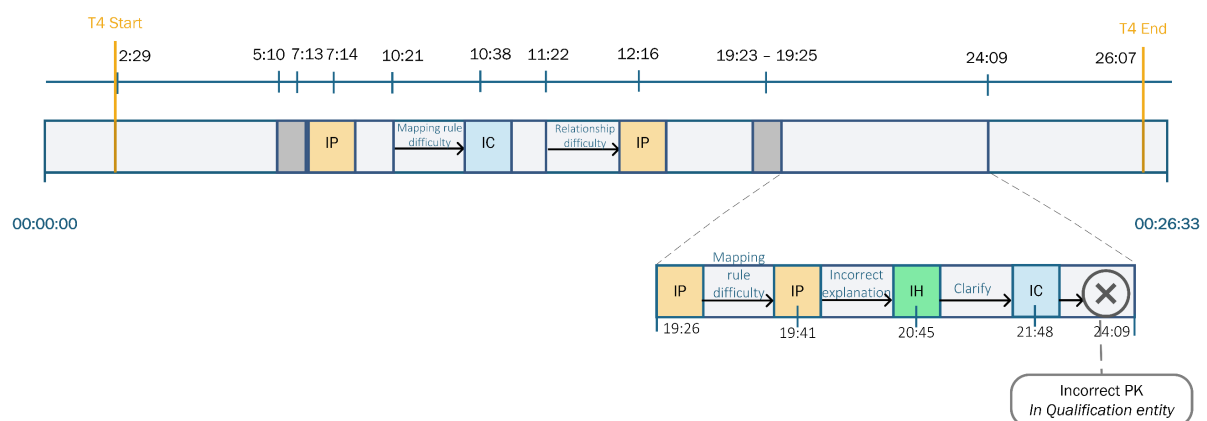


Figure 8.14 Morgan's LT/P at the end of Task 4.

Task 4: Retrospection

Despite finishing the task, Morgan's logical ERD was not in 3NF as required. They failed to resolve composite attributes like *name* and *address* in the *Trainee* entity. When the interviewer asked retrospectively about the composite attributes, Morgan mentioned they recalled composite keys but were unsure about composite attributes. The interviewer directed Morgan to read the trainee section of the case study, and immediately, they were able to identify that *name* was a composite attribute.

Throughout this think-aloud session, Morgan struggled to use the correct terminology but was able to quickly grasp the ideas when the interviewer used illustrative examples to explain the terms. Moreover, Morgan demonstrated a lack of familiarity with the mapping rules required for the transformation of the conceptual model to a logical model with the exception of the rule related to resolving any many-to-many relationships. Throughout the session, Morgan failed to mention mapping one-to-many relationships as the mapping rules indicate that one side entity's primary should be in the many side entity as a foreign key. The interviewer asked why this was the case, and Morgan said that this was because, in their Visual Paradigm setup, one-to-many relationships were added by the tool by default.

As discussed earlier, the primary key of the newly introduced *Qualification* entity was also incorrect. The interviewer discussed this with Morgan here in the retrospective interview. Morgan was able to understand this concept when it was explained to them from the implementation perspective using terms of tables, rows, columns, etc. The interviewer also pointed out the incorrect foreign keys (*instructorID* in *Trainee* and *traineeID* in *Instructor*) in both the *Qualification* entities connected to *Trainee* and the *Instructor*. Morgan immediately realised it was their self-described "silly" mistake of copying and pasting the same *Qualification* entity twice instead of creating a new one. Overall, Morgan concluded that the mistakes made were their own fault and that they arose from not thinking things through.

This makes sense. I wasn't thinking about it properly and was just looking at this [pointing to their previous solution in Task 3] and using it.

8.5 Blake

8.5.1 Session 1: Analysing the case study and extracting business rules

Blake attempted both tasks in the allocated session time, where they spent 16 minutes and 33 seconds on Task 1 and approximately 15 minutes on Task 2. They fully completed both tasks however there were a few errors they made in Task 1 (see Table 8.2), which Blake didn't realise until the interviewer brought it up during the retrospective interview. With the assistance provided by the interviewer, Blake identified their mistakes in Task 1 and correctly rectified them during Task 2. In Task 2, despite having some challenges with a few interventions, the interviewer managed to keep Blake within their ZPD.

Blake was able to solve issues and learn from them. Overall, Blake wanted to solve the tasks independently, and persisted with the tasks and put considerable effort into comprehending any newly acquired knowledge gained through interviewer intervention. Notably, when starting Task 2, Blake identified errors made in Task 1 and rectified them in the process of developing the business rules for Task 2.

Task 1: Narrative

Blake started Task 1 (see section 6.3.2) by reading the case study's (see Appendix H) preamble and immediately underlined the term *CareerAcademy*. After a short pause, Blake indicated that they were trying to figure out whether *CareerAcademy* is an entity and mentioned:

At the moment, I'm just trying to figure out if Career Academy is an entity. And I think it is because following it they talk about trainees, instructors, courses and classes they offer.

Despite this being an incorrect choice and flawed reasoning, the interviewer allowed Blake to continue to see whether they discovered the misunderstanding later.

Blake identified *Trainee* as an entity along with its attributes and noted them in the space provided. This was followed by identifying *Employer* and its associated attributes. Even though Blake initially started by underlining *CareerAcademy* and *Employer*, they did not follow through with this tactic. Instead, Blake shifted to writing the identified terms in the space provided on the paper sheet, as shown in Figure 8.15.

Next, Blake moved on to identify the relationship between the entities. After a short pause, they listed the *Training Course* as an entity. Then they wondered whether the *Employee* was the same as the *Trainee* and mentioned, “*Are employees the same entity as trainees not sure, Yeah. Ah never mind.*”. It seemed Blake was not sure whether they were the same and decided not to worry about it. However, despite questioning his understanding, Blake didn't make any further effort to think this issue through or look to the interviewer for help. Instead, they immediately decided to move on. Once the entities were listed, Blake started reading the sentences which explained the relationship between the *Trainee* and the *Employer*. They identified the relationship between the two entities and noted them on the paper, as indicated in Figure 8.15, and added the cardinalities. At this stage, the identified cardinality of *Employer* side was incorrect, probably because Blake failed to read the last sentence of the relevant section in the case study. They were unaware of this mistake.

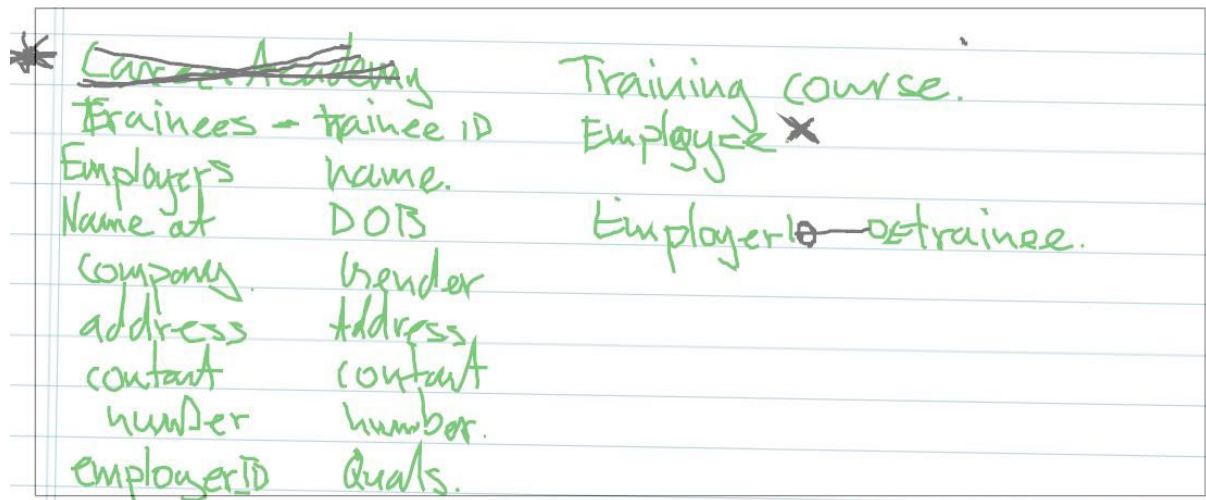


Figure 8.15 Blake's identified constructs for section 1. The text in green is the items identified during the think aloud and the edits marked in black are the changes Blake made after the retrospective interview.

Blake moved on to section 2 of the case study. After reading the section, they mentioned that *Course* is an entity and noted that in writing along with the attributes. This was followed by adding the *Program* along with its attributes. Blake then indicated the relationships between the *Course* and *Program* and *Program* and *Trainee* by verbalising while doodling them. However, Blake thinking aloud, stated that they were unsure about the symbols they had used, asking the interviewer for guidance. The interviewer assured Blake since the requirement was only to identify the existing relationships at this stage, the model diagram symbols were not important. Even though Blake hesitated, they actually used the taught crow's foot notation to doodle the cardinalities (Figure 8.16). After a short pause, Blake verbalised:

*I'm wondering what a class is maybe it's an entity. Class is it the same as the course?
I think the class is an entity.*

Reading further, they realised *Class* was, in fact, a separate entity since there were several attributes that were relevant to a *Class* in the case study. With that information, Blake added the *Class* and its attributes to their list and noted the relationship between the *Program* and *Class*.

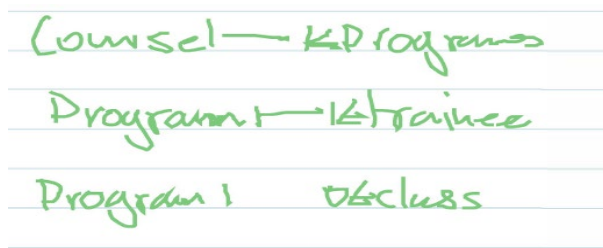


Figure 8.16 Blake's doodling of the identified relationships for section 2.

Finally, Blake started reading section 3 of the case study. They listed *Instructor* as an entity along with the relevant attributes. They further identified that attribute *status* could hold two different values and added it as *status (freelance or professional)*. Blake then read the last couple of lines and indicated the relationship between *Instructor* and *Program*. Interestingly, this time Blake opted to explain the cardinalities by writing them while verbalising (see Figure 8.17) instead of using symbols like previously:

*An instructor can be allocated, assigned to them, based on the number of trainees.
 So there could be more than one instructor for each program, oh it could be zero or many [add 'o or many' on the instructor side and '1' on the program side].*

The annotations Blake used for the relationships were inconsistent. But most importantly, the cardinalities were incorrect. This is presumably due to either not reading the related line, partially reading it, and missing information to interpret the relationship or misinterpreting the provided information. With this, Blake indicated that they were finished. They did not verify their solutions either during the process or at the end of the process. Failing to check as they went meant that they failed to correct errors as they went, and by the end, the interviewer suspected that even if they had verified their overall answer, they would not have identified all of their errors. The interviewer decided to have a retrospection before starting Task 2. Figure 8.18 indicates Blake's LT/P at the end of Task 1.

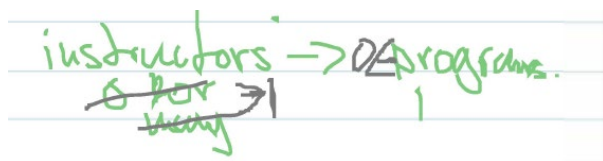


Figure 8.17 Blake's presentation on the relationship between instructor and program. The changes in black colour indicate the changes Blake made after the retrospection.

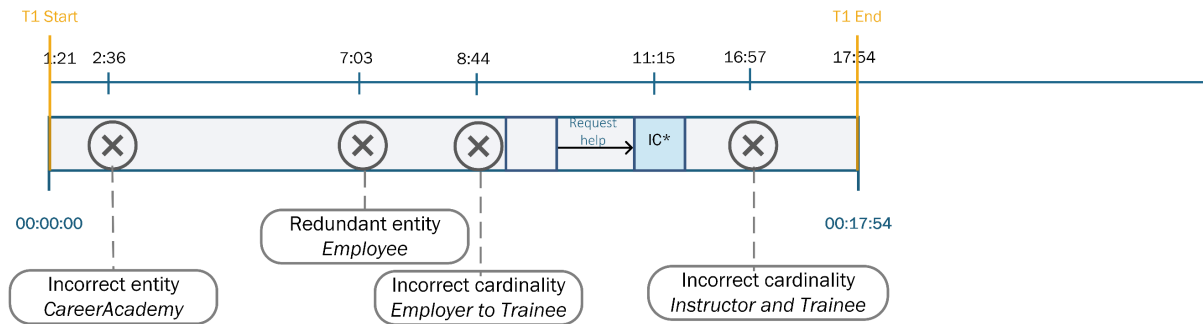


Figure 8.18 Blake's LT/P at the end of Task 1.

Task 1: Retrospection

Blake finished the task with only one high-level assistance. The incorrect cardinalities were discussed immediately after the completion of the Task rather than waiting until the end of the session retrospective interview. Blake was asked to read the sentence in the case study they had skipped over related to *Trainee* and *Employer*. Initially, Blake failed to recognise they had the interpretation of cardinality incorrect, and the interviewer had to explain how to interpret cardinalities in both directions of a relationship. With this instruction, Blake was able to recognise from the case study that the cardinality of the *Employer* side should be *zero or one* instead of only *one*.

Right. So it doesn't have to be one, it could be zero to 1? I get it now because trainee doesn't have to have an employer.

Then the interviewer directed Blake to the incorrect cardinality between *Instructor* and *Program*. This time, Blake was able to apply their newly gained knowledge of needing to interpret the cardinalities in both directions and immediately identified their mistake.

I think it'll be one or more instructors assigned. Instructors and programs... instructors maybe assigned to any number of programs. So one to many and going backwards while some programs can have more than one instructor So I think that's one to many. Yeah. So I think it goes from at least one or many instructors to zero or more programs.

Before finishing the retrospection, the interviewer gave Blake feedback on handling synonyms by keeping the noun closest to the case study description while eliminating the other. Further, the interviewer explained *CareerAcademy* is not an entity. Blake understood the issue and mentioned:

Yes, I wasn't a hundred per cent sure about this. I get it, and it's the entire system. Isn't it?

Task 2: Narrative

Blake started reading the preamble for Task 2, almost immediately stating that they couldn't recall how to write the business rules. They did not request assistance and instead tried to remember the syntax they had been taught during class:

Um, it goes like ... each entity one, then the verb. Oh, wait. Yeah, the verb and then the entity two and then, oh ...

Despite their perseverance to continue on their own, Blake appeared to be uncertain about the syntax they recalled and kept mumbling incoherently without attempting to write a business rule. The interviewer decided to intervene by suggesting Blake write the business rule for the relationships identified in section 1. She asked Blake to break down the problem and tackle a section at a time, reassuring them that she would provide assistance if required. This prompt meant Blake was able to make an attempt, and they wrote the following business rule (see Figure 8.19).

A screenshot of a handwritten business rule in green ink on a white background. The text reads: "Q2 Each employer refers zero or many trainees." The text is underlined.

Figure 8.19 Blake's first attempt at writing business rules.

After reading aloud what they wrote, Blake immediately indicated:

Each employer refers zero or many trainees. No, that doesn't sound right.

Despite being able to identify there was an issue with the written business rule, Blake couldn't exactly articulate what the issue was or figure out how to handle it. Blake's business rule was almost correct but missing the term, which suggests an entity's participation (may/must) in the relationship. When Blake was unable to progress through indecision, the interviewer gave a hint by providing the correct grammar used for business rules and then asked Blake to compare it with what they wrote to identify the issue they had. With this, Blake was able to identify that the may/must phrase was missing and started updating the written business rule.

Despite the low-level intervention – a hint from the interviewer in providing the syntax, Blake could not proceed and asked for help from the interviewer:

Okay let me try it, hmm...Sorry I forgot when to use may and when to use must. Can you help me with this?

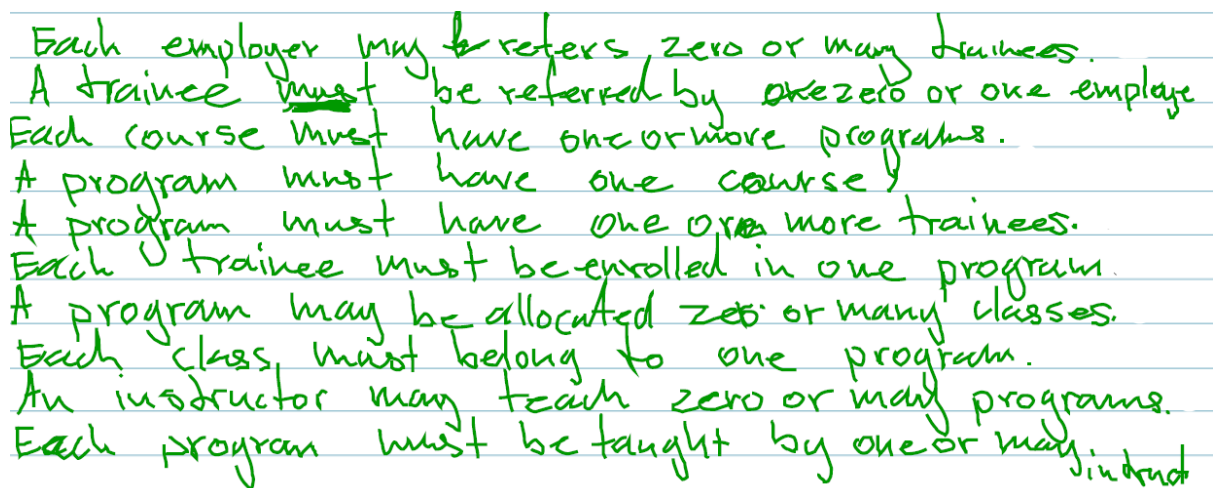
Blake started strongly and continued calmly during Task 1, and this was the first time they appeared hesitant and unsure of what they were doing. The interviewer realised that even though Blake realised

the terms may/must from the provided grammar, they did not understand the types of participation (i.e., partial or mandatory) to use the terms in business rules correctly.

Due to this, the interviewer decided that Blake should be redirected to a simpler task (T2.1 –see section 6.3.2) that asked them to complete a simple fill-in-the-blank task based only on one section of the case study. This task should allow Blake to attempt a similar task with less complexity to check whether that could help their cognition. Blake tried to fill in the blanks of some business rules provided in the redirected task. Blake said that it was easy to fill in the blanks since they were partially completed with the options provided. They were able to fill in the blanks correctly and independently. Then, the interviewer asked Blake to compare two of the business rules completed by Blake in T2.1, taking the cardinalities and the participation (may/must) into consideration. This prompt seemed to help Blake further since they indicated:

Okay, so based on this, when may is used the cardinality is zero or one, and each program is taught by one or more ... oh I get it that's a must or required. This was helpful thank you.

With the acquired knowledge, Blake returned to where they left on Task 2 and reattempted the task. Blake correctly updated the first business rule between the *Employer* and *Trainee*, followed by the business rule between the same entities in the opposite direction. After the T2.1 redirected task and interviewer prompt, Blake was able to complete all the Task 2 business rules without intervention but with one error, which was revisited in the retrospection interview. Blake's final set of business rules is given in Figure 8.20, and their LT/P at the end of Task 2 is provided in Figure 8.21.



Each employer may refer zero or many trainees.
A trainee must be referred by one zero or one employee.
Each course must have one or more programs.
A program must have one course.
A program must have one or more trainees.
Each trainee must be enrolled in one program.
A program may be allocated zero or many classes.
Each class must belong to one program.
An instructor may teach zero or many programs.
Each program must be taught by one or many instructors.

Figure 8.20 Blake's finalised business rules.

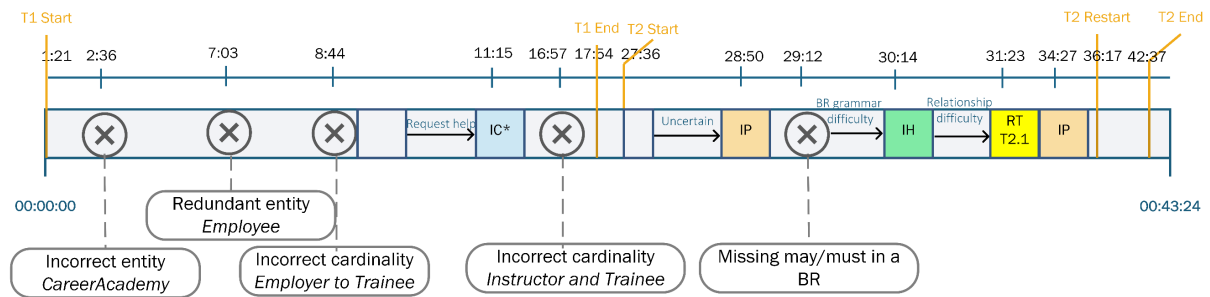


Figure 8.21 Blake's LT/P at the end of Task 2.

Task 2: Retrospection

At the end of the session, the interviewer found one instance where Blake had used “must” instead of “may”. Then, the interviewer directed Blake to that business rule and asked them to check the participation. Blake immediately understood the issue and mentioned: *“I don't know how I made that mistake. Silly mistake, I guess”*. Apart from that, none of the errors identified in Task 1 were propagated into Blake's Task 2 solution. Blake found it easy to write the business rules after the redirected task they attempted and said:

Uh, it was, uh, after that small exercise and once you explained it and showed me, I kind of got the idea. Definitely, I got clarity on some issues I had before this session. Okay with this in the next stage is ER diagram where I'd use attributes and identify primary and maybe foreign keys. Okay all right I see the process.

8.5.2 Session 2: Developing a conceptual model

Session 2, conducted in person, consisted of a single task (Task 3 described in section 6.3.3), which asked the participants to develop a conceptual model on the knowledge they gained from the tasks they attempted in session 1.

Blake managed to complete the task within 22 minutes. The majority of the time, Blake referred to the solution they developed in the previous session to create the conceptual ERD. They appeared to be well-composed throughout the session and seemed to follow a systematic approach to developing the conceptual model but struggled at certain points when encountering foreign keys. However, the interviewer managed to navigate Blake through the problem through the use of two high-level and two low-level interventions, and Blake finished the task despite having a couple of errors in the produced model (See Table 8.2).

Task 3: Narrative

Blake was provided with the previous session's solution for both Tasks 1 and 2 at the beginning, and they started Task 3 by going through the answer for Task 1. Blake indicated they would refer to the business rules and the case study to develop the conceptual model. Blake was observed skim-reading the case study mentioning, *"I need to remind myself a bit before I start"*, and indicated that *CareerAcademy* is not an entity, which was something that had come up during the retrospection of Task 1. After revisiting the case study and their solution to Task 1, Blake began by adding the *Program* entity. They then added the attribute *programID*, marking it as the primary key. Then Blake added the remaining attributes, *startdate* and *enddate*, to the *Program* entity. After a short pause, Blake indicated that *Program* was related to *Course* after skimming through the business rules from the previous session and added the *Course* entity to the diagram along with its attributes. Blake then added the cardinalities to the relationship by referring to the business rules developed in the prior session (Figure 8.22). Once that part was done, Blake returned to the business rules and read the two rules related to the *Course* and *Program* out loud to verify the added cardinalities were correct.

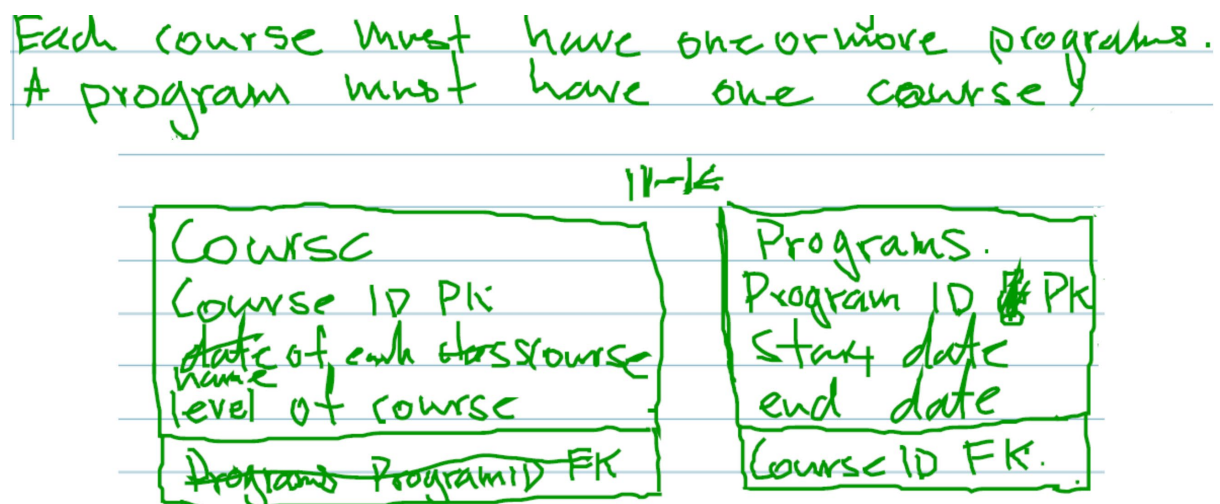


Figure 8.22 Blake's initial entities and the relationship (bottom) and the referred business rule (top).

Blake then moved on to the *Employee* but indicated they were unsure about the order in which to add the entities:

I'll do employees now. I'm actually not too sure which one to start with. I started with the program because I just saw program repeatedly. I think I'm going to add classes next.

Then Blake added *Class* followed by *classID*. But Blake abruptly stopped and mentioned that the *programID* should be a foreign key for the *Course*. But Blake seemed unsure about their decision but decided to add it anyway, which was incorrect, and they did not realise this issue. Then, they continued

to add the remaining attributes to the *Class* and indicated the *classID* as the primary key. However, Blake still seemed to be unsure about the foreign key they added earlier, mumbling, “*I’m not quite sure about the foreign key. There should be one, though.*”. Blake sat silently for more than a minute, and the interviewer decided to intervene by asking how they might identify the foreign key. After a short silence, Blake mentioned:

To identify a foreign key. Uh, I think it is in the lecture. I just don’t remember. I just thought the foreign key would become the primary key from the parent.

Since it was apparent that Blake had an incomplete knowledge of foreign keys, the interviewer decided to provide a low-level intervention, a hint and asked whether they recalled how the direction of the foreign key was decided or to which entity the foreign key should be added. This question seemed to have prompted Blake to recall what they were taught and realise the foreign key was added to the wrong entity:

Oh yeah. Mali [the lecturer] said something about the master, which is the primary key of the one [entity] with the 1 [cardinality] which is the mandatory entity that goes to the other one. Yeah. So I think I got this the wrong way around, so yeah. Yeah. So courseID should become a foreign key here [pointing the pen to the program entity] Yes. [Blake]

Then, Blake corrected the diagram, as shown in Figure 8.23. Subsequently, the interviewer further clarified the reason for using foreign keys and how they can be used during the database design development stage.

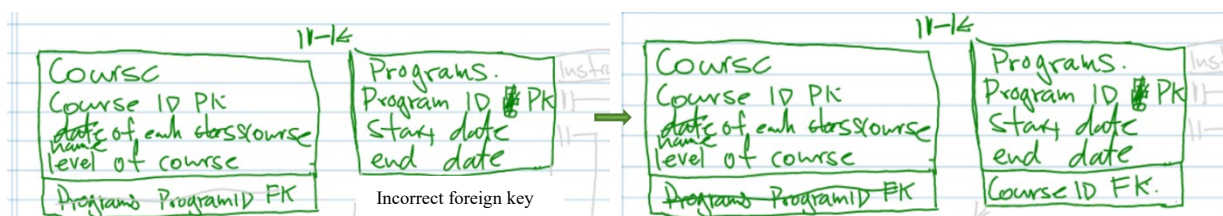


Figure 8.23 Blake’s diagram before (left) and after (right) the interviewer’s intervention.

After the interventions, Blake continued and added the correct relationship, cardinalities and the foreign key between *Program* and *Class*, referring back to the business rules related to the *Program* and *Class* to verify their answer before moving on.

Then Blake added to the diagram the *Instructor* entity, the relevant attributes, and the primary key. Then, Blake identified the cardinalities correctly. However, it was apparent that they were struggling with the foreign key, but in this case, it was probably due to the many-to-many relationship:

And the instructor may teach zero or many programs. Each program must be taught by one or many instructors. Um... in this case, I'm not too sure which way ... I think maybe this should go to that. What I meant was the instructor's primary key should go to program I think.

Blake kept reading the sentence explaining the relationship repeatedly and was uncertain about how to proceed and hesitant about what to do next. It seemed that Blake had a very fragile knowledge of how to handle foreign keys. After a minute, the interviewer decided to intervene by providing a hint relating to the earlier intervention and asked: “*In the earlier one, you put programID in Class as a foreign key since you need to know the program each class was related to. So in here if you take an instructor, what do you need to know in regard to the program and the other way around also?*” After a quick deliberation, Blake caught up to the hint and suggested:

Okay if you take a program you need to know like which instructors are teaching it. And instructors should know what program they are teaching. Hmm... wait a minute do they both swap? Like instructor take program's primary key program takes instructor's primary key.

Then Blake managed to recall what was taught in class and explained that a new entity is needed to resolve the relationship, which was not required at the conceptual level. Hence, Blake swapped the primary keys of *Instructor* and *Program* entities and added them as foreign keys.

Then, Blake moved on to add the *Trainee* entity and its attributes. The relationship and the cardinalities were added correctly. The interviewer observed that Blake referred to the business rules as they thought about and added the cardinalities to the diagram. When adding the relationship between the *Employer* entity and the *Trainee*, Blake had to read the developed business rules a couple of times and then add the relationship with cardinalities correctly. Figure 8.24 provides the conceptual model developed by Blake, and Figure 8.25 illustrates Blake's LT/P for Task 3.

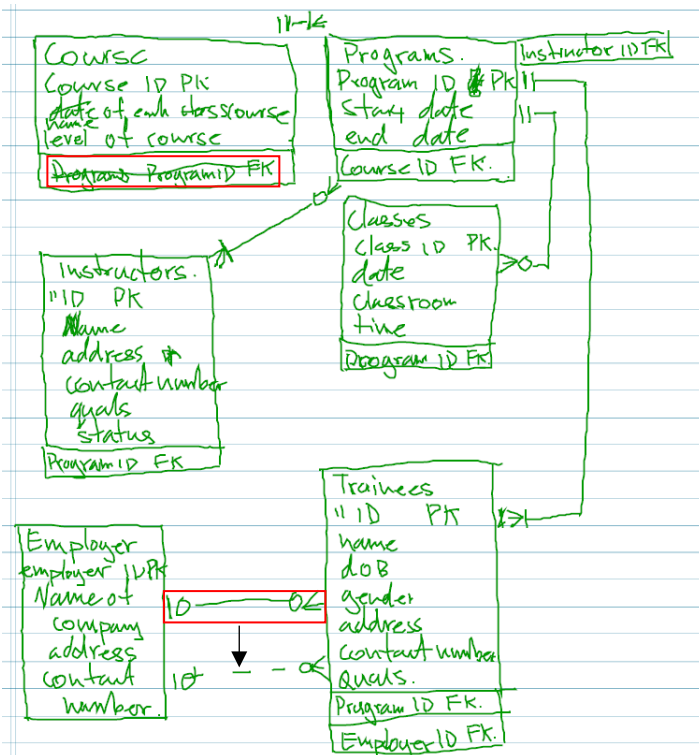


Figure 8.24 Blake's conceptual model. The red boxes highlight the errors they made, which were later corrected during the retrospection. The error at the bottom indicates a syntax error: using a solid line (indicating a strong relation) instead of a dashed line (indicating a weak relation).

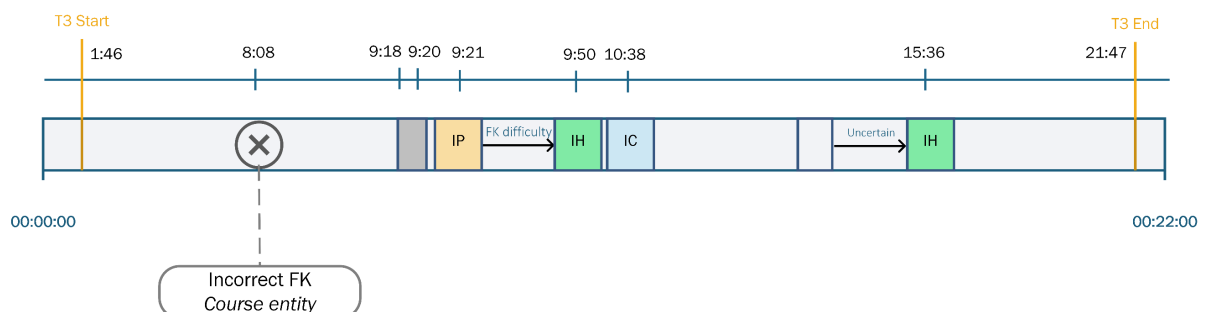


Figure 8.25 Blake's LT/P for Task 3.

Task 3: Retrospection

During the retrospective interview, Blake mentioned that the task was not too hard, but they were confused about applying foreign keys. However, with the provided interventions, Blake managed to complete the task. The interviewer explored some minor errors she had identified. She highlighted that Blake had not distinguished between strong and weak relationships in the diagram (as shown in Figure 8.24). Blake was able to explain how to differentiate them using a solid line to denote strong relationships and a dashed line to indicate weak ones.

However, Blake was unsure about the concept of strong and weak relationships and asked the interviewer to explain it further. The interviewer, for example, uses a relationship between an employee and a dependent to illustrate the strong and weak relationships. This led to a discussion, and after some time, Blake mentioned: “So... *there might be hmm... a weak entity we can have here in between Instructor and Program, isn't it?*” It seemed Blake at least understood to a certain extent what a weak entity was, and the interviewer explained further why it is a weak entity and the connected relationships are strong relationships.

8.5.3 Session 3: Transforming the conceptual model into a logical design

This was the first session Blake participated in online, but they used a tablet to complete the task, which closely resembled the face-to-face data collection using a pen and paper. Hence, Blake, unlike the other participants, did not use Visual Paradigm to produce the models and instead hand-drew them. No notable issues were encountered during Blake’s online sessions. Blake seemed to be coherent, able to think aloud, and confident in handling the online session.

Session three consisted of a single task (Task 4 described in section 6.3.4). Blake completed the task within 20 minutes. Throughout the task, Blake operated within their ZPD. With high-level interviewer assistance, they were able to progress. While working within their ZPD, Blake did make some mistakes, which were discussed in detail during the retrospection to gain insight into the source of these errors.

Task 4: Narrative

Before beginning the session, Blake was provided with the solutions they developed in the previous session via email, and they used a separate machine to view them. Once the interviewer explained the task, Blake started Task 4 by reviewing the conceptual model developed during the previous session. Then they mentioned:

At the moment, I have my conceptual design and going from conceptual to logical, it's just changing the many-to-many relationships. I only have one between course and instructor hmm... yes so I'm going to start now.

It was apparent before starting the task that Blake was trying to formulate a plan on how to proceed. However, it seemed they had a misconception about how to derive a logical model from a conceptual model in that they believed that this only required resolving many-to-many relationships. But for the time being, the interviewer decided not to intervene and allowed Blake to continue.

Blake added the *Course* and the *Program* entities with attributes and primary keys and mentioned that there was a one-to-many relationship between the entities. After a short pause, they added the *courseID*

as the foreign key in the *Program* entity. Blake was able to clearly articulate this decision, which would suggest that Blake managed to transfer knowledge gained in the previous session to this current session, indicating progress and that some degree of consolidation of knowledge had taken place with respect to their understanding of foreign keys.

Blake moved on and indicated that an associative entity is needed to resolve the many-to-many relationship between the *Instructor* and the *Program*. Blake created a new entity called *Program_Instructors*, and added a composite primary key to this associative entity consisting of the *programID* and *instructorID* attributes. Blake demonstrated the skill of resolving this many-to-many relationship by explaining how the foreign keys were handled in a scenario like this, and they seemed confident in what they were doing (See Figure 8.26).

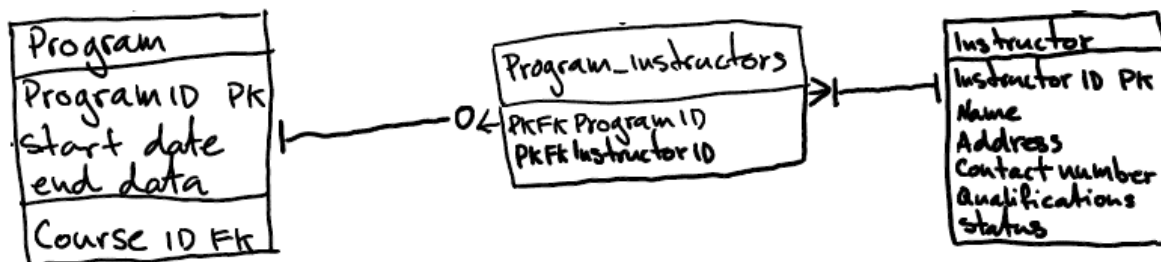


Figure 8.26 Blake's resolution of the many-to-many relationship in the logical model.

Next, Blake added the *Class* entity and mentioned that it has a one-to-many relationship with the *Program* entity while drawing it. And they then added the foreign key *programID* to the *Class* entity. This was followed by including the *Trainee* entity and connecting to the *Program* entity. The interviewer observed that Blake was not resolving the composite attributes (e.g., *address* and *name* in the *Trainee* entity). This could have been due to Blake's misconception that transforming the conceptual model into a logical one only required resolving many-to-many relationships and otherwise merely transferring information from the conceptual model without actually resolving them. The interviewer decided to intervene at this point to see whether a prompt could be provided to help Blake realise their mistake. The interviewer asked Blake whether they could identify what type of an attribute *address* was, and this prompt allowed Blake to think about what type of attribute *address* actually was and they immediately mentioned:

That would be a composite attribute, so this should actually be atomic. So I should add street, city etc. Yes, I actually forgot about that in the logical model, all the attributes should be at the atomic level.

Blake updated the *address* attribute with *street, city and postcode*. Another observation made by the interviewer was that Blake did not resolve the multi-valued attributes (e.g., *qualifications* in the *Trainee* attribute) either. Then Blake mentioned the *Trainee* entity had two foreign keys *programID* from the *Program* entity and the *employerID* from the *Employer* entity, which they had yet to add. Overall, it was evident that Blake had a good grasp of handling the relationships in the logical model but was less able to manage attributes.

Consequently, Blake added the *Employer* entity and the relationship between the *Employer* and the *Trainee*. This time, they handled the composite *address* attribute correctly, recalling the earlier discussion on composite attributes. Blake asked whether they needed to denote the strong and weak entities on the diagram. Upon Blake's inquiry, the interviewer indicated that to denote the strong and weak entities, using 's' and 'w' near the entity just as a way to ensure whether they can identify the difference. Blake managed to differentiate between the two types of entities but still failed to use the correct ER notation (i.e., solid and dashed lines), although the same issue occurred in the previous session. It appeared that Blake couldn't recall the syntax to differentiate the two types of relationships. With this, the interviewer clarified that the strong and weak entities could be identified if the difference is denoted using the relationships. The clarification appeared to trigger Blake to recall what they had learnt previously, and then they updated the logical ERD with the correct syntax (See Figure 8.28). Blake's LT/P with all the interviewer interventions is provided in Figure 8.27.

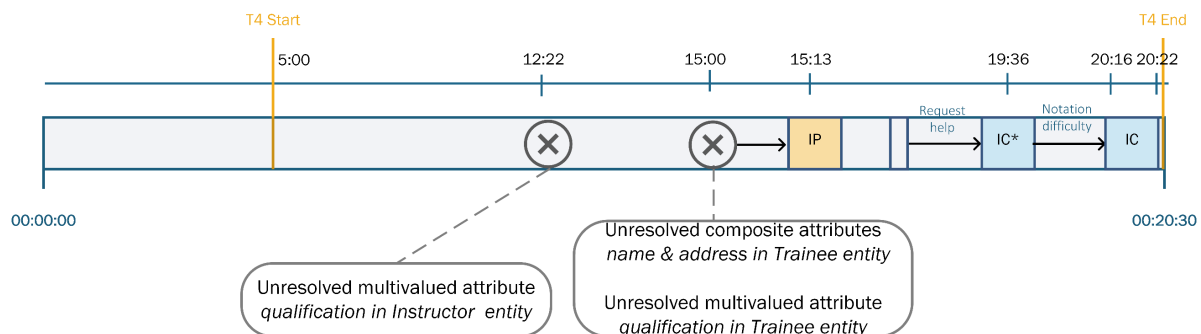


Figure 8.27 Blake's completed LT/P at the end of Task 4.

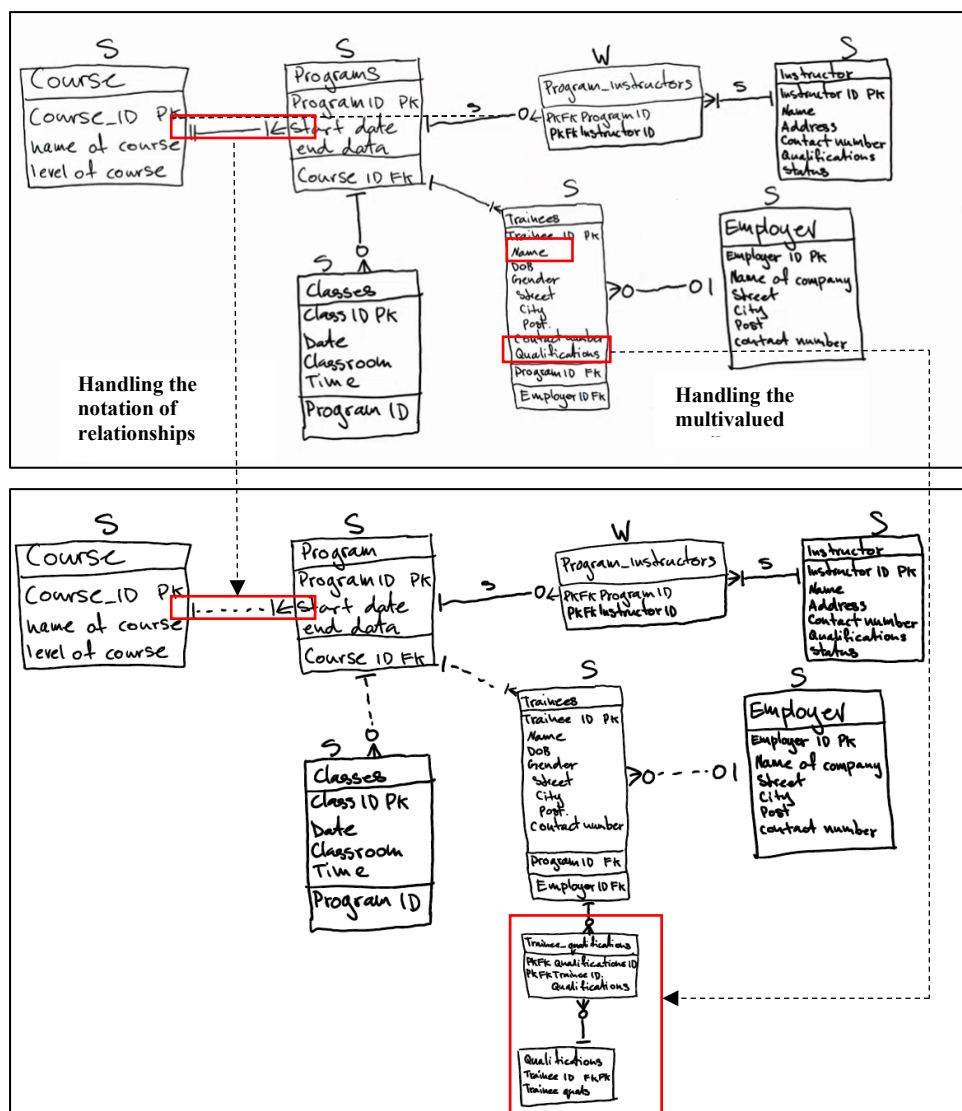


Figure 8.28 Blake's logical ERD before (top) and after (bottom) the retrospection interview.

Task 4: Retrospection

Even though Blake indicated that he completed the task and was confident the solution was correct, they had actually missed several mappings during the transformation. The interviewer asked Blake how they would represent the *qualifications* attribute, considering that a trainee might have more than one qualification. Blake immediately indicated they might need a separate entity to handle that attribute and mentioned: "Is it a recursive one... no right?". Blake was unsure and remained silent, so the interviewer asked whether they had discussed multivalued attributes in class. Blake admitted that it was taught but couldn't recall the details. Blake appeared to be struggling with the terminology and was confused about the different types of attributes like composite and multivalued, but mentioned:

I think it's uh attribute that could take different values within one multivalued. But I do remember we should make new relations.

This indicates that even though Blake learnt about the types of attributes, it was difficult for them to recall and apply these types in the given context. With this knowledge, the interviewer decided to explain multivalued attributes in detail and encouraged Blake to resolve them on the diagram.

Blake was sceptical about how to proceed and then did nothing for close to two minutes before deciding to add a new entity that represents a *qualification* with *traineeID* as a foreign key and *trainee qualification* as an attribute. Blake assumed a one-to-many relationship between the *Trainee* and the *Qualification* entity. Since Blake didn't add a primary key to the new Qualification entity, the interviewer inquired about that, and Blake mentioned that maybe *traineeID* could be the primary key which was partially correct, but Blake seemed unsure. But before discussing the primary key, the interviewer asked whether a qualification should only belong to one trainee highlighting Blake's earlier incorrect assumption that the relationship was a one-to-many one between Trainee and Qualification. Blake considered this and then mentioned:

I guess not, if maybe more than one trainee went to the same institute to get qualified. Yeah, so most probably in this scenario, it'll be a many-to-many relationship, isn't it? And that makes more sense to track each trainee's qualification details using an associative entity.

Based on this discussion, Blake updated their logical model, as shown in Figure 8.28. The interviewer then commented on Blake's naming of entities, explaining that the names should be in singular form, not plural.

Blake found it difficult to explain whether the logical model he developed was in the 3NF or not. This may have been due to a lack of practice in similar questions. Nonetheless, the interviewer briefly explained how to decide whether the logical ERD is in the 3NF

8.6 Casey

8.6.1 Session 1: Analysing the case study and extracting business rules

Casey was one of the participants who struggled throughout the sessions with facing numerous challenges, but they persevered and completed the study. During the first session, Casey was the only participant who failed to complete both tasks during the allocated session time. From the practice session itself, where they spent nearly 30 minutes, Casey seemed to face barriers largely associated with their knowledge of database modelling concepts. In order to maintain the consistency of the narration, the practice or think-aloud familiarisation task is not added in the thesis however, the interviewer had to assist Casey while attempting even this task, which was designed to be very simple and doable in order to ease the participants into the process of think-alouds. When the allocated time was reached, the

interviewer stopped the session and postponed Task 2 until the next session, as Casey was overwhelmed with all the information they were trying to process. Casey completed Task 1 but made some errors, as indicated in Table 8.2. Casey attempted Task 2 in the next session and was outside of their ZPD most of the time. At a certain point in the session, the interviewer decided to provide the exact solution for one section of the case study since Casey was out of their ZPD, where they couldn't progress even with the interviewer's assistance. Afterwards, the interviewer's interventions managed to bring Casey back on track to complete the task.

Task 1: Narrative

Casey started Task 1 (see section 6.3.2) by reading the task preamble and then went on to read the first section of the case study. From the beginning, Casey annotated the text using an underline to highlight the *Trainees* and *Course participant* as potential entities. Casey continued reading and underlining the terms *traineeID*, *name*, *dob*, *address*, *contact number* and *qualifications*. Even though Casey annotated the potential database constructs, there was no symbolic distinction between entities and attributes. Casey moved on and underlined the terms *Trainee*, *referred*, *Course*, and *Employers* from the sentence that explained the attributes related to *Employer*. At this point, Casey underlined all the nouns and verbs in the sentence without consideration of which were relevant. However, for the attributes of the *Employer*, Casey continued differentiating between entities and attributes saying:

So I'm thinking these [pointing to the attributes of Employer] are the attributes employers need to have [underlined each], I'm going to put them within brackets as well.

Then Casey went on to the sentence, which described the relationship between the *Trainee* and *Employer* and underlined all the nouns (i.e., *Employer*, *Employee*, *Training course*, *Trainee*, *Course*). Further, they circled the terms *many*, *only*, and *one*, which appeared to be an annotation related to the cardinalities of the relationship, and Casey mentioned: "*I'm circling the process or relationship that I'm thinking is there*". Finally, Casey read the last sentence of the case study section, which provided further information on the same relationship and circled the term *few* while underlining the terms *Employer* and *not referred*. After circling the term *few*, Casey mentioned that they circled the terms, which could be the relationships, even though the term represented a cardinality. It appeared to the interviewer that even though Casey attempted to follow a systematic process, it was not consistent at all, but waited to see how the information would be presented. See Figure 8.29 for Casey's attempt to use annotations in analysing the first section of the case study.

Trainees: It is necessary to store details about all the trainees who participate in the courses. Each course participant is identified by their trainee ID and this is stored along with their name (composed of first name and last name), date of birth, gender, address (composed of street number, suburb, city, postcode), contact number. Along with these details the previously acquired qualifications of the trainees should be stored in the database. Since most of the time these trainees are referred to the course by their employers it is also necessary to store the details of the employers including the name of the company, address, and contact number. Each employer can be uniquely identified using the employerID. An employer can refer many employees to a training course and a trainee can be only referred to a course via one employer. However, there can be few trainees who are not referred by an employer.

Figure 8.29 Casey's attempt to annotate the case study.

Next, Casey decided to add the information into the provided space on the task worksheet as they referred back to the case study text. Casey created a heading for Entity, Attributes, and Relationships listed the *Course* under Entity followed by the *Participant* (see Figure 8.30). Then Casey drew arrows from the *Participant* to indicate its attributes and added *traineeID*, *name*, *dob*, *gender*, *address* and *phone number* as attributes. They questioned whether *first name* and *last name* should be added and listed them in brackets next to the *name* attribute. It was interesting to observe that Casey identified the importance of separating out the parts of the *name* attribute and replacing it with two separate attributes even though they didn't vocalise that it was a composite attribute. Casey then read the sentence where *qualifications* were mentioned and hesitated but then added the term to the list of entities. Moving on, Casey added the *Employer* also to the entity list and indicated *referred* under relationships (see Figure 8.30). Then they started reading the same sentence of the case study again and seemed a bit unsure, asking themselves: "Wait a minute, participants, can I do slash trainee?" Casey realised that *Trainee* and *Participant* represented the same entity and added the *Trainee* next to *Participant* in the entity list separated with a forward slash. Casey then added the related attributes to the *Employer* and continued to read the next line. After reading the line, "An employer can refer many employees ..." Casey muttered quietly to themselves as they thought about whether the relationship between the *Trainee* and *Employer* was unary but, in the end, did not record this and added *only referred* to the relationships list. Next, they connected the *only referred* term to the *Employer*, pointing the arrowhead to the *Employer* entity and connected the *Employer* to the *referred by* relationship where the arrowhead was pointed to the relationship name and added *1:M* next to the relationship name (see Figure 8.30). Casey read the last sentence of that section of the case study but appeared to ignore the cardinality described there, indicating they were finished.

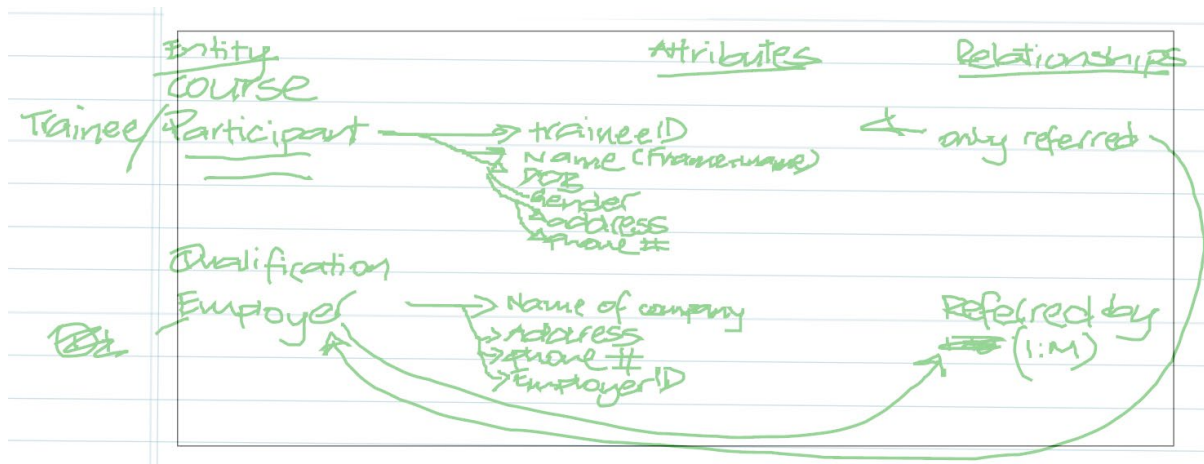


Figure 8.30 Information extracted from analysing the first section of the case study.

The interviewer questioned how they felt, and Casey mentioned it made them think about what they learnt in class while trying to extract the information. With this, the interviewer asked Casey to move on to the next section of the case study. Casey started reading the section and underlined *Course* along with the *courseID*, *name of the course* and *level of the course*. Moving on, Casey underlined the phrase *generate several programs*, followed by *Program* entity and its related attributes. At this point, Casey abandoned the distinguishing annotations of brackets and circles they had used earlier, using underlining for all components of the model. Then Casey added *Course* to the entity list and then its attributes, followed by the relationship *generate several programs* on the relationships list. Once the relationship was added, Casey listed *Program* as the next entity and went on to the sentence that described the nature of a *Program* in the problem description. It appeared that the term session, which was used to explain what a program was, confused Casey a bit, and they added *different sessions* as an attribute of the *Program* with a question mark followed by the rest of the attributes. Once they read the sentence “A program can have many trainees”, Casey drew a line between *Program* and *Traineer*, but it appeared they struggled when it came to the rest of the information related to the cardinalities and seemed to be adding the cardinalities randomly (See Figure 8.31). Next, Casey added the entity *Class* and its attributes and noted a potential relationship between *Class* and *Program*. However, they mistakenly linked *Class* to *Traineer*, and not noticing their error, Casey was satisfied with their solution and seemed eager to move on to the next section but mentioned: “*The pen gets quite heavy after a while, but otherwise, yeah, it's good. It's good.*” The interviewer offered Casey a small break before they moved on to the next section of the case study.



Figure 8.31 Casey's presentation of the relationships.

Casey started reading the last section of the case study and underlining all potential entities, attributes and relationships. Then Casey noted down *Instructor* as an entity and also its attributes. Once Casey read the next sentence, they realised there was a relationship between *Instructor* and *Program* and added *Program* to their list of entities. The sentence that expressed the relationship between a program and an instructor seemed to confuse Casey, and they read it multiple times. It should be noted that the sentence was quite lengthy and connected with a conjunction. Despite understanding that there was a relationship (see Figure 8.32), Casey could not comprehend the cardinalities on both sides, deciding in the end to ignore this aspect, deciding they were done:

Hmm... [pause] now how am I going to do that? So this instructor [connect instructor to program and program side put a 0 to the line] will be not. Yeah, I think I haven't read those, that symbols that they're showing us. So I don't know how to use them. Okay. I think I am done.



Figure 8.32 Casey's representation of the relationship between *Instructor* and *Program*.

Interestingly, despite making a few errors, Casey didn't seem to be struggling much. Casey did not check their answers before announcing that they finished the task. Hence, the interviewer didn't assist Casey during the interview and decided to discuss these issues during the retrospection. Casey's LT/P at the end of Task 1 is given in Figure 8.33.

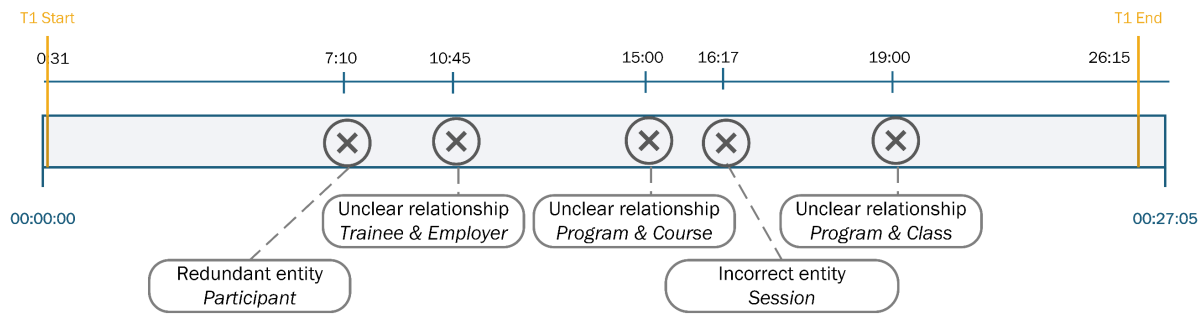


Figure 8.33 Casey's LT/P at the end of Task 1.

Task 1: Retrospection

During the retrospection, the interviewer explained what needed to be done when similar nouns were identified, like the case of the *Trainee* and *Participant* where one term needed to be removed based on the context. Then, the interviewer questioned Casey about listing *qualifications* as an entity in the first section. In reality, *qualifications* was a multivalued attribute that should end up being represented as a separate entity during the development of the logical model when the many-to-many relationship between trainee and qualifications should be resolved. The interviewer was interested to know whether Casey had made this leap in understanding the need to resolve the many-to-many relation. Casey attempted to explain their reasoning, but the explanation was very vague and imprecise:

Well, because of how they said it in here [pointing to the separate line which explained the qualification attribute] acquired qualifications. So then wouldn't the attributes come under the qualification? But it didn't mention any attributes, which is why I just thought, oh, maybe it's an entity then.

The interviewer's query led Casey to believe that their assumption was incorrect. Casey went on and explained that qualifications couldn't be an entity after all since only people can be entities – demonstrating an incorrect understanding of the definition of the term entity. The interviewer decided to explain what an entity was, and then Casey went on to explain their thinking further:

Because you know, when you get a qualification, you're not only learning one thing, you're learning several different things like database design and programming, and those are all attributes of the qualification.

It almost seemed they wanted to convey the correct idea but lacked the basic language and conceptual knowledge needed to express it properly. The interviewer explained to Casey that their instinct was correct and they could explore it further in the next session, but at this stage, they could safely consider the attribute to be an entity.

The discussion moved to the next section, and the interviewer queried why the *Session* was identified as a potential entity. Casey couldn't recall why they had included it and vaguely mentioned it because it was within brackets. The interviewer explained that the term session was used to express the *Program* entity further. Casey's tendency to translate the case study description in a way that treated each sentence in isolation and word-by-word, without considering as a whole the overall picture of the case study, to identify the modelling constructs led to this incorrect entity.

The interviewer then directed Casey's attention to the relationship between the *Trainee* and *Program* entities. Casey stated:

"Oh yeah that was hard. I don't know about that one. Yeah. And then that just kept going back from entity to entity and I was like ... it was confusing"

The interviewer realised that it was because Casey didn't consider the relationship in both directions. She helped Casey to identify the relationship between *Trainee* and *Program* and how to present the cardinalities correctly. After some time, Casey seemed to grasp the concept, and the interviewer asked them to recheck the relationship between the *Instructor* and the *Program*. Again, Casey read the sentence aloud, which stated, "... some programs can have more than one instructor assigned to them based on the number of trainees allocated for the program", which confused them since they felt the trainee should also be included in the relationship between the *Instructor* and the *Program*. The interviewer explained that misunderstandings could happen when students translate each and every word one by one without considering the overall scenario. Casey agreed that this was something they tended to do.

The interviewer suspected that Casey used to seeing shorter and simpler case study descriptions where the information was provided directly without being obscured by extra and redundant information. However, with the interviewer's assistance, Casey was able to interpret the relationship correctly. At the end of the retrospection of Task 1, the one-hour set for the session was over despite Casey not completing the task. The interviewer decided to ask Casey whether they wanted to continue since they appeared to be overwhelmed, and Casey confirmed, saying:

If I'm going to go, for today maybe it'll be too much information to process. But at least this gave me a fair idea of how to start work on my case study in the assignment.

8.6.2 Session 2: Analysing the case study and extracting business rules

As decided, Casey's Task 2 was conducted in a different session. Unfortunately, this was when the sessions were transitioned into an online mode. Hence, there was a longer time period between attempting tasks 1 and 2 than for the other participants. However, Casey's solution for Task 1 was

provided ahead of the session via email. Casey opted to use MS Word to develop the business rules. At the beginning of the session, the interviewer had to spend some time explaining the task. They also had to show Casey how to use the MS Teams functions, particularly on sharing their screen, as Casey was unfamiliar with the software.

Task 2: Narrative

Casey started by creating a table with syntax elements as the headings for each column to structure writing their business rules, which was a similar process used during the lecture. They sequentially added *Course*, *Trainee*, and *Employer* to the Entity1 column, as indicated in Figure 8.34.

Entity1	MAY/MUST	Relationship verb phrase	Number	Entity2
Course		o		
Trainee				
Employer	May	Be referred by	1:M	

Figure 8.34 Casey's first attempt at writing the business rules.

With the interviewer's reminder to think aloud, Casey mentioned that they were referring to their solution from the previous session to identify the entities specified in section 1 of the case study. Casey started adding the following: *Employer may be referred by 1:M* but appeared to be struggling to determine the second associated entity to complete the business rule. Also, it seemed that the presentation of their answers to Task 1 might be causing issues since Casey found it difficult to interpret and read their Task 1 solution. Casey mentioned that:

This is too hard to figure out what is where. I might have to read the whole thing again...

Casey struggled for a while, going through their previous answer to no avail, and the interviewer intervened to direct Casey to section one of the case study and read it, mentioning that the relationship between the *Trainee* and *Employer* can be found in that section. With this focus, Casey managed to identify the sentence that described the relationship between the *Employer* and *Trainee*. They then changed the earlier written business rule given in Figure 8.34 as follows: Employer may refer 1:M employees (see Figure 8.35). However, Casey was referring to *Trainee* as *Employee* in the business rule. Casey then moves on to complete the business rule in the other direction as follows:

Trainee may be only referred by employer.

Casey struggled to figure out the missing cardinality/number. The interviewer realised that Casey was not reading the adjoining sentence where this information was presented. It seemed Casey struggled with sentence structures, particularly when conjunctions are used in a sentence. Casey stopped reading and was silent, and the interviewer prompted them to read the next line in the case study. After reading the sentence multiple times, Casey added another business rule, as highlighted using a red box in Figure 8.35. It seemed they were unaware that they had just duplicated the existing business rule in the second row of the table between the *Trainee* and *Employer*, albeit with trainee expressed in plural form.

Entity1	MAY/MUST	Relationship verb phrase	Number	Entity2
Course				
Trainee	May	Be only referred	1	Employer
Employer	May	Refer	1:M	employees
Trainees	May	Not be referred by	few	employer

Figure 8.35 Newly added business rule in the red box.

At this point, it was evident that Casey was not systematic and continuously tried to develop business rules in an ad-hoc manner, not realising that each relationship needed to be expressed using two business rules, one for each direction of the relationship. Further, it seemed that Casey was reading the sentences superficially and lacked focus:

A trainee can be referred hmm...Now I'm lost where I was reading... hold on I'm looking for where I left off my reading.

Since Casey was unable to independently progress successfully and none of the hint-based assistance provided had worked, the interviewer decided to stop working on the main task and asked Casey to try the redirected task (T2.1 – see section 6.3.2). This was a simple fill-in-the-blank exercise based on section one of the case study with a list of options provided for the missing parts of the business rules. Casey started reading the task and completed the first rule correctly. However, Casey had trouble working out the cardinality when the business rule was written in the opposite direction despite being scaffolded with a list of options (See Figure 8.36).

1. Each __ **Employer** _____ may refer _ **one or more** _____ TRAINEES.
2. Each TRAINEE may be refereed by _____ EMPLOYER.

Figure 8.36 Casey's attempt on the redirect task – T2.1.

The interviewer intervened and directed Casey to read the last two lines of that section of the case study again in order to identify the correct cardinality. After much deliberation, Casey commented:

I'm thinking that I should use Entity 2, but I can't because Entity 2 is already there. So I wonder if that might be a number, something to do with the numbers. So each trainee may be referred by..., but it doesn't make sense if I said zero or one or if I said one or more, it doesn't make sense.

It seemed Casey was experiencing issues in transferring information between different forms, along with a high intrinsic and germane cognitive load (Sweller, 1988). The interventions provided by the interviewer were not helping Casey to move forward. The interviewer decided to explain how to transfer information from the natural language to the grammar of the business rules using the redirected task (T2.1) as an example.

The interviewer breaks down the sentences explaining the relationship word by word to emphasise how cardinality can be decided. The interviewer also explained the use of the terms may/must in writing business rules. Casey stated that after the explanation, it was much clearer to them. Figure 8.37 illustrates Casey's LT/P to the point where the interviewer decided to redirect them to a simpler task focusing on cardinalities.

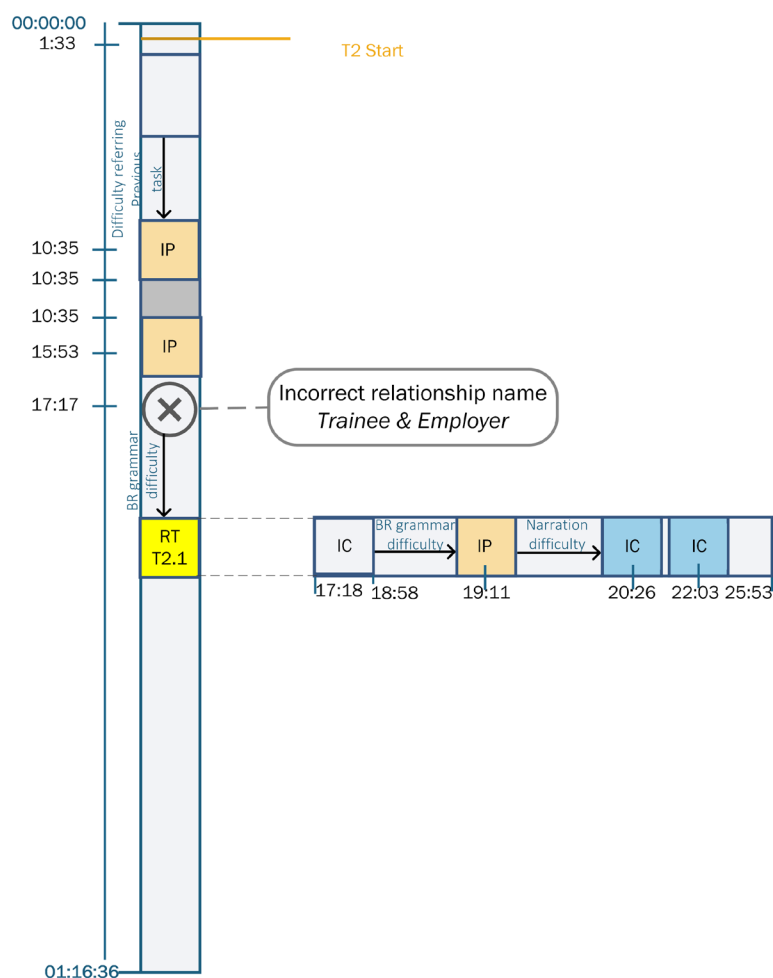


Figure 8.37 Casey's LT/P up to T2.1.

With this, the interviewer asked Casey to continue with their original task by restarting it since they had made no real progress. Casey once again was faced with the same difficulties as before and showed a lack of focus on the case study narrative and skimmed it in an ad hoc manner. Casey was aware of their approach but seemed to believe that it was effective because of the nature of the task, mentioning:

I'm just jumping from line to line. I'm not actually going in any order, so I could be on line two and then all of a sudden, I jumped to line eight. I'm looking for keywords rather than the attributes such as you know, like Suburb, City, postcode. I don't need to read that because I'm looking for keywords here, not attributes.

Moreover, Casey demonstrated a lack of understanding of the basic terms and concepts needed in order to be able to develop business rules. Despite explicit instruction during their attempt on the redirected task and Casey seeming to have understood when returning to the main task, Casey was unable to form the correct business rules on their second attempt (see Figure 8.38).

	Entity1	MAY/MUST	Relationship_verb_phrase	Number	Entity2
	Trainee	May	Only be referred	1	course
Each	Employer	May	Refer	1:M	course
	Trainees	May	Not be referred by	0 - M	Employer

Figure 8.38 Casey's second attempt at writing the business rules

The interviewer stopped the task and decided to discuss the business rules for section one, as Casey couldn't progress after a considerable amount of intervention for almost 30 minutes of the session. The interviewer provided the solution in full to Task 2 (i.e., business rules for section one of the case study) and explained the process they went through in order to answer the task. The interviewer also explained the relationship in both directions while using crow's foot notation on a whiteboard. Finally, Casey mentioned they understood what it meant to write the business rules in both directions.

After this, Casey started to read section two of the case study and provided the business rules for the relationship between the *Program* and *Trainee* correctly. Almost 45 minutes into the session, this was the first instance of Casey showing evidence of progress and learning. There was a slight error in using *may* instead of *must* in one of the business rules, and with a quick prompt from the interviewer, who asked Casey to double check on the related cardinality, Casey realised their mistake and fixed it. Then Casey moved on to read the sentence that detailed the relationship between *Program* and *Class*, saying:

Hey I don't have Class. I don't have Class as an entity. Can I add it?

Casey had difficulty understanding their solution to Task 1 due to its inconsistent representation of the information and was adamant that they didn't identify *Class* as an entity despite the fact they had actually done so in the first session. However, Casey went on to add the business rule between *Program* and *Class* without assistance and seemed to be in more control of what they were doing. Casey started to move on to the last section but missed developing the relationship between the *Course* and the *Program*. The interviewer informed Casey that they had missed a sentence in the case study. Casey managed to catch onto their omission and added the business rule, which describes the relation from *Course* to *Program*. Interestingly, they were able to make a correct assumption, which was required as the term *several* is imprecise when writing one of the business rules:

Uhm, each Course must generate several... so that's one wher we don't even know what several means. So let's go with 1 to many and that's it.

However, Casey forgot to add the business rules that describe the relationship in the other direction from *Program* to *Course*, but when the interviewer prompted whether the business rules were completed, Casey realised the mistake and added the business rule.

Finally, Casey moved on to the last section despite exceeding the allocated time for the interview and quickly developed the next business rule, expressing the relationship between the *Instructor* and the *Program*. Yet again, Casey mentioned:

So yeah, because I see here it says an instructor may be assigned to teach any number of programs which I've got, and then this is where it starts to play games on me where its says while some programs can have more than one instructor assigned to them based on the number of trainees allocated for the program. Now it's starting to confuse me because there is so much information in there that I need to decide which keyword to use. But it's like extra information as you said earlier, isn't it? I'll continue.

Despite their confusion, Casey managed to complete the task with the assistance provided by the interviewer. Casey's LT/P at the end of Task 2 is given in Figure 8.39.

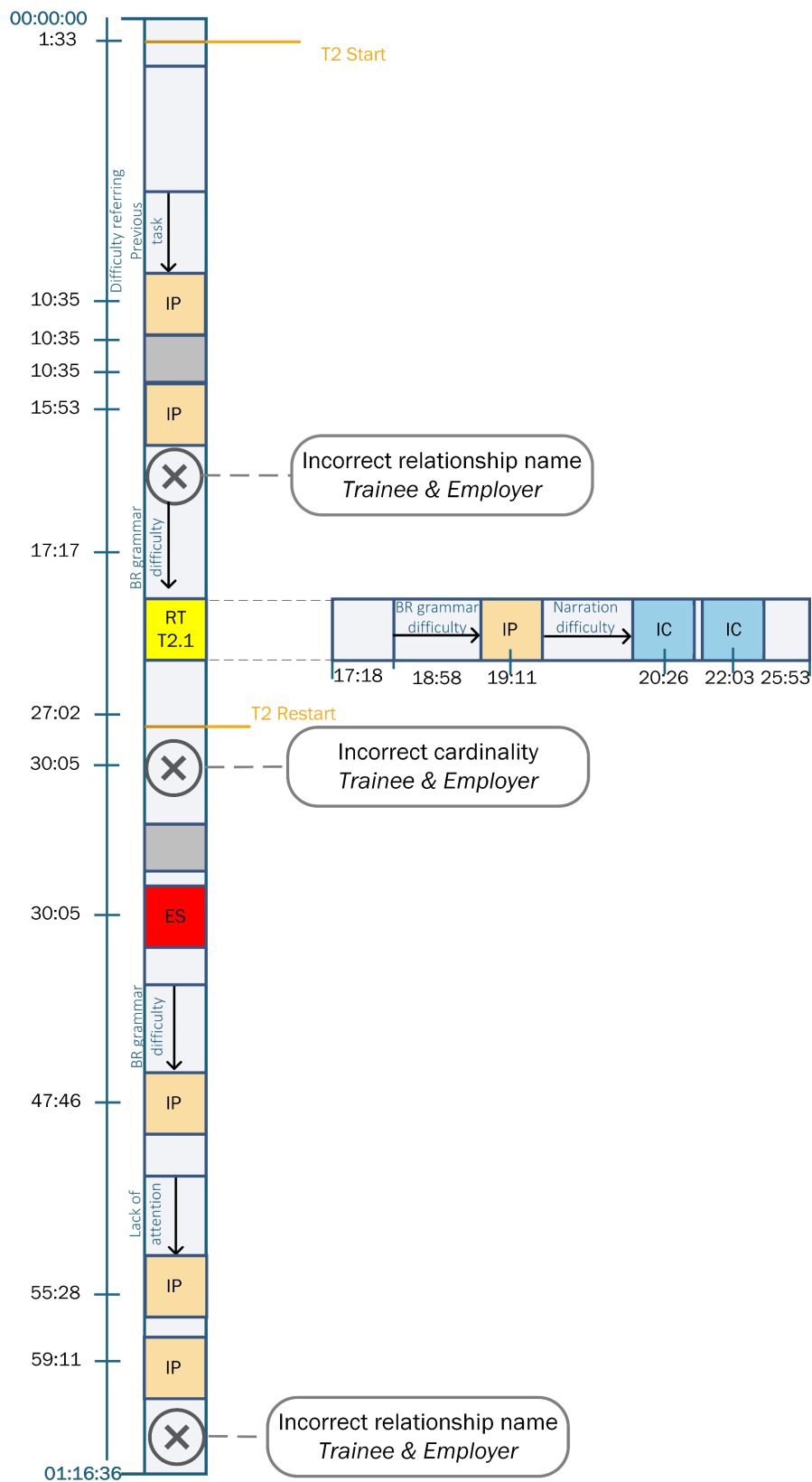


Figure 8.39 Casey's LT/P at the end of Task 2.

Task 2: Retrospection

As the allocated time had passed, a separate retrospective interview was not conducted with Casey. However, since a lot of time was utilised during the session to discuss the issues, the interviewer decided that it was not required. The interviewer pointed out some minor errors in the business rules (i.e., incorrect use of may/must or use of incorrect relationship name (i.e., be generated if they are run), which Casey understood. The interviewer provided the option for Casey to contact them if they needed any further help.

8.6.3 Session 3: Developing a conceptual model

Session 3 was comprised of only one task – Task 3 (see section 6.3.3). In this task, the participants were asked to develop a conceptual model based on the case study. Like the other participants, due to COVID-19 lockdowns, Casey did this task online. Since Casey didn't own a tablet, they opted to use the Visual Paradigm tool to develop their model, and their screen was shared via MS Teams.

Casey managed to complete this task within the time limit. But like Task 2, the interviewer had to assist Casey throughout the session in order for them to progress. Casey's fragile knowledge of the concepts related to relationships and cardinalities once again was the source of much of their difficulties. One notable observation was that they were unable to transfer their business rules into the conceptual model despite having access to the correct business rules from the previous session. It is reasonable to assume that the added cognitive strain from using the Visual Paradigm meant that they were even less able to recall and apply their fragile knowledge than they could during Task 2. However, with the interviewer's assistance, they completed the task despite making several mistakes along the way (see Table 8.2).

Task 3: Narrative

Once the interviewer explained the task and helped Casey share their screen, Casey started the task. Casey was provided with the solutions to their previous tasks at the beginning of the session. Casey mentioned that they would refer to their solutions to Task 1 to develop the conceptual model and keep both the tool and solution open on the screen. Casey added the *Trainee*, *Employer* and *Course* entities to the ERD in Visual Paradigm. Then they appeared to be somewhat confused for a while and mentioned:

So I'm scrolling through this [while scrolling through their solution to Task 1], and it looks too messy, hmm...

As Casey struggled to proceed in identifying the entities, the interviewer directed Casey to refer to the solution of Task 2 (business rules) instead of Task 1 to identify the entities. This allowed Casey to

proceed and indicated that they would add all the entities first and would refer to the solution of Task 1 when adding the attributes. After that, Casey added the *Program*, *Class* and *Instructor* entities. Casey decided to verify the added entities with the business rules to ensure they had not missed any. Then Casey queried whether there was an appropriate order to add the constructs. The interviewer told them just to do what they would normally do, and Casey decided to proceed by adding the relationship between the *Course* and the *Trainee*. After reading the relevant business rule, Casey selected the required relationship and dragged the mouse pointer from the *Trainee* to the *Course*. However, when dragging the mouse pointer, Casey mistakenly deleted the name of *Course* entity, and as they were trying to rename it, the tool did not allow it, which led to the relationship be not added. Casey wondered whether this was since they attempted to add relationships first and asked whether to add the attributes next, and the interviewer again clarified they could follow any order they liked. Then, Casey went on to add further attributes instead of continuing with the relationships.

Casey referred back to Task 1 and successfully managed to add all the attributes for the *Trainee* entity, including the primary key. However, Casey missed the *qualifications* attribute despite it having been discussed during the last retrospective interview. Next, Casey moved on to add the attributes for the *Course* entity but seemed to get confused: “*That’s different sessions, what’s different sessions where does that come from?*”. The notion of sessions had also been discussed with the interviewer in the last session. Casey was unable to recall this even though they had added a cross sign next to where the session was added during their last retrospection. Casey decided to read the section of the case study which mentioned sessions but was distracted from the sessions due to being hesitant about whether the *courseID* was a unique key since it was not mentioned as a unique key in the case study. The interviewer questioned them on how they would handle such cases normally. Casey mentioned that they would assume that was the key and marked *courseID* as the primary key. Casey then added the attributes for the *Program*, *Employer*, *Instructor*, and *Class* entities.

Finally, Casey decided to add the relationships and referred back to the business rules they developed. Casey added a one-to-one relationship between the *Trainee* and the *Course* entities. It appeared that Casey did not realise that they were using the incorrect business rules they had developed in their first attempt, even though the incorrect business rules were marked separately in their answer sheet and were also discussed during the last session. Hence, the interviewer queried Casey with the intention of making them realise their mistake and prompted Casey to read the first section of the case study to confirm whether there was a relationship between the *Trainee* and the *Course*. Casey read the section to understand the issue and decided to remove the relationship and add the relationship between the *Employer* and *Trainee*, still referring to the incorrect business rules. They were unsure about their solution, and the interviewer mentioned that they were referring to the incorrect business rules they had developed earlier in the previous session. Casey’s LT/P to this point is given in Figure 8.40.

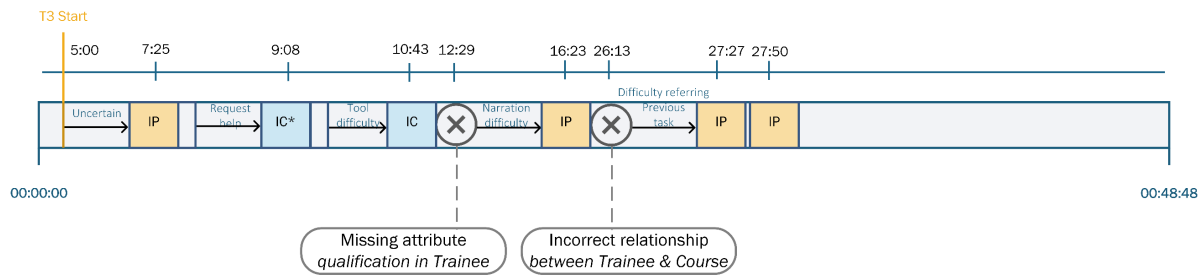
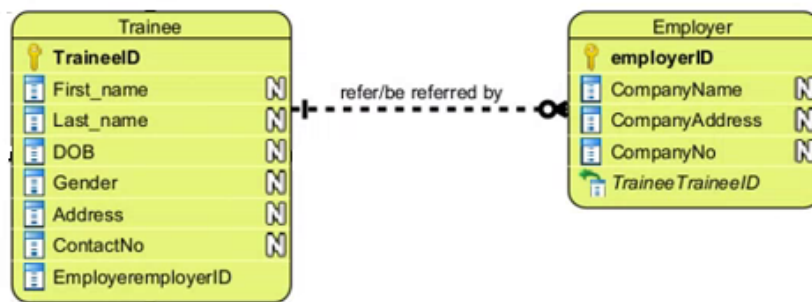


Figure 8.40 Casey's LT/P during the first half of the session.

Finally, Casey managed to identify the corrected version of their solution to the previous task and added the relationship between the *Trainee* and the *Employer*. However, Casey added the relationship in the wrong direction from *Trainee* to *Employer*, creating an incorrect relationship and a foreign key that didn't resemble the requirements of the developed business rules (see Figure 8.41). Casey did not notice this error, and the interviewer ended up hinting that they should crosscheck their model with the business rules. It appeared to be a genuine mistake in using the tool since Casey realised the issue immediately:

So it says each employer... Oh no, the crow [pointing the cursor to the cardinality] is not where I want it to be. I want to change but don't know how to. How do I change that?

Casey asked the interviewer to explain how the cardinalities could be changed using the Visual Paradigm tool. However, Casey didn't realise that the foreign key was still incorrect and the cardinality of the employer side was one instead of zero-to-one despite these aspects being correct in the business rule. Throughout, it was quite evident that Casey had issues transferring information from one form to another. It is possible that the inability to transfer information was due to the additional cognitive load brought by using Visual Paradigm. However, similar issues were observed in the sessions in which Visual Paradigm was not used, and it is more likely that Casey tended not to pay attention to detail. Casey seemed to prefer to skim-read the specification and start on the task without fully understanding the modelling problem.



Each	Emp	May	Refer	0 or more	Trainees
Each	Trainee	May	Be referred by	0 or one	Employer

Figure 8.41 Incorrect relationships vs business rules.

Then Casey moved on to the relationship between the *Trainee* and the *Program*. Before adding the relationship, Casey moved the entities here and there on the canvas, and it seemed like they were trying to manage the layout properly. Casey used the tool correctly this time and added the relationship in the right direction. However, they still didn't change the cardinality of the *Trainee* side of the relation from zero-to-many to one-to-many (the relationship default is visual paradigm is zero-to-many), even though a similar scenario had been discussed in the context of the Trainee-Employer relationship. Casey did not attempt to verify their model so far against their business rules. Instead, they continued to move on to add the relationships between *Program* and *Class* and *Course* and *Program* entities. Again, Casey accepted the default cardinalities added, which did not map to their business rules. Next, Casey added "be generated if run" as the relationship name between *Course* and *Program*, which was not an appropriate verb that could explain the relationship between the entities, and this was discussed during the previous retrospection. The interviewer asked whether the relationship "name" was correct to prompt them to realise the issue, but Casey answered, stating it was a many-to-many relationship, which was not the answer to the question. With the interviewer's further explanation, the interviewer went ahead and explained the issue with the relationship name given by Casey, which included the condition (i.e., if run) that was not required. Casey realised the mistake and changed it to a suitable verb phrase.

Finally, Casey added the relationship between the *Instructor* and the *Program*. However, they seemed hesitant to add the relationship and, when asked by the interviewer, indicated they were unsure about which business rule to use since there were two. The interviewer asked them to look at the business rules in both directions since that appeared to be the source of confusion. But, Casey still read only the first business rule and added a one-to-many relationship instead of a many-to-many despite the prompt

of the interviewer and announced that they had completed the task. Casey's LT/P at the end of Task 3 is provided in Figure 8.42.

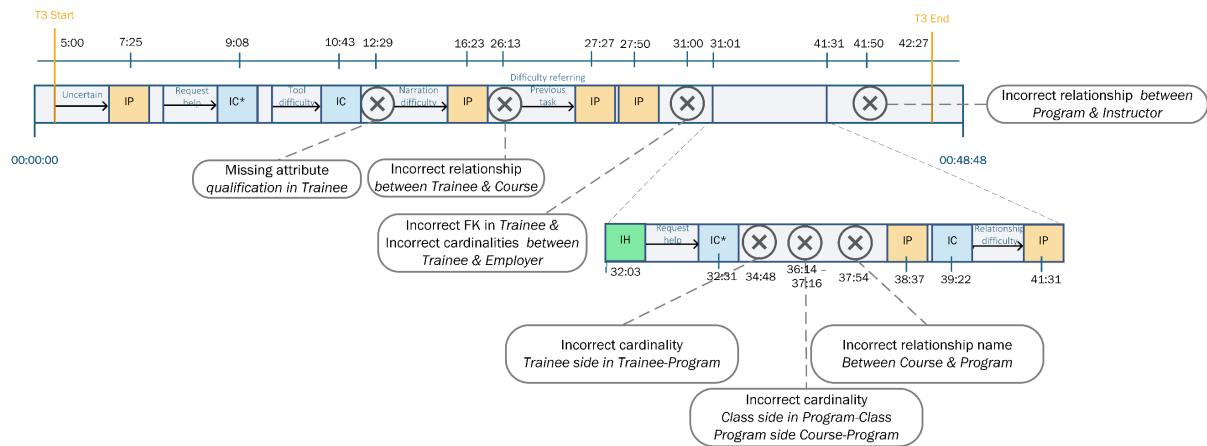


Figure 8.42 Casey's learning trajectory at the end of Task 3.

Task 3: Retrospection

During the retrospective interview, the interviewer started reviewing the model with Casey and asked them to read the business rule between *Program* and *Class* and compare it with the model. Casey was able to find the error in the model and amended the cardinality correctly. The interviewer explained why the error had happened and that it was probably due to the way Visual Paradigm adds the same default cardinalities, which Casey did not correct. The interviewer asked Casey to double-check the cardinalities of all the business rules. But, it seemed Casey was not interested in verifying the model there and then and mentioned:

Yes I get it, so that's all that and I've saved the model. Shall I send it to you?

It appeared they wanted to end the session and had enough after 45 minutes. However, the interviewer asked Casey to check the relationship between the *Instructor* and the *Program*. Again, Casey managed to figure out the issue with the cardinalities and corrected it. The interviewer pointed out the other cardinality issues and instructed Casey to always verify the conceptual model alongside the business rules. This time, Casey managed to fix most of the issues, apart from the cardinalities between *Trainee* and *Employer* relationship. Overall, Casey had an issue transferring information from business rules to the conceptual model. Interestingly, this appeared to be a recurrent scenario where, in the earlier session, they had difficulties transferring information from the case study to modelling constructs. See Figure 8.43 for Casey's ER model, highlighting the errors and corrections made during the retrospection. At the end of the retrospection, Casey admitted that they need to practice more to get used to the syntax

and use them correctly in the tool but mentioned: “I learnt something new. Thank you so much for that.”.

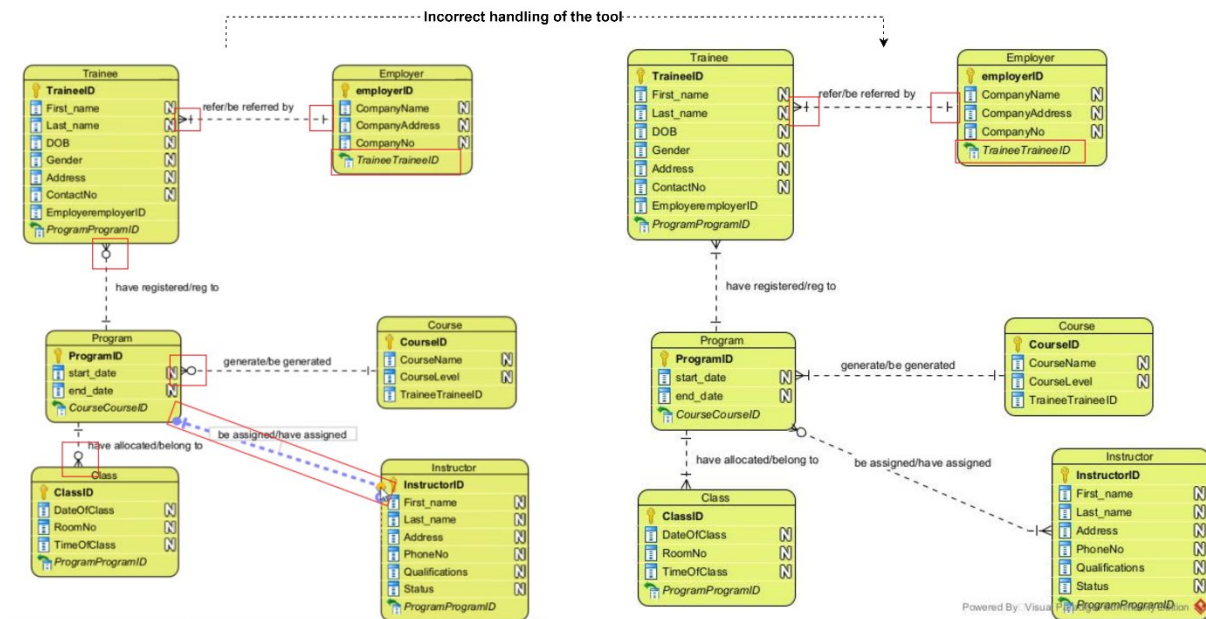


Figure 8.43 Casey’s conceptual model before (left) and after(right) the feedback. Red boxes indicate the errors in their solution.

8.6.4 Session 4: Transforming the conceptual model into a logical design

This session consisted of a single task (Task 4 – see section 6.3.4), in which the participants were asked to transform their conceptual model into a logical design.

Casey took more than an hour to complete this task, and despite completing it, they encountered several barriers to progression. One issue (missing *qualification* attribute in *Trainee* entity) they had was a repeating mistake that had also been made during their previous think-aloud session (see Table 8.2). Casey again introduced errors due to their use of the Visual Paradigm. However, the main difficulty Casey appeared to have was related to their fragile knowledge of database modelling concepts. Casey also had difficulty recalling and applying the mapping rules taught in class that are used to guide the transformation of a model from one form to another through the database modelling process.

As in Casey’s previous sessions, in this session, the interviewer had to provide assistance numerous times throughout the session in order to see Casey’s progress. Throughout the study, Casey showed a tendency to request assistance. They asked for help rather than trying to work through the task themselves. It seemed in Task 4 that they were overwhelmed by the task and did not know how even to begin.

Task 4: Narrative

The interviewer provided the solution Casey developed in the previous session before the session began and explained the task. It seemed right from the start that Casey was confused, mentioning:

That's right, so in order to transform them into a logical design, I have to start off with those primary keys and then break them down, like think I can remember.

The interviewer, at this point, assumed that Casey was referring to normalisation. Due to this, the interviewer explained what was expected of this task and how it differs from normalisation, but Casey was confused. As a result, the interviewer intervened and directed Casey to a simple redirected task (T4.1 as discussed in section 6.3.4) and asked whether they could recall the mapping rules required to transform the conceptual model into a logical one. T4.1 was asked with the intention of recalling the mapping rules they were taught during the class. Casey still seemed unsure and started referring to their notes but was silent for a while without talking out loud. As they were not progressing at all, the interviewer again asked whether Casey could recall the mapping rules related to many-to-many relationships since this was a significant rule demonstrated during the lecture and had appeared in several practice exercises in the lab classes as well. That appeared to trigger their memory, and Casey mentioned it was what they had explained earlier at the beginning, where the interviewer thought they were explaining the steps in normalisation. It seemed despite having incomplete knowledge, Casey could recall what they had learnt but did not have the language needed to express the ideas coherently. With some further discussion, it became clear that Casey was indeed having trouble with the terminology, and this lack of knowledge was leading to, at least to some extent, their confusion. The lack of terminology was also exacerbated by their lack of conceptual knowledge, which was quite fragile. This meant that Casey's explanations were rather vague. As the interviewer clarified the mapping rules using the lecture materials that Casey was somewhat familiar with, Casey mentioned:

The logical, the conceptual it's really hard to distinguish you know like the meaning behind every term.

The terminologies seemed to add further cognitive strain for Casey when attempting the task. The interviewer then provided a hint and asked Casey to start from a single entity at a time to resolve the attributes before moving into the relationships and directed them to restart the task. Casey's LT/P, till they were asked to reattempt the task, is provided in Figure 8.44.

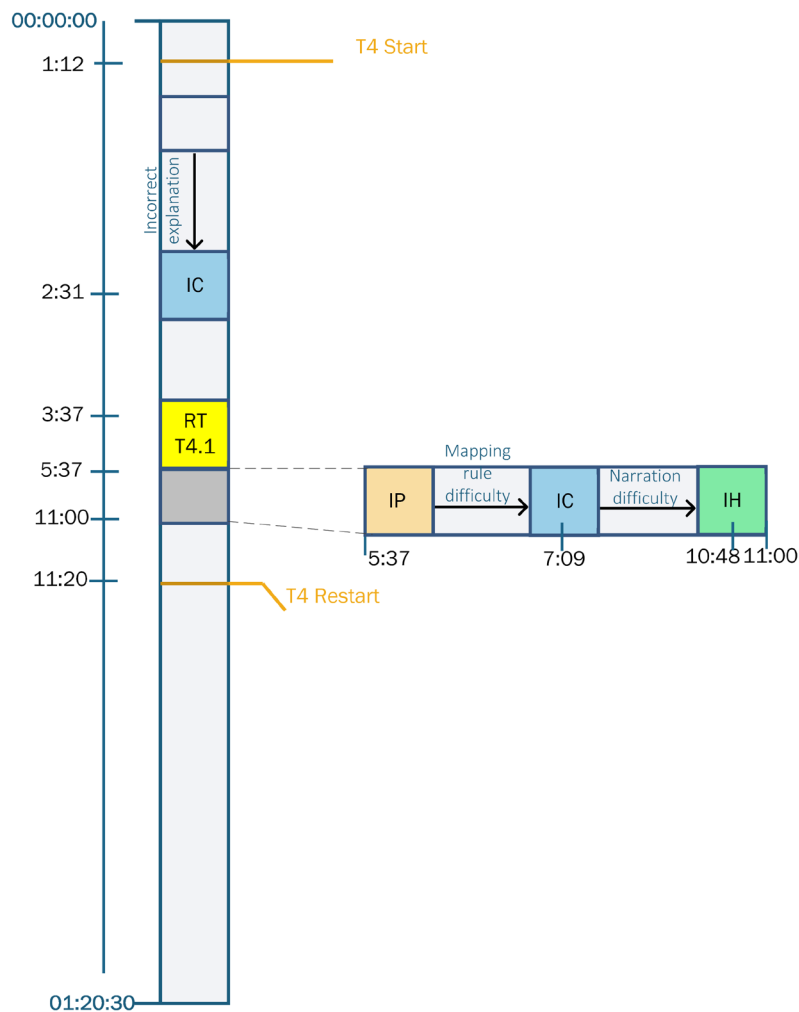


Figure 8.44 Casey's LT/P during their first attempt on Task 4.

Casey started the task again however, it appeared that they still hadn't caught up to the instructions provided for the task and started the task on pen and paper. The interviewer had to intervene and clarify that they had to use the Visual Paradigm to complete the task, mainly due to conducting the think-aloud online and being unable to "see" what Casey was writing otherwise. There is a probability that Casey was trying to avoid the Visual Paradigm as it caused them issues previously.

Casey could not understand what the Task was asking them to do. They did not know what it meant to transform the conceptual model (ER model) into a logical one (relational model) by applying appropriate mapping rules. As in Casey did not understand how to map model constructs such as multivalued and composite attributes and relationship cardinalities (e.g., one-to-many, many-to-many) to a relational model.

The interviewer decided to intervene. She took an example from Casey's conceptual model developed in Task 3, the *address* attribute of the *Trainee* entity and discussed with Casey why it needed to be

resolved and how to do it. She then took another example for relationships, of a one-to-many relationship between Trainee and Employer, and asked Casey to check it against the mapping rules. Finally, Casey appeared to be clear on what was meant to resolve the constructs in the conceptual model. Casey seemed to be a bit overwhelmed at this point and mumbled, “*That’s a lot, it’s a lot to do.*” – understandably, since it was more than 18 minutes to the task and they were still struggling to start it.

Finally, Casey opened their previously developed conceptual model (Task 3) in Visual Paradigm and asked the interviewer how to edit their existing model to resolve the composite attribute (*address*) using Visual Paradigm. The interviewer instructed that they could delete the existing attribute and add new rows to indicate the components of the address (i.e., street, city, postcode), and with that explanation, Casey successfully resolved the *address* attribute in the *Trainee* entity. Interestingly, they recalled the feedback they had received for one of their assignments on a similar task where they were asked to maintain a consistent naming pattern (e.g., First_Name, Contact_No) for attributes and apply it correctly.

Once Casey confirmed that they believed that the *Trainee* entity had been resolved, they moved on to the *Employer* entity. At this point, somehow, Casey began referring back to the case study rather than their conceptual model. Casey was the only participant with such an approach where, instead of transforming directly from the conceptual model, they resorted to using the case study again to guide the transformation. It appeared that Casey could not read and understand the previous model developed in Task 3 and was required to read the case study instead. Even after engaging with the case study for approximately three hours over three sessions, Casey struggled to locate the information about the *Employer* entity and seemed incoherent about the whole situation:

Employer ... How come I can't find my employer? Just going to highlight the ones that are in bold [pointing to the section names of the case study]. So instructors, courses, trainees... Hey, I don't have an employer anywhere?

The interviewer decided to assist Casey again, prompting that the employer information could be found in the first section of the case study. Casey finally identified the required information describing the *Employer* and *Trainee* relationship. While reading these sentences, Casey again struggled:

... Employer can refer employees to a training course, and a trainee can be only referred to a course through one employer. How do I work that one now?

Despite having discussed this same issue (i.e., literal or word-to-word translation) in the previous session, Casey was unsure how to interpret the information. It appeared that Casey often had difficulty referring and understanding their own solutions and recalling learnings and discussions from prior sessions. Again, Casey asked for help from the interviewer. The interviewer went through the sentence

Casey mentioned and explained how the sentence describes the relationship between *Employer* and *Trainee* while *Course* was used as a term to explain the relationship. With the provided clarification, Casey managed to recall the issue from the previous session. After this, Casey decided that the *Employer* entity had nothing to resolve, but they missed a mistake they had made during Task 3 that had not been resolved. Namely, the cardinality on the *Employer* side of the relation was labelled one instead of zero-to-one. The interviewer attempted to draw Casey's attention to the issue by asking them to read the sentence from the case study related to this cardinality and to check that against the model they developed for Task 3. Finally, Casey realised their mistake and made the necessary changes to the model.

The interviewer reiterated the importance of understanding and reading relationships in both directions. At this point, Casey appeared somewhat overwhelmed and mentioned: "*It's making my head spin with everything*", but decided to move on even though the interviewer encouraged them to take a small break before continuing. Casey's LT/P halfway through the session is given in Figure 8.45.

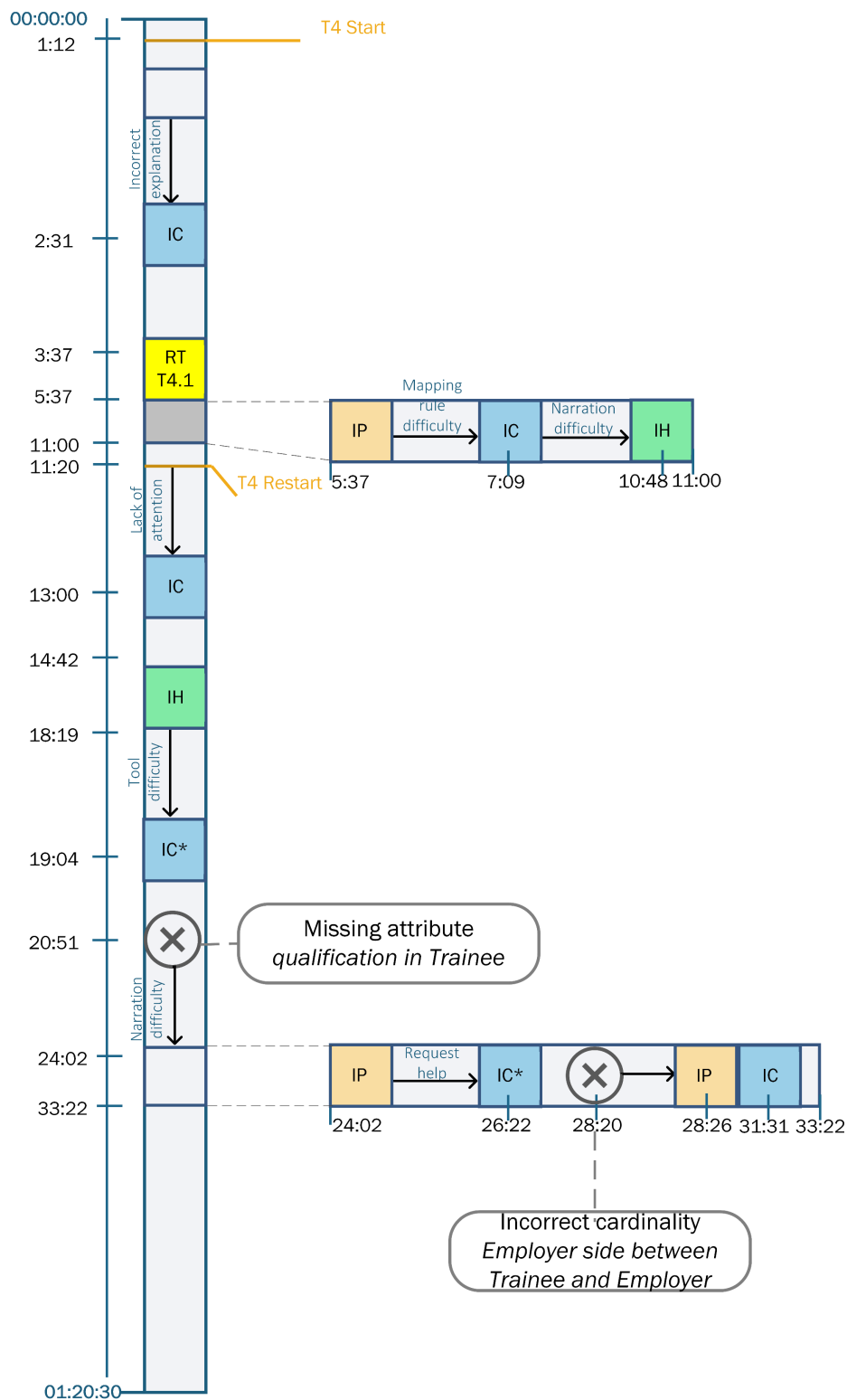


Figure 8.45 Casey's LT/P halfway through the session.

Next, Casey moved on to the *Program* and *Class* entities and made a few changes to the attribute names by renaming them according to the naming conventions they recalled and otherwise decided there was nothing to resolve. Casey moved on to the *Course* entity. They mentioned that the *course level* attribute

was a composite attribute since the case study described possible course levels such as introduction, certificate and diploma. Casey did not realise that these were simply possible values for the attribute and did not mean that the attribute was a composite one. This clearly indicates that at this point, Casey was still struggling with basic concepts, and it was no wonder they struggled to transform the conceptual model as such a task requires knowledge of several interrelated concepts to proceed. Hence, the interviewer decided to clarify why the *course level* attribute was not composite. Casey continued on to look at the relationship between *Program* and *Course* and mentioned that the *programID* should be a foreign key in *Course*. This was incorrect, and yet they had dealt with the foreign key for *Program* correctly in their conceptual model. This further evidenced Casey's inability to read and interpret their own conceptual model correctly. The interviewer decided to intervene and asked about the mapping rule related to one-to-many relationships. This triggered Casey to recognise the error in their logical model and fix it, move on to the other entities, and finally mention that they were done. However, Casey did not bother to check the remaining relationships to make sure that they were resolved, and the interviewer directed Casey to consider them. Even then, Casey was quite uncertain and struggled to recall the mapping rules they learnt and apply them when transforming the conceptual ERD into a logical one.

Considering that Casey was making no progress and clearly outside of their ZPD (almost 50 minutes into the one-hour session), the interviewer considered terminating the session to provide the exact solution and asked Casey if they wished to continue. Casey was determined to continue, so the interviewer gave them a redirected task (T4.2 detailed in section 6.3.4). T4.2 was a simple task where Casey was asked to explain how to resolve the many-to-many relationship between student and class entities provided in a partial ERD (see Figure 6.9) and where they would place the grade attribute if they want to store the grade students receive for each class. Casey had some idea but was unable to explain it properly and mentioned that the grade should be an attribute in the student entity, which was incorrect. The interviewer had to explain why the grade should be placed in the new associative entity instead of placing it in the student. She went on to explain the concept using the same example provided in the lecture materials using ERD notations, as Casey seemed to understand more with visuals. Casey thanked the interviewer for the explanation and indicated:

I always wondered how did they get this [pointing to the composite primary key in the associative entity] because I never got that. But thank you I got it now.

Finally, Casey decided to resolve the remaining relationships in the conceptual model and looked at the one-to-many relationship between *Program* and *Trainee*. Casey started explaining how to resolve it by adding a new entity, which was incorrect. The relationship was not a many-to-many relationship and therefore did not need to be resolved. Instead, it was a one-to-many relationship where they had to

ensure the correct placement of the foreign key (related mapping rule –primary key on the one side becomes a foreign key on the many side). The interviewer prompted Casey to check the notes they made earlier related to the mapping rules they had learnt in class. With the help of the interviewer, Casey finally managed to identify that the one-to-many relationships were already resolved since the foreign keys had been added automatically from the tool in cases where the relationship was added to Visual Paradigm in the right direction. It seemed Casey had a major issue constructing new knowledge in their working memory, disrupting how the knowledge structures or schemas get stored in the long-term memory. This appeared to be why Casey struggled with the mapping rules throughout the session. Finally, Casey managed to resolve the many-to-many relationship correctly despite requiring some assistance from the interviewer to handle the tool. Their completed logical model and the LT/P at the end of Task 4 are given in Figure 8.46 and Figure 8.47, respectively.

Task 4: Retrospection

There was no separate retrospective interview for this session, as the allocated time had already passed, and most issues were discussed as the task was attempted. The interviewer decided Casey was already overwhelmed and overloaded with information from the session and decided further retrospection would not benefit Casey and stopped the session at this point. However, Casey missed resolving the multivalued attribute *qualification* in both *Trainee* and *Instructor* entities.

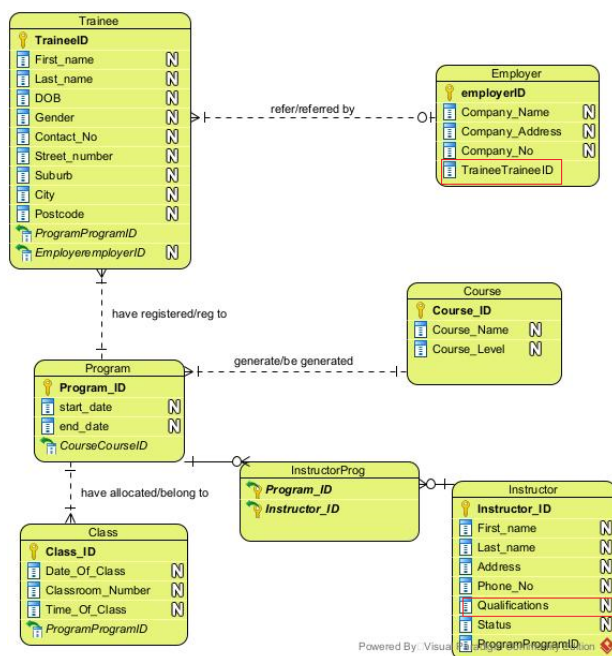


Figure 8.46 Casey's completed logical model. The red boxes highlight the remaining errors in their model.

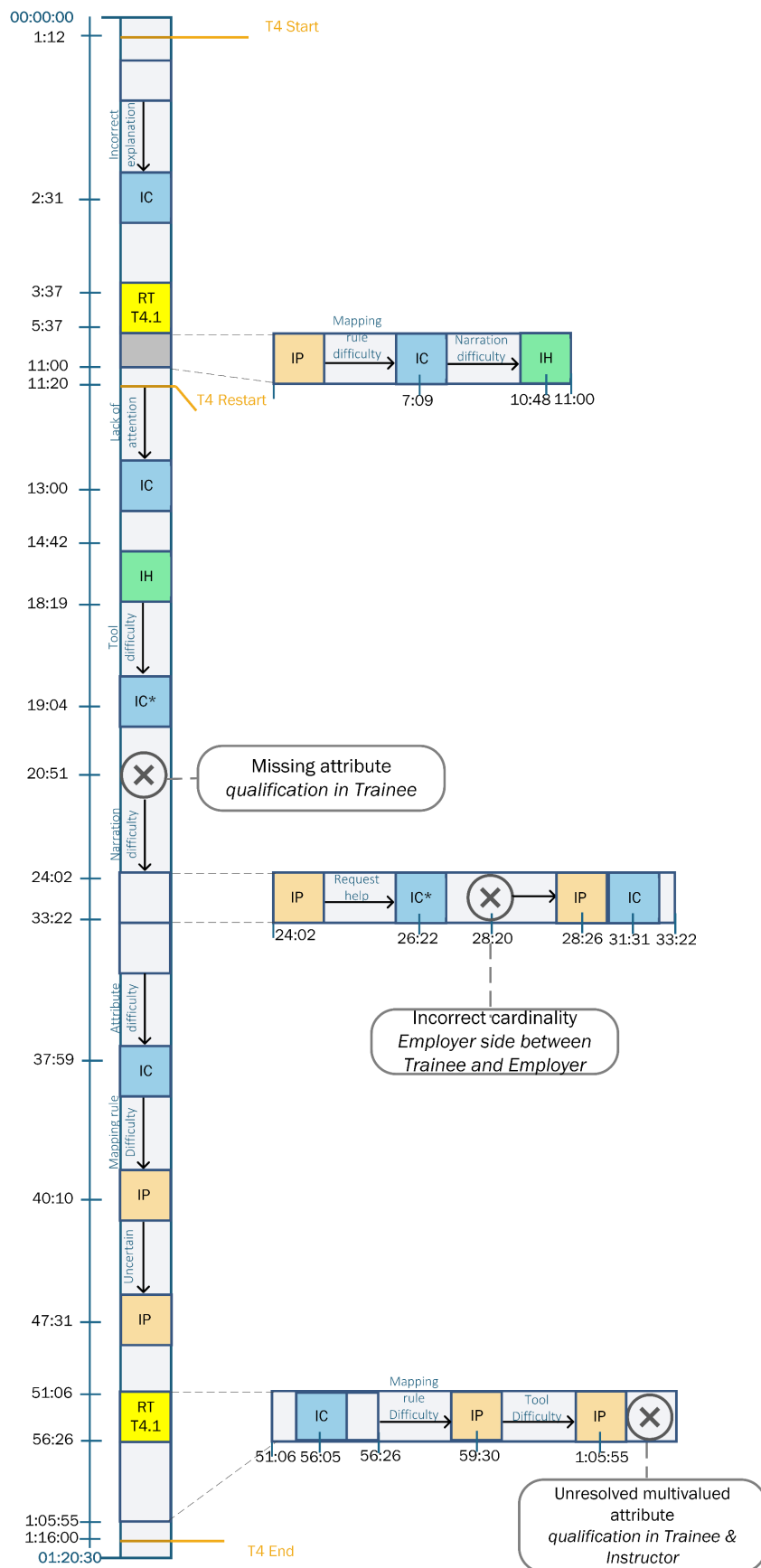


Figure 8.47 Casey's LT/P at the end of Task 4.

8.7 Discussion of LT/P's Tasks 1-4

Developing the learning trajectories provided an opportunity to identify how the participants progressed throughout their sessions, giving insight into their individual learning abilities and their difficulties. Even though the narratives are used for exploring each individual in-depth, they can also be used as a mechanism to compare learners who are operating at different levels to understand their thinking processes, practices and tactics and possibly which teaching interventions might be helpful in different scenarios.

Morgan, the best-performing participant, managed to progress their learning and gain an improved understanding of the concepts explored with the interviewer's assistance throughout all three sessions. Morgan managed to complete Tasks 1 to 3 by applying their own processes and tactics, which it seemed they had developed over time since they had some, albeit limited, exposure to databases at secondary school. Despite having more difficulties in completing Task 4, the task involving transformation from a conceptual to a logical model, Morgan was again able to complete the task with assistance from the interviewer. During Task 3, Morgan was provided with more interventions than Blake. However, out of the six interventions, five were related to using the tool rather than their comprehension of the knowledge needed to solve the problem. This was the major difference in the learning trajectories of Blake and Morgan.

In analysing the case study description, Morgan was the only participant who used a consistent annotation system to identify and extract the relevant data. This meant they finished Tasks 1 to 2 in less time than the other participants. Morgan also had a knowledge of the proper presentation of the relationship information using crow's foot notation. Despite a few errors (see Table 8.2), Morgan was rather confident and calm while completing the tasks in session one. Casey faced challenges from the outset and struggled to understand the case study narrative. This, along with an inconsistent annotation scheme, led to constant difficulties throughout. These struggles Casey had from the outset further explain why it was difficult for them to interpret the case study and understand the problem presentation, which were themes identified in the prior thematic analysis. Unlike Morgan or Casey, Blake did not follow a systematic annotation scheme for identifying and extracting the relevant data from the case study during Task 1. This meant Blake made more mistakes than Morgan (see Table 8.2).

In the online data collection sessions, Blake used their tablet with a pen, a closer resemblance to the original in-person sessions with pen and paper, while Morgan used the Visual Paradigm modelling tool. It seemed that using the tool put an additional cognitive strain on Morgan when completing the task, and indeed, as mentioned, the main challenges he encountered arose due to the tool and unfamiliarity with its use and default settings for model elements like cardinalities.

The main source of Morgans' notable difficulties when attempting Task 4 was a lack of a systematic approach for performing such a transformation. Both Morgan and Black could only recall one of the taught mapping rules for transformation – the rule related to the resolution of many-to-many relationships. Even though they both recalled that there were other mapping rules, they couldn't recall, explain or utilise them correctly during the task. This indicates that more time, examples and practice are required in order to reinforce learning. Indeed, in the case of Morgan, simple hints and prompts were sufficient to trigger recall of either related teaching in class or examples in class. From that recall of similar examples, Morgan was able to apply that prior knowledge to the new context of this research's case study. Moreover, the interviewer did not need to ask Morgan to complete a redirected task at any stage. This suggests that one main reason for Morgan's lack of a systematic approach, which was identified during the thematic analysis, directly relates to their lack of practice in similar tasks rather than having a fragile knowledge of modelling concepts.

Morgan was more used to the concrete implementation of a database (perhaps because of their prior experience), which they used to understand the models, and interestingly, none of the others mentioned the implementation aspect during their sessions. However, sometimes the prior knowledge can cause hindrance in their learning, which was evident from Morgan's narration. In fact, Morgan performed better due to their persistent nature and also due to practising on more questions outside of their interviews during the lectures and labs.

Like Morgan, throughout the sessions, Blake demonstrated the ability to learn from interviewer assistance. As a result of interventions, Blake was able to recall their in-class learning and make associations between what they learnt and the tasks attempted to achieve a higher level of understanding. This was evident in how Blake completed each task with less assistance as they progressed through the modelling problem. Comparatively, Blake needed more assistance than Morgan. In Blake's case, redirecting them to simpler tasks with a focus on one or the least number of concepts helped fill the gaps in their knowledge and enabled them to complete the task at hand without any further assistance. Despite encountering some barriers to progression and needing more directed teaching interventions than Morgan, Blake was able to fix errors and move forward on the tasks. Blake was observed continuously verifying/checking their solution before moving on to the next task. Consequently, unlike Morgan and Casey, Blake had no persistent errors or errors transmitted from one task to another. This further provides insights that a proper verification of the decisions made during a task leads to the correct knowledge transfer between tasks, which explains a primary reason for the difficulty of the lack of transfer discussed during the thematic analysis. Unlike Morgan and Casey, Blake showed evidence of always attempting to follow the processes taught in class. However, Blake had very fragile knowledge of foreign keys, where they couldn't recall the rules on this and therefore couldn't handle them during Task 3. Despite their struggles in Task 3, during Task 4, Blake was more

in control and managed to complete the task, only requiring clarifications or general prompts from the interviewer.

The best-performing participant, Morgan, performed well and only had a few errors in their solutions from Tasks 1 to 3. However, they found Task 4 relatively harder and made more mistakes than the other three tasks due to their fragile knowledge of the concepts of mapping rules required to complete the task. Contrastingly, Blake found Tasks 1 and 2 harder and made more mistakes, particularly in Task 1, mainly due to not following proper tactics in interpreting the case study. Further, it seemed they built their knowledge during the first two tasks while familiarising themselves with the problem domain, making these tasks more difficult than Tasks 3 and 4. By the time they started the last two tasks, they managed to transfer their knowledge more readily from the previous tasks, making them complete Tasks 3 and 4 with fewer errors. Casey, however, struggled throughout the database model development problem despite the nature of the tasks.

As mentioned, Casey, the lowest-performing participant, struggled more during all the sessions and repeatedly required the most assistance from the interviewer. Casey appeared to be out of their comfort zone most of the time and spent more time completing the tasks. Encouragingly, the interviewer's assistance to begin with seemed to get them to progress, and Casey often indicated that as a result of a prompt or redirected task, they understood. Casey was able to fix an error or move forward there and then. However, it was doubtful whether the knowledge was retained from one task to the next and one session to the next. When referring back to their own work, Casey struggled to recall/understand what they had learnt and what they had done to solve the previous task in the sequence. The lack of verification of design decisions was discussed earlier as a reason for the lack of or incorrect knowledge transfer between tasks. But in Casey's situation, they could not read and understand their solutions from one task to the next, which could be another reason for the struggles related to the problem recall theme identified during the thematic analysis. As Casey progressed through the tasks, they relied more and more on interviewer instruction and encouragement, and the interviewer interventions became less effective.

During work on Task 3, the interviewer had to intervene multiple times. Most instances were not linked to the use of the Visual Paradigm tool, though. Casey was overwhelmed by the complexity of the tasks throughout, and this was most obvious during Task 4. Casey was unsure of what to do and how to start. The interviewer had to redirect Casey to a simple task designed to help them recall the mapping rules, as they could not undertake the task without this knowledge. In this case, the redirected task did not help Casey, unlike others, since they struggled to recall what they had learnt, and also the different terminologies used in the process also appeared to hinder their memory. It was evident that Casey could not continue the task since they lacked the knowledge of the basic concepts and didn't have the correct

knowledge structures to relate to the mapping rules as it required the knowledge of a combination of interrelated concepts. Despite providing many different types of low-level assistance, Casey didn't seem to grasp the concepts correctly. However, the session continued since Casey wanted to complete the task despite the difficulties, which was commendable. In an odd case, Casey referred to the case study during the transformation instead of the conceptual model they developed. It seemed Casey had a hard time reading their own model, which added additional struggles which they faced during the previous tasks.

Overall, three different levels of participants who represent anecdotally the spread of abilities seen in the course were analysed through their narratives. Despite their varying performance levels, some common threads were seen across all three participants. Despite their performance level, all the participants struggled to understand the terminologies used when expressing their ideas correctly. Further, at certain times, all of them struggled with their incomplete or fragile knowledge of certain concepts. Both the best-performing and middle-level participants demonstrated a consistent approach to developing solutions and made progress with the interviewer's assistance. However, these interventions (i.e., high-level interventions such as clarifications and general prompts) that proved helpful for particularly the high and middle-level participants did not yield the same results for Casey, the lowest-performing participant. The study identified various types of prompts that were effective in aiding participants, providing valuable insights for educators in designing learning exercises and personalising instruction. These findings can also inform the development of automated personalised learning systems. For the worst-performing participant, it became evident that they lacked the foundational knowledge necessary to tackle the tasks effectively. Their struggle with grasping the fundamental concepts and terminologies made it difficult for them to make progress, even with the interviewer's assistance. In such cases, it becomes crucial to recognise the importance of establishing a solid understanding of basic concepts before attempting more complex tasks. Addressing these knowledge gaps and providing targeted support to build foundational understanding and reinforce the fundamental concepts can be essential in helping the worst-performing participant and others facing similar challenges. An overview of each participant based on their performance (i.e., number of interventions required, time on tasks, errors they made and probable reasons) is provided in Table 8.2.

Table 8.2 Summary of interviewer interventions, time on task and errors made by the participants across the sessions.

Participant	Session	Task	Number of interventions during the session		Time on task	Error	Possible reason for the error
			High-level	Low-level			
Morgan	1	Task 1	-	-	24 minutes	Missing attribute – qualifications in Trainee entity	Lack of attention to detail and skipped reading the line which included this information.
						Incorrect cardinality – Program to Course	Incorrect assumption based on real-world experience while the required information was available.
		Task 2	-	-	10 minutes	Incorrect cardinality in one business rule – Program and Course on the course side	Transferred from error in Task 1.
	2	Task 3	4	2	22 minutes	Incorrect cardinality – Program and Course on the Course side	Transferred from session 1.
	3	Task 4	6	1	27 minutes	Unresolved composite attribute – name and address in Trainee entity	Incomplete knowledge of mapping rules.
						Incorrect primary key – in Qualification entity	Incomplete knowledge resolving a multivalued attribute.
						Incorrect foreign keys – instructorID in Trainee entity and traineeID in Instructor entity	Incorrect use of the Visual Paradigm tool.
Blake	1	Task 1	1	-	18 minutes	Incorrect entity – CareerAcademy	Not having a proper mechanism to interpret the case study
						Redundant entity – Trainee and Employee	Not having a proper mechanism to interpret the case study.
						Incorrect cardinality – Employer to Trainee and Instructor to Program	Lack of attention to detail (skipped reading the related line) hence incorrect interpretation.

		Task 2	3	2	16 minutes	Incorrect use of participation (may/must) – in one business rule	Lack of attention to detail.
2		Task 3	2	2	22 minutes	Incorrect foreign key – in the course entity	Fragile knowledge
						Incorrect syntax – when indicating the strong/weak relationships (Not a major issue in the conceptual model)	
3		Task 4	3	-	20 minutes	Unresolved composite attribute – name and address in Trainee entity	Incomplete knowledge or rather misconception about transforming a conceptual model into a logical one.
						Unresolved multivalued attributes – qualifications in Instructor and Trainee entities	Misconception about transforming a conceptual model into a logical one.
							Struggle with the terminology for not being able to identify the type of the attribute.
Casey	1	Task 1	-	-	27 minutes	Redundant entity – Trainee and Participant	Not following the taught process consistently.
						Incorrect potential entity – session	Literal word-by-word translation.
						Unclear representations of the relationships	Not considering the relationships in both directions.
		Task 2	8	2	1 hour 17 minutes	Reviewed after section 1	
						Incorrect business rule – between Trainee and Course	Literal word-by-word translation.
						Incorrect cardinalities – between Trainee and Employer	Difficulty in referring to the previous solution and struggles in understanding the narration.
						Incorrect verb phrases – <i>may not be referred by</i> between Trainee and Employer	Incomplete knowledge of writing business rules.

2	Task 3	10	1	49 minutes	Reviewed after section 2 and 3	
					Incorrect verb phrases – <i>be generated if they are run</i> between the Program and Course	Incomplete knowledge of writing business rules.
					Incorrect cardinalities on Trainee, Program, and Class entities	Incorrect use of the tool and the inability to transfer business rules correctly to the model.
					Missing attribute – qualifications of the Trainee entity	Transferred from session 1.
					Incorrect foreign key – traineeID in Employer entity	Incorrect use of the tool.
3	Task 4	15	4	1 hour 20 minutes	Incorrect relationship – between Program and Instructor	Inability to transfer business rules correctly to the model.
					Missing attribute – qualifications of the Trainee entity	Transferred from session 2.
					Incorrect attribute – Employer entity	Not handling the tool correctly.
					Unresolved multivalued attribute – qualifications in Trainee and Instructor entities	Fragile knowledge related to mapping rules.

8.8 Narrative Analysis of the Normalisation Problem

The normalisation problem is described in detail in section 6.3.5. The normalisation problem was broken into three tasks as follows:

1. Identify the primary key/s for the provided data table and the current normal form of the data table. Briefly justify your answer.
2. Identify all the dependencies using a dependency diagram and decompose the relation into a set of 3NF relations. State clearly any assumptions you make.
3. From your 3NF relations, draw an Entity–Relationship (ER) model and include all attributes, primary keys, foreign keys, cardinality, and participation.

In the normalisation problem, the participants were provided with an existing data table that was susceptible to data anomalies. The data table consists of information about students taking different courses and the lecturer information delivering the courses. Their task was to identify the issues in the given data table and normalise the relations within the data table.

8.8.1 Morgan’s think-aloud session

Morgan completed the normalisation problem within a shorter period than the two other participants, taking approximately 15 minutes. As Morgan undertook the task, they followed their own approach rather than using the heuristics taught in the course.

Morgan first read Task 1 (see section 6.3.5) and went through the given data stating: “*OK, so we have information of students, courses and lecturers in this table.*”. Morgan immediately identified the primary key of the data table provided as a composite primary key consisting of the attributes *Student_ID* (*S_ID*) and *Course_ID* (*C_ID*). Morgan explained that the table was in 1NF since there were no repeating groups (multivalued attributes) in the data. Morgan didn’t require the interviewer’s assistance and seemed to be in their Comfort Zone attempting this task.

Morgan immediately moved to the next task (Task 2), where they were required to identify the functional dependencies in the provided data and decompose them into a set of 3NF relations. They managed to identify the dependencies and derive a solution within a few minutes. Interestingly, as indicated in Figure 8.48, Morgan listed all the new relations, skipping over the step of identifying the type of dependencies. They correctly identified two of the four relations. They added *semester* as a redundant attribute into the relation determined by *C_ID* and *S_ID*, omitting the fact the *C_ID* can solely determine the *semester*.

Stud_ID -> LName, FName

Crse_ID -> Crse_Name, Lect_ID

Stud_ID, Crse_ID -> Grade, Semester

Lect_ID -> Lect_Lname, Lect_Fname

Figure 8.48 Morgan's solution to Task 2. The red box indicates the error in their solution.

Once Morgan confirmed that the task was finished, the interviewer decided to question why they placed the *semester* attribute where they did (see Figure 8.48). Morgan replied:

So you have your student ID and the course ID. And those two determine what grade that person gets. From that logically I'm thinking if you determine the grade from that, you also have to determine when that course was taken by that student. Because it doesn't, there's nothing that specifically details what semester would be defined by in the system. So I'm thinking the student and the course ID would determine that.

Based on Morgan's response, it was evident that they had speculated beyond the information provided for the task and made an assumption. This assumption was based on their existing domain knowledge instead of relying on the data provided, despite the information provided being adequate to make a decision in this scenario. This assumption led to Morgan introducing an error into their solution despite getting the relations mostly correct. However, Morgan could not explain how they derived their solution. Morgan read Task 2 again and wondered how to decompose the relation, not realising that they had already decomposed the given data into a set of relations. The interviewer decided to intervene at this point rather than wait until the end of the task to discuss the apparent gap in their knowledge of the normalisation process.

The interviewer asked Morgan whether the tables they had just identified were in 3NF in an attempt to trigger a recall of Morgan's knowledge of normal forms. Morgan struggled to explain what they had done and what needed to be done to complete the task, stating:

Again, I mean, I know that the problem is I know the definition but it's just like I've forgotten The actual meaning of what I'm saying. So it's like I remember the words, but I don't remember the context.

The interviewer decided to provide more low-level assistance and asked what functional dependency they were wondering about and whether it was “transitive”, providing a direct hint. This seemed to trigger Morgan’s memory:

Yes, so I remember to be in 3NF you need to be in 2NF. So I'm confident that this is at least the second normal form. But I'm not 100% sure if this is 3NF yet.

Despite decomposing the relation correctly, Morgan was unaware of the normalisation process or how they applied it (or not) to derive the solution. Morgan attempted to explain, recalling the term “transitive” but not the theory behind it:

So the problem I'm having is remembering. Because I'm looking at this [pointing to the dependencies they derived] and I just if there is a transitive dependency here, I've got a problem with that, I couldn't right now point it out to you.

Hence, the interviewer directed Morgan to a rather simpler task (T2.1 – see section 6.3.5), asking them to explain the steps to transform relations from one normal form to another. However, Morgan’s knowledge was fragile. While they recalled the names of the dependencies, they could not recall the correct process of using them in normalisation. At this point, the interviewer decided to explain each normal form briefly. Once this was explained, the interviewer directed Morgan to another task (T2.2). This task required the identification of the different types of dependencies illustrated in a simple dependency diagram. Morgan’s answers were partially correct. They correctly identified D1 (see Figure 6.15) as a full functional dependency but believed that D2 was a direct dependency instead of a partial dependency. Morgan was able to identify D3 as a transitive dependency, however the explanation they gave was unclear. The interviewer decided to explain how to identify transitive and partial dependencies in detail. Morgan informed that they understood the explanation stating:

Okay, I think I got it now. So transitive dependency is if you have an element or you have an attribute that is dependent on a non-primary key attribute. And then partial dependency is an attribute that relies on the primary key but not the whole primary key.

After this, the interviewer explained how to move from 1NF to 3NF step by step using the identified dependencies. During this discussion, Morgan explained they had determined the relations without following the steps taught in the class since they had some experience in developing basic databases during high school. But apparently, they never really understood the process of achieving a set of relations without any anomalies and mentioned:

I never really like I understood sort of the formula or sort of the way of going about identifying it and moving it out and putting it in its own place. And I sort of naturally learnt that, but I never really learnt the terminology or the exact concept.

The interviewer asked Morgan why they placed the *Lecturer_ID* (*L_ID*) as a dependent of the *C_ID* since this is typically derived when removing the transitive dependencies later in the normalisation process. Morgan responded by saying:

This is also my practical thinking coming, so I'm what I'm doing here is when I looked at the lecturer ID and I thought, well, there's lecturer ID, lecturer last name and first name. We don't want to store that information with the courses, so we'll do it with lecturer ID. So I automatically saw that. And then I saw, okay I have to somehow connect the course with the lecturer to know who is teaching each course. So that's why I used the primary key of the lecturer as a foreign key in the course to connect the tables.

In practice, this was correct, however, lack of knowledge and lack of a systematic process is likely to fall over when faced with more complex normalisation tasks even though it was a successful approach to a novice-level task like the one they had attempted here. It appeared that Morgan's limited experience in databases had, to some extent, enabled them to bridge the gap they had in their fragile grasp of the theoretical aspects of normalisation. After the discussion they had with the interviewer, Morgan mentioned the possible drawbacks of their approach by saying:

Yes, and that's very much a me thing where I sort of just ... I run and gun and I just go way too fast and I'll get the main stuff like the main content. But I always miss the little things that make up the big picture. So the little things that always make up the marks I always forget them, which is not a good way to do this.

Further, Morgan stated, "It's very it's much better to connect how I'm thinking about it to the actual theory behind it. It's now going to be a lot easier, I think, for me to do a question like this.". And then Morgan moved to the next task.

Task 3, described in section 6.3.5, required the participants to transform the decomposed relations identified in Task 2 into an ERD. Initially, Morgan added an entity called *Course_Lecturer*. Morgan realised this was incorrect when they referred back to their Task 2 solution. Morgan removed the Course Leader entity from their solution and continued adding all the entities correctly. Next, Morgan opted to add the relationships. In this instance, they decided to apply their real-world knowledge of the domain to make educated assumptions on the cardinalities of the relationships as required. Despite using the same tactic unsuccessfully in Task 2, in this instance, the tactic resulted in a correct solution. It is interesting to note that students, or at least Morgan, tend to apply assumptions but do not have the knowledge to judge when they are necessary or appropriate.

Throughout the normalisation problem, Morgan followed their own approach to work out how the relations should be developed to reduce the anomalies. While Morgan's approach was not the one taught in the content of this simple normalisation task, it seems to compensate for Morgan's lack of theoretical

knowledge on the subject. This meant they were able to bridge their gaps in knowledge with fewer interviewer prompts than the other participants. In the retrospective interview, Morgan reflected on the fact that it would have been easier for them to complete the problem if they had understood the concepts correctly. Morgan mentioned appreciating the interviewer's prompts and assistance. Morgan appeared to have managed to bridge the gap in their knowledge using the interviewer's interventions. Further interviews with Morgan, which were outside the time frame of the course and this study, would have been required to verify whether this knowledge was consolidated and if their experiences had indeed led to a more systematic approach to solving normalisation problems. Morgan's LT/P throughout the problem is given in Figure 8.49.

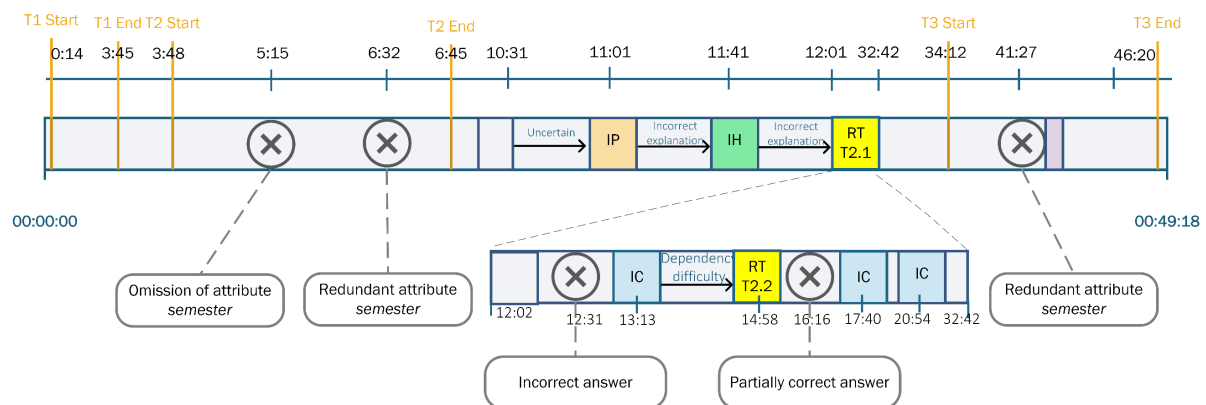


Figure 8.49 LT/P of Morgan during the normalisation problem.

8.8.2 Blake's think-aloud session

Unlike Morgan, Blake followed the process taught in class to complete the normalisation problem. Blake initially struggled with the normalisation problem but with the interviewer's assistance, Blake was able to complete the tasks within 40 minutes.

Blake read the preamble and started Task 1 (see section 6.3.5). Without any significant trouble, Blake mentioned that the relation was in 1NF. Further, they justified their decision as being because the table did not have any repeating groups. Blake seemed a bit unsure of the primary key and mentioned they find it hard to identify a primary key when there is more than one potential determinant as a candidate. Blake's knowledge seemed to be incomplete despite knowing the primary key was a composite one. Since Blake didn't proceed, the interviewer decided to assist them by prompting Blake to focus on a single row of data and then go through the table row by row to identify the determinants that might explain all data presented in the table. Blake decided to analyse the table data with this prompt and identified the primary key as a composite of S_ID and C_ID .

Next, Blake moved to Task 2, and after reading the task, they asked the interviewer about the dependency diagram. The interviewer provided further clarification on this, and it seemed Blake was more familiar with presenting the dependencies as an algebraic representation (e.g., $X \rightarrow Y$) instead of the diagram format mentioned here. The interviewer asked Blake to proceed with the version they were most comfortable with.

Blake's approach was to follow the process taught in class and determine the dependency types first. To do this, Blake, talking aloud, tried to recall the dependency types beforehand. Once convinced they had recalled what was required, Blake attempted the task by identifying the relations. Blake struggled with the terms "determinants" and "dependents" as they started explaining their answer. It appeared the terminology was difficult for Blake as the terms were quite new. The interviewer used a simple example (e.g., employer ID $\rightarrow$ employer name) to explain the difference between the determinant and the dependent in a dependency. With this quick clarification from the interviewer, Blake understood and used the terminology correctly.

Blake correctly identified the dependents of S_ID but seemed unsure about the dependencies identified by the S_ID and C_ID , as indicated in Figure 8.50. Blake's answer was partially correct in that both determinants (S_ID , C_ID) were required to identify the *grade*. But, S_ID was not needed to determine the L_ID . Instead, C_ID should determine the lecturer's details. Blake appeared to be going in circles, unable to progress and unable to identify their mistake. At this point, the interviewer intervened, directing their attention to S_ID and L_ID . The interviewer asked whether S_ID was required to determine L_ID . Blake realised that the L_ID should be removed from being dependent on the S_ID and C_ID . But then, they struggled to identify the determinant of the *semester* attribute, wondering:

Okay, the one remaining is the semester, so do I need both course ID and student ID? Or not...?

Stud-ID $\rightarrow$ LName, FName

Crse-ID $\rightarrow$ Crse-Name

Stud-ID, Crse-ID $\rightarrow$ Grade, lec-ID

Figure 8.50 Blake's incomplete answer for Task 2 (identification of dependencies).

At this stage, the interviewer intervened again, temporarily stopping work on the task and redirected Blake to a much simpler fill-in-the-blank task (T2.3, see section 6.3.5).

The objective of T2.3 was to check whether Blake could identify simple dependencies in the data table. Blake managed to complete this simpler task without any assistance from the interviewer. Blake returned back to the main task. At this point, Blake reexamined the data table more carefully. This time, they realised that the *semester* does not depend on *S_ID* and *C_ID* but still failed to realise what it depends on. The interviewer suggested that Blake try visualising the dependencies using a dependency diagram instead of the representation they had been using so far. Interestingly, Blake managed to progress and solve the task successfully using the dependency diagram but was unable to progress using the algebraic representation that they favoured (see Figure 8.51).

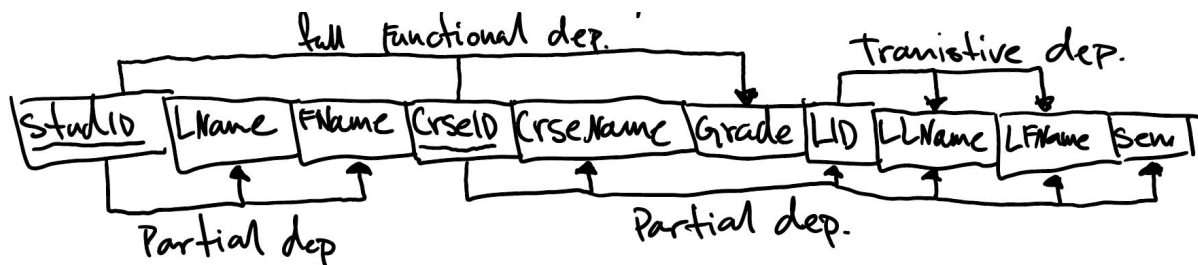


Figure 8.51 Blake's dependency diagram.

Once the dependency diagram was done, Blake explained how to decompose the relation. Blake had an overall idea of the process but didn't follow it in chronological order and stopped without doing anything for nearly 2 minutes. Hence, the interviewer prompted Blake to see if their memory could be triggered by asking them about the first step needed to move from 1NF to 3NF. This appeared to trigger their memory, and Blake explained that partial dependencies should be removed first, and they successfully decomposed the relation into 2NF without any assistance. However, they realised that there was no foreign key to connect the *Course* and *Lecturer* relations. At this point, the interviewer asked how they could move from 2NF to 3NF with the intention of prompting Blake to remove the transitive dependency, which would solve the foreign key problem. This prompted Blake to recall that the transitive dependency needed to be removed. However, Blake was unsure of how to proceed and requested help. The interviewer explained that the determinant of the transitive dependency becomes a primary key of the new relation and also serves as a foreign key in the old relation. With this explanation, Blake managed to decompose the relation into 3NF without any further assistance but omitted the arrowheads, which indicate the source of the foreign keys. Once the interviewer asked Blake to add the arrowheads, Blake was able to add them correctly, showing that Blake clearly understood how the relations were connected. Blake's solution is given in Figure 8.52.

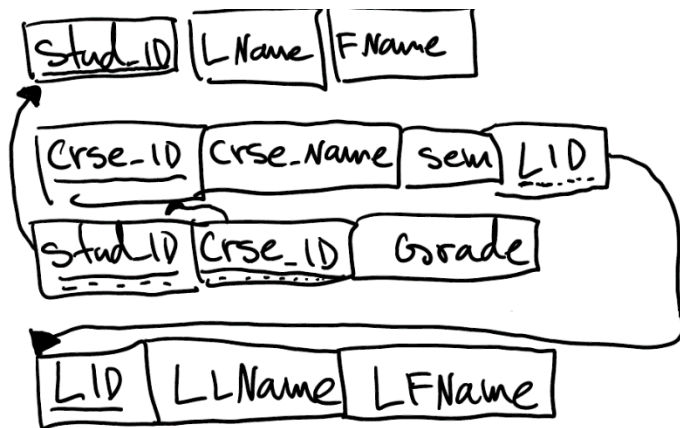


Figure 8.52 Blake's decomposed relations in 3NF.

Blake then moved to Task 3 and created the entities first based on the decomposed relations. However, Blake overlooked the need for the cardinalities but took the step to complete the task once prompted by the interviewer when she asked what was missing in the diagram. Blake correctly created the relationship between the *Course* and *Lecturer* but appeared unsure of the connection between the *Student*, *Student_Grade* and *Course* entities. The interviewer intervened and asked what type of entity *Student_Grade* could be. This prompt made Blake realise that it was an associative entity. Blake completed the task without any further assistance, and Figure 8.53 shows their final solution.

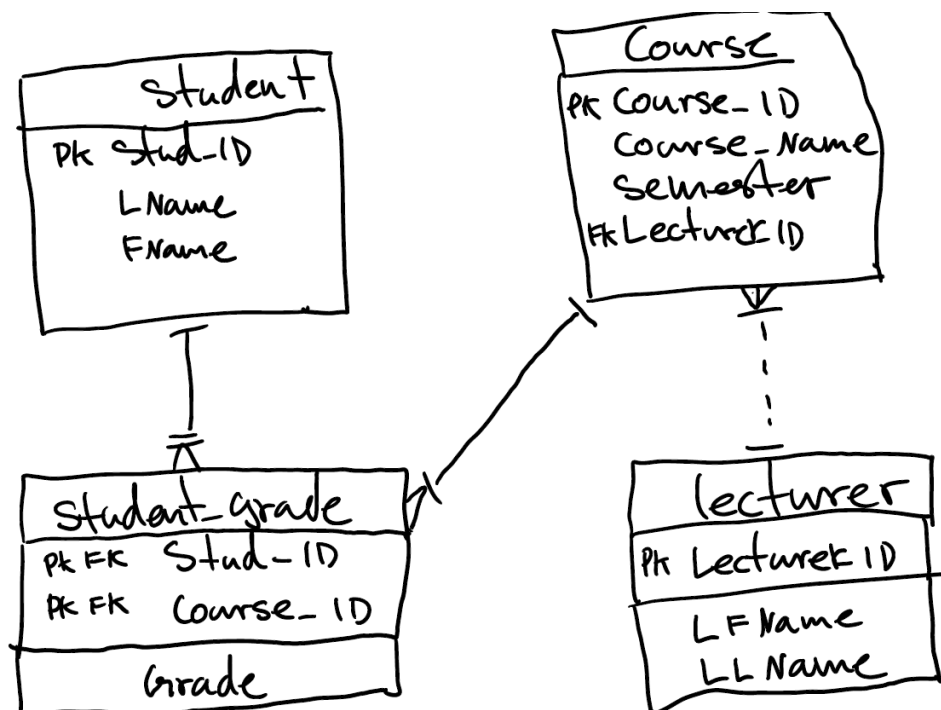


Figure 8.53 Blake's final solution.

Even though Morgan performed better by completing the normalisation problem faster despite using their own approach, which worked for this task, Blake had a better approach even though they did not follow it consistently. They mostly were able to recall the steps in the process but could not see the process as a whole some of the time. Hence, their misunderstandings and fragile knowledge surfaced more during the session. Despite the difficulties they had, the interviewer's interventions appeared to steer them in the right direction, which brought Blake into their ZPD. Therefore, as indicated in Figure 8.54, more interviewer interventions were required for them to proceed.

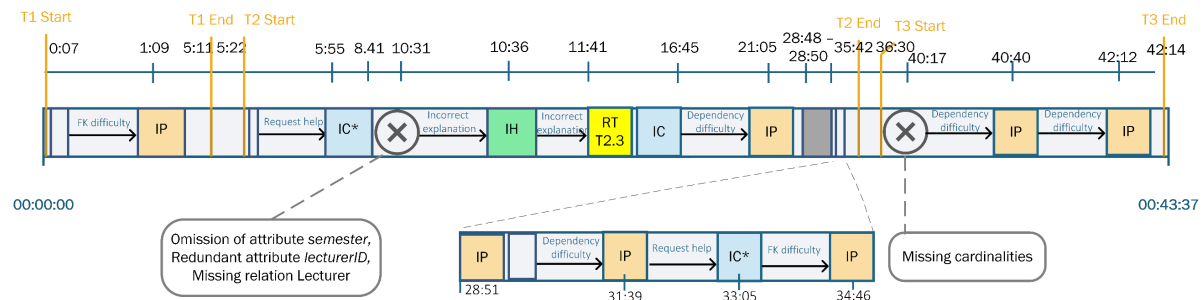


Figure 8.54 LT/P of Blake during the normalisation problem.

8.8.3 Casey's think-aloud session

Casey struggled the most in this last session, and the interviewer stopped the session before Casey completed the task. Despite their persistence and all the assistance from the interviewer, Casey failed to progress. Like the previous sessions, Casey spent more time on the task (approximately 1 hour and 30 minutes) than the other participants. Throughout the session, they demonstrated incomplete and fragile knowledge of the concepts related to normalisation. None of the interviewer interventions was particularly helpful.

Casey read the problem and immediately mentioned that the student data was in the 1NF but didn't provide any justification for how they reached that conclusion. They then moved on to identify the primary key for the relation and decided that the three determinants (*S_ID*, *C_ID*, *L_ID*) together were the primary key for the provided student data. Even though Casey seemed confident in their answer, it was apparent that they lacked the knowledge to derive this answer based on the vagueness of their verbalisation:

So the reason why I'm saying that the primary keys are student ID, course ID, and lecture ID is because the student ID, like ID as an identification, So without the person's name you don't have an ID. And without the name of the course and the grade of that particular course, you won't be able to link it back to the course ID.

Even Casey's written justification as an answer for Task 1 was vague and unclear (see Figure 8.55). Not realising their error, Casey moved on without pausing to sanity check/verify their decision.

Q2.a) Justify answer as to why Primary key are identified as below:

The Stud_ID will need a Full Name to identify the student.

The Crse_ID will need the attributes to define the Crse_ID by naming the course and grade.

The Lect_ID will need a person's Full Name to identify the Lect_ID which is the code to identify the person and what semester they will teach in, hence Sem1 or Sem2.

Figure 8.55 Casey's justification for selecting the primary key composed of S_ID, C_ID, L_ID.

The interviewer observed that Casey had trouble identifying the primary key from the very beginning of the normalisation problem. Casey was unaware that they had made an error.

Next, Casey moved to Task 2 to identify the dependencies and decompose them into a set of 3NF relations. Casey confidently started identifying the dependencies (see Figure 8.56). As Casey again didn't comment on the provided data while identifying the dependencies, the interviewer decided to intervene and asked them to explain how they recognised the primary key since it appeared that Casey was identifying the dependencies without any systematic approach. Again, Casey's answer was quite vague, and it appeared that Casey was confused with the terms used in normalisation and thus lacked the knowledge to provide a precise justification. Since Casey was struggling, the interviewer prompted Casey to focus on the dependencies by asking whether both *grade* and *course name* depend on the *C_ID*. Casey still seemed unsure and asked for help from the interviewer. At this point, the interviewer directed Casey to examine the data related to *grade* and *C_ID* in the table provided. Casey answered:

So well the grade A means that it stands for database whereas B means it stands for German. Because if you look at the rows you've got the same ID...

This showed that Casey did not know how to read the data in the table. The interviewer provided a more direct hint by asking Casey whether that means that everyone taking the *Database systems* course gets an A. Casey appeared to realise that there was an error in their identified primary key but still seemed unsure. With this, the interviewer asked Casey to find an example in the student data to explain why *S_ID* was a part of the primary key, but it appeared they didn't know how to explain this nor explain

the dependencies using the data. The interviewer realised that Casey lacked a comprehensive understanding of the concepts of functional dependencies and the normalisation process. Hence, the interviewer decided to redirect Casey to a much simpler task (T2.1) with the intention of breaking the problem down into manageable chunks to identify exactly where Casey's lack of understanding began.

STUD_ID ->	LName	Fname	
Crse_ID ->	Crse_name	Grade	
Lect_ID ->	Lect_Lname	Lect_Fname	Semester

Figure 8.56 Casey's partial solution for Task 2.

In T2.1, they were asked to explain the process of moving from UNF to 3NF through 1NF and 2NF with the intention of making them recall the required functional dependencies needed to go through the process. Casey could not recall any of the steps, so the interviewer asked them to attempt a different task (T2.2, see section 6.3.5) where they were asked to identify the different types of dependencies illustrated in a simple dependency diagram, hoping that a visualised form might help Casey to recall what they have learnt in the class. The main objective of this exercise was to provide a bridging step that scaffolds the dependency types in a dependency diagram and tests the participant's ability to recall and identify dependency types. The attributes in the diagram were presented in a generalised form (e.g., C1, C2, C3, etc.) to remove the requirement for having real-world knowledge to understand the dependencies. However, Casey appeared to be even more confused with this redirected task as they explained:

Well D1 block.. it's dividing the two groups which are D2 and D3. We've got C4 and C5 in that's in the D3 group and then we got C1, C2, C3 in the group D2. That's what I can see.

As Casey appeared not to be progressing, the interviewer stepped in to explain how to interpret and read dependency diagrams. Interestingly, Casey was adamant that they had not learnt that type of diagram provided in the redirected task and proceeded to point out a different diagram used in the lecture slides with the interviewer's permission. After a lengthy discussion, it appeared to the interviewer that Casey thought the diagrams were not the same because of how attributes were named in T2.2. This was an interesting observation as it seemed that Casey could not generalise from the example presented in the lecture slides to a diagram representing different data in a different context. This included the simplified context-free example in T2.2. Perhaps the intended context-free version resulted in an abstract representation of the diagram. Either way, Casey saw both abstracted and concrete examples in a

different context represented as a dependency diagram as totally different things and was unable to see the commonalities in the representation. It was not surprising that Casey struggled to explain the dependencies when they couldn't identify them in a diagram. Perhaps Casey had not seen enough diagrams to generalise at this point, but both Blake and Morgan seemed to be able to understand the diagram.

Next, the interviewer decided to explain the functional dependencies in detail using the same diagram Casey pointed out in the lecture materials. Casey then attempted the task again, correctly identifying D1 (full functional dependency) but not D2 (partial dependency) or D3 (transitive dependency). It was evident that Casey struggled with the terminology and lacked the knowledge to complete the task. However, Casey repeatedly affirmed that they understood the interviewer's explanations and opted to continue when the interviewer suggested stopping the session. At this point, Casey appeared to be overwhelmed since they had spent almost 50 minutes on the normalisation task.

Considering the lack of progress, the interviewer provided Casey with the primary key to see if this would help Casey move forward with the task. Casey was then able to suggest two potential dependencies, as indicated in Figure 8.57, but appeared unsure about *grade* and where it would belong:

Now the only one I'm missing from here is grade because I don't know where I should put it Or maybe I should just delete it, ignore it.

Stud_ID -> Stud_LName, Stud_FName

Crse_ID -> Crse_name, Lect_ID, Lect_LN, Lec_FN, Semester

Figure 8.57 Casey's second attempt on Task 2.

Then, the interviewer provided another hint by asking if *grade* couldn't be placed under *S_ID* or *C_ID* where it should be placed. The intent was to make Casey realise that more than one key was needed to determine a *grade* for a particular student. Even with the hint, Casey seemed to think that *grade* was unnecessary. With no progress, the interviewer clarified that the *grade* should depend on both the *S_ID* and *C_ID*. Since Casey couldn't identify the transitive dependency, as indicated in Figure 8.57, the interviewer asked Casey to return and attempt T2.1 again. This time, Casey realised that the lecturer's details needed to be moved to a separate relation but was unsure about the dependency type and how to enact the move.

Then Casey explained that they needed to decompose the relations even though they had done it up to 2NF, albeit incorrectly. It was evident that Casey did not know or understand the process of

normalisation. Casey showed real persistence despite all the barriers they encountered and was determined to continue, even though the allocated time was up. The interviewer went along with Casey's desire to continue.

To decompose the relations, Casey started by writing create table SQL statements, as shown in Figure 8.58, going for a more concrete approach that focused on the database implementation. It was quite evident that Casey did not understand what the phrase "decompose the relations" meant. Finally, Casey said: *"So yeah I'm not sure. I give up."* approximately 1 hour and 30 minutes into the session. Casey did not complete the normalisation task.

Create Table Student	Create Table Course
Stud_ID (PK)	Crse_ID (PK)
Stud_Lname,	Crse_name,
Stud_FName;	Lec ID FK

Figure 8.58 Casey's attempt to decompose the relations.

From the beginning of the task, Casey was not systematic in attempting the normalisation problem. Despite their persistence, Casey lacked the required knowledge to complete the problem. Even with the interviewer's detailed and directed support of alternative examples and simpler redirected tasks, Casey was unable to progress. Along with the lack of knowledge, Casey appeared to have difficulty with the terms related to the normalisation process. In fact, they were not really ready to learn such a process due to their lack of comprehension of very early fundamental concepts. As a consequence, they were out of their Comfort Zone and their ZPD and unable to learn. From Casey's narrative, it was evident that without this early understanding of the concepts, students cannot begin to model or solve simple novice-level database modelling, let alone work on more complex case studies such as those encountered in take-home assignments and develop databases from scratch. The LT/P of Casey with the interviewer's interventions is provided in Figure 8.59.

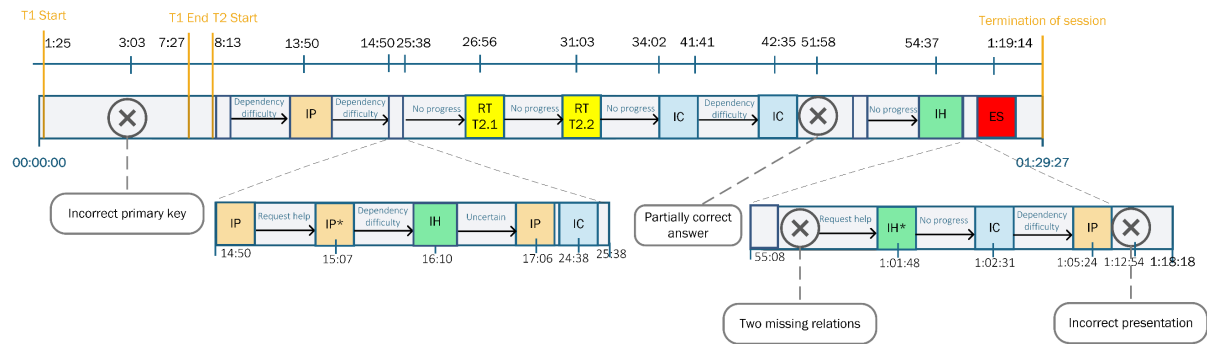


Figure 8.59 LT/P of Casey during the normalisation problem

8.9 Summary

This chapter focused on the analysis of think-aloud data collected while attempting database modelling tasks of three participants (Morgan, Blake, and Casey). The participants attempted two novice database modelling problems, one using a bottom-up approach and another using a top-down approach. A narrative analysis was used to analyse their data to understand each participant's learning progression.

The limitation of using only one problem broken into tasks in this research is duly acknowledged. As a result, drawing broad conclusions or making definitive statements based solely on this limited scope can be challenging. The observations made in the past literature might not be fully applicable or generalisable to this specific study. However, it is essential to recognise that this research serves as a valuable starting point and provides interesting insights for future investigations that can be explored using multiple problems and tasks.

The next chapter discusses the findings of the verbal protocol analysis in light of the existing learning and cognitive theories to examine the factors that influence the learning of novice modellers to comprehend how they learn to model.

Chapter 9 Linking Findings to Theory

9.1 Introduction

This chapter explores different learning and cognitive theories and examines to what extent these theories explain the findings observed during the qualitative analyses detailed in Chapter 7 and Chapter 8. Learning and cognitive theories commonly explain how learners acquire knowledge and skills, but they come from two different perspectives. Learning theories emphasise external factors in shaping learning, whereas cognitive theories focus on the internal mental process involved in learning.

The primary aim of this chapter is to map the observations extracted from the participants to the existing learning and cognitive theories to establish whether they can reasonably explain how novices learn to model and the source of novice difficulties when it comes to database modelling.

The chapter starts by relating the findings to different theories in the domain of educational psychology. Theories of knowledge construction and transfer, memory capacity and design thinking are explored.

9.2 Theories of Knowledge Construction

In this section, three theories concerning knowledge construction were explored that offer potential insights into the interpretation of findings derived from thematic and narrative analysis. The first theory examined is Piaget's, which is a theory on which other educational theories on learning are founded, and subsequently, Vygotsky's ZPD and Sfard's theory are considered both of which build on Piaget to some extent. All three theories acknowledge the learner's active role in knowledge construction and recognise the importance of interaction with the environment, even though the nature of interaction differs from one theory to the other. Unlike Piaget, Sfard and Vygotsky both consider social interactions, discourse, and communication as central factors in facilitating learning. The foundations of these theories are presented in the next section, and a discussion of their relevance, or not, to the knowledge construction observed in the think aloud sessions, is discussed.

9.2.1 Piaget theory

The aspect of Piaget's work that relates to knowledge construction and learning is the notion of schema development (assimilation, accommodation, equilibrium, and disequilibrium) and is the theory that is explored in this section. Piaget's work focused on studying a small set of children, including his own children, not adult learners. This is one of the reasons why his theory has often been criticised for its lack of generalisability and applicability in adult learning. Because Piaget's theory was originally formulated to explain the cognitive development of children, some argue that its principles cannot be

extended to adult learning (Matusov & Hayes, 2000). During the 1960s, psychologists raised questions regarding Piaget's position that the highest level of cognitive development is achieved in early adolescence (Long et al., 1979). While they were not criticising the applicability of Piaget's work in children's cognitive development, they were questioning using Piaget's theory to understand adult cognitive development. Since then, many researchers have used Piaget's theory in education and CSER with adult learners. In CSER, the main work tries to understand novice programmers' knowledge construction and cognitive development stages (Culligan & Casey, 2020; Kasto, 2016; Teague, 2015).

Piaget defined schema as "a cohesive, repeatable action sequence possessing component actions that are tightly interconnected and governed by a core meaning" (Piaget, 1952, p.7). Or in simple terms, a schema or schemata (plural of schema), as defined by Wadsworth (2004), is a set of index cards in the brain where each one indicates to an individual how to proceed or react to new information. Piaget claims that a learner is in a state of disequilibrium when they come across new information that is too far away from their existing schema to understand and make sense of it. When individuals are in a disequilibrium state, they actively construct meaning and add new information to the existing schema.

When a learner encounters such a new situation or new information, their schema is adapted. Piaget describes two different forms of schema adaptation: assimilation and accommodation. Assimilation occurs when new information is added to an existing schema. For example, if a schema has been formed of a dog that encompasses the physical appearance of a dog and then a dog is seen and it barks, the sound of barking is added to the existing schema for a dog. In contrast, accommodation involves a new piece of information being learnt that restructures a schema and can result in a new additional schema being formed. For example, given a schema of a dog, if you see a bear but the bear is silent, then you can assume the bear is a dog since, after all, it has eyes, ears, nose, four legs and fur. But the bear growls, not barks, hence, it does not fit into the schema for a dog. In this case, the new information results in a new schema being formed for a bear, which is different to a dog. In the case of accommodation, the existing schema no longer works when new information is encountered. This new information causes disequilibria, which trigger the formation of the new schema and, hence, learning. Piaget suggests that in such situations, a teacher or parent can provide feedback and reinforcement, which supports the development of a new schema.

The idea of Cognitive Dissonance arises from disequilibrium (Piaget, 1929). From the educational perspective, Piaget viewed cognitive dissonance as a mechanism that facilitates the cognitive processes of accommodation and assimilation and is integral to knowledge development. According to Piaget, cognitive dissonance is an essential trigger in learning that prompts learners to resolve the discrepancy between existing schema and new information.

There is no way to see a learner assimilating and accommodating their schemata directly. However, researchers can observe learners and ask learners to articulate their thoughts and interpret these observations. This was the approach taken in this doctoral work, and the discussion of observations that may be attributed to cognitive dissonance, assimilation, and accommodation are explored here.

The evidence from the lowest-performing participant, Casey, suggests that they frequently experienced a state of disequilibrium when encountering new information during the modelling tasks. Casey showed little evidence of being able to adapt their existing schema. For Casey, it could be argued that the tasks involved too much contradictory information, resulting in unmanageable levels of cognitive dissonance. Casey struggled to form a schema for both the domain of the modelling problem and the database concepts needed to solve the tasks. They rarely reached a level of disequilibrium that facilitated their learning and instead were often overwhelmed, which led to appealing for help or giving up. For example, in transforming the conceptual model into a logical one, Casey even struggled to understand what the task required them to do, which questions whether they had a schema. They appeared not even to have a faulty schema. Based on Casey's discussion with the interviewer, it was evident that the main reason for their lack of a well-formed schema was their fragile knowledge of concepts and terms, a theme identified in the thematic analysis (see section 7.8.5). When the interviewer explained how to resolve one-to-many relationships by modelling the relationship and adding the foreign keys correctly (i.e., adding the primary key of one side entity as a foreign key to the entity on the many side of the relation) between *Trainee* and *Employer* entities Casey managed to recall that they had done this during the lecture. This indicates that they may have had a schema to handle one-to-many relationships. In this situation, the interviewer's explanation could have allowed them to assimilate the new information and fix their faulty schema. However, Casey again incorrectly resolved the one-to-many relationship between *Program* and *Course* entities and had trouble handling this new information in their existing schema related to one-to-many relationships. This was a clear indication that Casey struggled to adapt their existing schema to encompass the new information introduced by the case study.

Throughout the think-aloud sessions, Casey struggled to assimilate and accommodate new information into their schema and spent more time completing the tasks than the other participants. Time on task and inability to progress despite significant effort suggests that Casey experienced prolonged periods of unmanageable cognitive disequilibrium in which meaningful learning and its associated schema development did not occur. This led to frustration and a tendency to rely on the interviewer for help.

Both the top-performing and middle-level participants demonstrated instances where they constructed their solutions through assimilation. They effectively utilised existing schemas as references and incorporated new information into their pre-existing schema structures. For example, when adding the relationship between the *Program* and *Class* entities in developing the conceptual model, Blake

incorrectly added the foreign key but indicated their uncertainty about it, leading them to a state of disequilibrium. However, with a prompt, they immediately managed to recall what they learnt during the lecture and realised their mistake. This could be considered an example where Blake managed to fit the new information into their existing schema through assimilation, consolidating their conception of foreign keys. During the normalisation problem, when they were asked to decompose the relation, Blake started the task by recalling the different dependency types required for the decomposing, which was a clear indication that they had an existing schema required to solve the task. Blake opted to use the representation of using dependencies and determinants (i.e., $X \rightarrow Y$) to indicate the dependency types, but after only identifying two dependencies out of four, Blake was in a state of disequilibrium and could not continue. However, later, Blake opted for a more visualised representation (i.e., dependency diagram), allowing them to complete the task. Even though Blake preferred the first representation, they could accommodate the new representation and restructure their existing schema.

Through the lens of Piaget's cognitive development theory, evidence of learning to model was observed when participants experienced a state of disequilibrium triggered by the tasks at hand. This disequilibrium serves as a trigger for successful schema adaptation through processes of assimilation and accommodation. However, for Casey, the disequilibrium often created a sense of frustration the majority of the time, where the disequilibrium state was not helpful. However, it is important to note that even though Piaget's theory can be used to explain the observations of how knowledge might be constructed, it does not tell educators how to reach desirable levels of disequilibrium suitable for learning or precisely what a desirable level is. Other theorists, such as Sfard and Vygotsky, have developed ideas such as scaffolding, assistance from knowledgeable others, etc., that educators can use to optimise learning experiences and bridge the gap between a student's existing schema and the required schema to complete a task, these ideas are discussed in the following sections 9.2.2 and 9.2.3.

9.2.2 Vygotsky's Zone of Proximal Development

The work of Vygotsky (Vygotsky, 1978a) is considered to be the foundation of sociocultural cognitive development theories (Gauvain & Richert, 2023). According to Vygotsky, learning and cognitive development occur through social interaction and collaboration (Vygotsky, 1978a). His theory emphasises the role of cultural and social factors in shaping an individual's cognitive growth. Vygotsky proposed that development occurs in two distinct ways: first through social interaction and then within the mind of the individual. Vygotsky argued, "*learning is a necessary and universal aspect of the process of developing culturally organized, specifically human psychological function*" (Vygotsky, 1978b, p. 90). This stance is in contrast to Piaget's belief that individual development is the core of learning rather than learning that comes from social interaction.

According to Vygotsky, adults play a crucial role in fostering a child's cognitive development. Through their interactions with children, adults provide guidance on how to think about and interpret objects, events, and experiences in their environment. They not only provide children with information (what to think) but also teach how to think, providing the processes and the cultural tools to think with. Both Vygotsky and Piaget define cognitive development in terms of developmental stages. However, Vygotsky emphasises the social aspect of cognitive development, highlighting the role of social interaction and guidance from more experienced individuals in the process.

Vygotsky's theory posits that students learn and develop cognitively when operating within their Zone of Proximal Development (ZPD) (see section 7.3). The ZPD refers to the range of tasks that a learner cannot yet accomplish independently but can achieve with the guidance and support of a more capable individual or More Knowledgeable Other (MKO). An MKO can be a teacher, a peer or someone with more knowledge and experience with respect to a particular task, process or concept. Vygotsky believed that a student would not be able to reach the same level of learning by working alone.

As a student leaves their zone of *current development* or *comfort zone* they travel through the ZPD towards their learning goal. Vygotsky categorised learning into three levels:

1. The *actual developmental level*: what a learner can achieve on their own.
2. The *potential developmental level*: what a learner can achieve or learn with MKO guidance.
3. A level that is beyond the reach of the learner, even with MKO guidance.

Over time, ZPD has become synonymous in the literature with the term *scaffolding*, however it is important to note that Vygotsky did not use this term in his work, and it was first introduced by Wood et al. (1976), who defined scaffolding as a "*process that enables a child or novice to solve a problem, carry out a task or achieve a goal which would be beyond his unassisted efforts. This scaffolding consists essentially of the adult "controlling" those elements of the task that are initially beyond the learner's capacity...*" (Wood et al., 1976, p.90). Many researchers have used the concept of scaffolding to explain the role of MKOs in guiding learning. The term *scaffolding* is borrowed from construction, where temporary structures are used to support workers as they build higher levels of a building. Scaffolding consists of activities provided by the educator, or MKO, to support the students as they are led through the ZPD. In the classroom, scaffolding can include modelling a skill, providing hints or cues, and adapting material or activity (Copple & Bredekamp, 2009). This support is gradually removed as it becomes unnecessary, much as a scaffold is removed from a building during construction until the student is able to complete the task or a similar task independently. Within the ZPD, learners engage in social interactions that provide scaffolding, which enables them to reach higher levels of cognitive functioning.

It is worth noting that a MKO does not necessarily have to be a human being. With the evolution of technology-supported learning environments, this assistance can also be provided through various forms of technology (McLoughlin, 1999). For example, educational software, online tutorials, or intelligent tutoring systems can act as virtual tutors, providing step-by-step instructions, demonstrations, and immediate feedback. These tools are designed to adapt to the learner's needs and provide support tailored to their current level of understanding.

As exemplified in the systematic literature review mapping study reported in this thesis, there has been a trend in recent years toward developing automated tools and tutors to accommodate personalised learning. However, understanding how students experience tasks and learning activities is crucial in developing useful personalised tools and tutors and to date, little work has been conducted examining the individual learning of novice database modellers.

Vygotsky's ZPD development and the notion of scaffolding were used for two primary purposes in this research: firstly, designing a set of tasks that were utilised in the think-alouds (see section 6.3) and secondly, to develop an intervention model (see section 7.4) to assist the participants during think-alouds to progress their learning. A range of interventions was provided, starting from high-level indirect/general prompts to trigger recall of knowledge to low-level modelling of problem-solving ending in a fully worked answer. If high-level interventions were unsuccessful, the researcher relied on lower-level, more directed assistance, such as hints on how to solve the task or redirection to a similar but less complex task. If none of the interventions worked, then the interviewer modelled the problem-solving process, explaining key concepts and processes as a last resort. This was intended for use in situations where a participant could not progress, and the task was beyond their reach. The model of interventions (specifically the redirected tasks) also played a role in helping understand where the difficulties lay that prevented the participants from progressing. Once the barrier was discovered then other specifically targeted tasks could be used to progress learning.

In order to explore the usefulness of Vygotsky's theory and scaffolding, the observations of the participants were categorised according to the three zones (Comfort Zone (CZ) – known as the actual developmental level, inside the ZPD – known as the potential developmental level, outside of ZPD – what they cannot do even with the help of a MKO) taking into consideration the types of interventions needed to complete a given task. In the ZPD, where participants have the potential to progress with appropriate support, the interventions provided by the interviewer serve as scaffolds to facilitate learning when the participants face barriers to progression. If participants were able to complete the task with the assistance provided, it indicates that they were operating within their ZPD and were able to utilise the scaffolds effectively to bridge the gap between their current abilities and the demands of the task. On the other hand, if participants were unable to complete the task successfully even with the

interventions provided, it suggests that the task was beyond their current capabilities and that the participants were outside of their ZPD. If the participants could complete the tasks independently without requiring any assistance, they were operating in their CZ.

Based on the observations from the participants of this study, it seems that they rarely operated in their CZ. Even the best-performing participant rarely operated fully within their CZ. Perhaps this is not surprising given that they were novices with limited prior experience in modelling databases. The participants who performed relatively well tended to operate in their ZPD. Both best and middle-level participants effectively utilised the MKO interventions to overcome any barriers they encountered during the tasks and to progress in their learning.

An observation regarding the best-performing participant, Morgan, raises an interesting question about their placement within the ZPD or their CZ. While Morgan often demonstrated a clear understanding of the process and had the necessary knowledge to complete the tasks, they still required prompts or high-level interventions to address overlooked aspects of the problem. In such cases, it appears that Morgan's need for interventions was not due to a lack of knowledge or an inability to perform the task within their ZPD. Instead, it suggests that their attention may have been focused on certain aspects while neglecting others. This could be attributed to factors such as selective attention (McCallum, 1996) or overconfidence in their abilities (Knorr, 2020). Determining whether Morgan was operating within their ZPD or CZ in these instances is difficult. It may be more appropriate to consider these situations as moments where Morgan's focus needed to be redirected rather than a lack of knowledge or skill. It is possible that they were in their CZ, but specific aspects of the task required external cues or reminders to guide their attention effectively. This observation highlights that even highly skilled learners can benefit from external support in regulating their own thinking process.

The lowest performing participant, Casey, was almost always outside of their ZPD and on rare occasions, they showed signs of being within it but were unable to consolidate the knowledge, which was evident from their inability to apply the gained knowledge in a similar task. Throughout the think-aloud sessions, Casey took more time to be able to do the tasks as they were struggling with the basic concepts required to do the tasks and learn from them. This is an issue for courses that have a lot of content and are intensive to deliver within a semester, which has been a constant issue discussed in the literature related to novice database courses where novices do not get ample time to consolidate their knowledge (Goldweber et al., 2022). Several participants from the thematic analysis also showed evidence of their struggles, which could have been the main reasons for not being able to reach their ZPD. These are discussed in section 7.8.5, related to the theme "fragile knowledge" on concepts, syntactic and terminology. Several researchers have discussed the struggles of novices when developing databases (see section 7.9.2 for a detailed discussion). One of the participants, Morgan, who had some

prior experience, expressed the same concern – *“Since I had some experience, it managed most of the time. But for most people, I would say the content was heavy. It's very easy to get lost early, I think. Honestly, if I didn't have that background, I'd be completely lost at this point.* Evidently, the course packed a lot to a shorter delivery time, not providing time to consolidate their learning. But this is universal in all first database courses.

The comparison between Blake and Morgan in terms of their need for assistance and their approach to problem-solving sheds light on the effectiveness of different intervention strategies. Blake often needed more direct assistance from the interviewer than Morgan. Blake benefited from the redirected task interventions, while Morgan benefited more from hints. While they both operated in their ZPD, the level of interventions required the majority of the time was different. But both of them could stay in their ZPD with the help of the interviewer, and there was evidence that those experiences consolidated that knowledge when applying them to a new or similar task. Overall, the participants in their ZPD did not need a modelled solution (worked solution) and were able to progress with minimal high-level interventions. The interventions needed focused more on consolidating the participants' existing knowledge and correcting minor misconceptions. In contrast, Casey was always outside of their ZPD, and in such situations, the interventions helped eventually, and they could solve that specific task, but when they moved on to similar but different tasks, they were again outside their ZPD.

These findings suggest which interventions could be helpful based on student performance. However, the question remains as to which types of interventions could help students like Casey. One obvious solution is to provide them with more time to learn and consolidate the basic knowledge required for database modelling. But this raises the question of how to teach them these basic concepts and terms. It is clear that they cannot even start a task until they grasp this basic knowledge. One possible way to help such students is to reduce the cognitive load associated with terms and concepts (Mason & Seton, 2020, 2021). It is possible giving Casey a glossary of terms may have provided a scaffold that allowed them to reach their ZPD. This is discussed more in-depth in section 9.3.

Vygotsky's theory and the notion of scaffolding proved to be a valuable lens to facilitate learner progress in database modelling. Particularly, the notion of MKO was helpful where it was evident that the help participants received from the interviewer helped them to progress throughout their think-aloud sessions. Morgan mentioned this: *“But having sort of feedback and having someone to bounce off of that, that's been very useful. I can throw my idea out there and get it criticised. That helped me realise what was wrong and from there, what I needed to do on the next task. In this way I can actually improve instead of just getting an idea of what I need to do”.* This suggests how important social interaction and learning are for novice database modellers—intentionally integrating pair tasks in novice database

courses since they need someone to bounce off their ideas or articulate their thoughts, which does not necessarily need to be the lecturer.

Moreover, the concept of ZPD provides a means of determining where a learner is situated in the context of a course using the notion of ZPD and measuring what the learner can achieve on their own and what they can achieve in collaboration with an MKO. In the context of this study, learners within their ZPD benefited from interventions like hints and redirected tasks provided by an MKO, designed with Vygotsky's theory as a framework with the intent of helping new knowledge to be adapted into the learners' cognitive schema. In such cases, the interventions help the learners navigate cognitive dissonance and achieve a state of equilibrium. However, it is important to recognise that learners who are outside of their ZPD face many challenges. They may lack the necessary foundational knowledge or understanding even to begin a task or make meaningful progress. In such cases, interventions and scaffolded tasks may not be sufficient to help them reach a level of independent learning. The question of how to support learners who have missed important foundational knowledge early on becomes more complex, as simply providing interventions may not be enough to bring them into their ZPD or enable them to progress effectively. While the concept of ZPD is valuable in understanding individual learners' needs and designing appropriate interventions, it is not pragmatic to provide personalised support in large classes. In order to effectively use automated tools for personalised learning, it is important to understand the types of interventions that work best for different students in depth. This warrants further research to identify the most effective strategies and approaches for various learner profiles.

9.2.3 Sfard's theory

Sfard (1991), in her work, developed a theoretical framework that focused on the development of an individual's conceptual understanding while learning mathematical concepts. She examined the dual nature of mathematical concepts, such as numbers or functions, and described them as having two fundamentally different aspects: structural (as *objects*) and operational (as *processes*). Sfard uses two metaphors, *acquisition* and *participation*, that inform understanding of the learning process (Sfard, 1998). Sfard (1998) defined *participation* as a sociocultural metaphor where learners engage in discourse and communication. Emphasised in the *participation* metaphor is the notion of interacting with more knowledgeable others, aligning with key ideas from Vygotsky's theory. *Acquisition* is a constructive cognition metaphor where individuals acquire knowledge through a process of accumulation. This metaphor conceptualises learning as the accumulation of concepts, which are seen as basic units of knowledge. The acquisition metaphor often assumes a static view of knowledge, suggesting that learners gradually build up their understanding by adding new concepts to their existing repertoire. According to Sfard, learners can acquire knowledge of mathematical concepts by transitioning from an operational understanding (*processes*) to an abstract understanding of

mathematical objects. Piaget too, has discussed how mathematical knowledge can be achieved via abstraction in two ways. One is from the objects the learners act upon, and the other is reflective abstraction which involves the action itself (Piaget, 1971). In later work Piaget (1985) hypothesised that reflective abstraction is the process by which every mathematical concept is constructed and is the foundation for all mathematical thinking.

Sfard believed that concept formation leads to abstraction in three stages: *interiorisation*, *condensation* and *reification*. Sfard (1991) defines *interiorisation* as a stage where learners start to familiarise themselves with the processes that eventually lead to understanding new concepts or modifying existing *objects*. Sfard explains that these processes are operations performed on lower-level mathematical objects. A classic example that Sfard provides is the process of counting, which leads to understanding the concept of natural numbers. Through the repeated process of counting objects, learners gradually internalise the concept of numbers and develop an understanding of the relationships and properties associated with natural numbers.

Sfard defines *condensation* as “a period of “squeezing” lengthy sequences of operations into more manageable units” (Sfard, 1991, p.19). At this stage, learners become capable of thinking of the process as a whole rather than as a series of steps. Sfard argues that due to condensation combining two processes, making comparisons or generalising becomes easier. Sfard also suggests that the condensation stage is where new concepts are realised, and learners start to integrate and combine different operations, recognising the relationships and connections between them. This integration of processes paves the way for the emergence of new concepts and the ability to apply them in various contexts. For example, in the case of negative numbers during the condensation stage, learners begin to combine the process of subtraction with other operations, such as addition or multiplication.

These first two stages happen gradually, whereas Sfard views the last stage, reification, as an ontological shift where “a process solidifies into an object, into a static structure” (Sfard, 1991). It is the stage where learners are able to identify and alternate between different representations of the concept, indicating a deeper understanding. Reification is a crucial stage because it marks the transformation of a process where it is fully grasped and turned into an object, finalising the process of abstraction. This object can now be used as a part of other larger processes or as a foundation for further development and learning. It represents a shift from purely operational thinking to a more structural understanding of a concept. For example, Sfard considers reification is achieved when a learner can treat negative numbers as a sub-set of the ring of integers (e.g., -2, -1, 0, 1, 2) without fully understanding the formal definition of a ring.

While Sfard's work primarily focuses on the development of mathematical concepts, researchers have applied her framework to understanding concept development in other domains, including CSER. The work of Kasto (2016) provides evidence supporting the three stages of *interiorisation*, *condensation*, and *reification* in novice programmers. This suggests that these phases could apply to novices learning database modelling since programming and database modelling require learners to grasp tightly integrated concepts that require understanding one concept fully before moving on to learning a new one. This notion is explored in this section.

When Blake added the relationship between the *Program* and *Course* entities when building the conceptual model, they realised: "*I'm not quite sure about the foreign key. There should be one, though*". They were familiar with the concept and appeared to be in an *interiorisation* stage however, they had to rely on the discourse they had with the interviewer to process it correctly for a one-to-many relationship between two entities. As Blake progressed on the task, they started to show evidence of *condensing* the foreign key concept as they realised which of the two entities should contain the foreign key. Blake continued being able to add correct relationships and foreign keys for one-to-many relationships. But, they were unable to identify foreign keys in the case of a many-to-many relationship. Despite attempting to *condense* the concept between foreign keys and the relationship type, Blake was still unable to see it as a single object to be applied in different situations. While struggling with foreign keys when developing the conceptual model, Blake handled the other concepts, such as attributes, cardinalities and primary keys, correctly. It is possible that although Blake was in a *reification* stage for some concepts, they were still at the *interiorisation* stage with others, making the overall modelling task difficult. Unlike Blake, Morgan immediately identified where to place the foreign key when handling the many-to-many relationship between the *Instructor* and *Program* entities by creating an associative entity and using the primary keys of *Program* and *Instructor* entities to create a composite primary for the newly added entity, indicating they were foreign keys. This indeed provides evidence for the *reification* of the concept of foreign keys. Morgan was able to treat the process of identifying and creating foreign keys as an object that could be used in the two different representations of the conceptual and logical model. The successful application of foreign keys in the logical model suggests that Morgan's conceptual understanding of foreign keys had progressed from the *interiorisation* and *condensation* stages to a more solidified *reification* stage. Casey appeared to be in the *interiorisation* stage throughout the task of developing the conceptual model. For example, when adding the foreign keys, Casey showed some familiarity with the concept but never moved beyond that stage: "*The logical, the conceptual it's really hard to distinguish you know like the meaning behind every term*". Casey failed to *condense* the foreign key concept, which suggests they were still in the early stages of concept development.

Based on Sfard's theory, the phases of *interiorisation* and *condensation* in concept development take time to occur fully. Observing and theorising about the concept development process of novice modellers during database modelling tasks can be challenging, especially within a limited timeframe of one semester and using a single task spread across multiple sessions. However, it is worth noting that despite the limitations, there were still valuable observations that provided insights into the concept development process of novice database modellers. All the participants, regardless of performance level for certain concepts, failed to condense and reify that concept.

While both Vygotsky and Sfard hold similar central ideas regarding knowledge construction through social interaction within the scope of this research, Vygotsky's Zone ZPD emerged as a more useful theory through which to view the findings. Sfard's theory did not really add any new insights, possibly due to the theories having similar theoretical underpinnings or possibly due to the fact the instrument was designed with the ZPD in mind, introducing bias. However, the former is the most likely reason.

Sfard's theory holds potential for future exploration of how concept development occurs among novices. But concept development, particularly the phases of *interiorisation* and *condensation*, takes time and *reification*, which involves a leap in understanding that would take an even longer time to establish (Sfard, 1991). Hence, a more longitudinal approach with an instrument with increasing levels of difficulty and an increasing number of concepts would be worth considering. Combining the temporal depth of a longitudinal study with the cognitive progression or acquisition would allow the researchers to gain insights into how novices build on their knowledge of concepts over time. Of course, this was not possible in the current research as undergraduate students at AUT have a single 12-week course on databases with only a few weeks dedicated to modelling.

9.3 Cognitive Load Theory

CLT is a learning theory that examines how an individual's working memory deals with information during learning and problem-solving tasks (Newell & Simon, 1972). The theory suggests that the cognitive load, or mental effort required to process information, affects learning and can influence learning outcomes. CLT views learning in terms of the processes related to storing and retaining information over an extended period in *long-term memory*. Duran et al. (2022) suggest that long-term memory has virtually unlimited capacity. Long-term memory serves as the repository of an individual's knowledge and past experiences. CLT emphasises the importance of leveraging long-term memory in the learning process so that learners can activate and integrate new information with their existing knowledge stored in long-term memory. It is argued that this integration facilitates meaningful learning and the construction of new mental *schemas*. To construct this knowledge in long-term memory, the information must first be processed in *short-term* or *working memory*, which is very limited in both

capacity (i.e., the number of *elements* a learner can process simultaneously) and duration. According to CLT, when the cognitive load exceeds this capacity, it can lead to cognitive overload and hinder learning. Therefore, effective instructional design should aim to manage the cognitive load placed on working memory to optimize learning outcomes.

There are currently two variants of CLT. The initial theory emerged in the 1980s from the work of Sweller and his colleagues (Sweller, 1988; Sweller et al., 1998). Kalyuga (2011) presents a revised version of CLT, referred to henceforth as *New CLT*. Sweller's theory will be referred to as *Old CLT* in this thesis.

In the initial 1980s work, Sweller emphasised the limitations of working memory and proposed guidelines for optimising the presentation of information to enhance learning. Later, Sweller (1998) revised CLT, introducing three types of cognitive load – *intrinsic*, *extraneous* and *germane* load:

- *Intrinsic load* is caused by the inherent complexity of the problem task, which is unavoidable when aiming for a particular learning objective. Typically, complex concepts or tasks with many interrelated elements have a higher intrinsic cognitive load which cannot be altered by instructional interventions.
- *Extraneous load* refers to the cognitive load imposed by the way the information is presented in the instructional design, which can be altered by instructional interventions. Poorly designed instructional materials, irrelevant information, or excessive visual or auditory stimuli can increase the extraneous cognitive load, impeding learning.
- *Germane load* refers to the mental effort required to process and integrate new information into existing knowledge schemata actively. It arises from a learner's engagement with a learning activity and is considered a "good" form of cognitive load to experience when learning.

According to the Old CLT, intrinsic cognitive load that occurs due to high element interactivity and extraneous load due to poor instructional designs are additive along with the germane load that creates the total cognitive load. They claim that when the intrinsic load is high, it is important to reduce the extraneous cognitive load to manage the overall cognitive load to ensure learning is relatively unhindered. When the learners have a manageable level of cognitive load, their germane load capacity is expanded, and they can be directed towards activities and processes that directly contribute to learning.

Kalyuga (2011) claimed that the germane load might be a redundant load since it was developed purely based on conjecture, unlike the other two loads, which had been justified using empirical results. Kalyuga said this might be why there were no empirical studies in the literature testing the concept of germane load. Thus, Kalyuga argued that germane load should be removed or reconceptualised and

redefined germane load as a cognitive resource associated with the working memory. This means the germane load is seen as a distinct dimension of the cognitive load allocated to handle the intrinsic cognitive load that leads to learning. This is not a new idea, as Sweller arrived at the same conclusion at around the same time (Sweller, 2010). This new idea changed the way CLT viewed the total cognitive load, which in Old CLT comprised a sum of all three types of load. Instead, in New CLT, the total cognitive load is the additive effect of intrinsic and extraneous loads only. Kalyuga (2011) claims that this redefined theory makes it more straightforward by eliminating the apparently redundant germane load and the need to develop separate distinctive measures for the three different types of load.

Considering both variants of CLT, it was decided to use Old CLT in this research as, even today, most articles in the computing education literature still refer to Sweller's Old CLT (Duran, 2022). Furthermore, Old CLT offers practical guidelines for reducing the cognitive load that are well-established, including worked examples and scaffolding to optimise learning. Old CLT effects that influenced the instrument design included segmentation (sub-goal effect), using redirected tasks (isolated-interacting element effect), using structured worksheet, and having sequenced sessions with the intention to reduce the intrinsic and extraneous cognitive loads (see section 6.3). All of these aspects were intended to free cognitive resources, which can then be allocated to coping with the germane load of the tasks involved, thus enabling learning.

During think-alouds, some participants showed evidence that the use of segmentation of the tasks, redirecting to simpler tasks, using a structured worksheet for the case study, and the sequence of presenting them in consequence sessions was a useful approach to reducing both extraneous and intrinsic loads and thereby freeing the working memory for the germane load where schema acquisition and construction happens. While this study did not have a control group to measure the effectiveness of segmentation of a problem, there was an instance where Morgan noted that the segmentation of the task helped them with the normalisation problem: *"Going back to it, the parts in the question helped I think..."*. There were other instances where presenting tasks in manageable chunks, in the form of redirected tasks, helped the participants. For example, when Blake was trying to develop business rules, they were redirected to a much simpler task that focused on the concept Blake could not grasp. The redirected task allowed Blake to focus on that particular aspect of the bigger problem, reducing the number of concepts to focus on at a time and the intrinsic cognitive load. In this case, the redirected task helped the interviewer observe that Blake appeared to be more focused after realising the source of their difficulty. This indicates that the redirected task reduced the cognitive load related to mental processing and that this freeing of working memory resources supported the construction of mental schemas.

Both Morgan and Blake mentioned that the way the case study was structured helped them organise their thoughts. Morgan acknowledged how it was easy to move between sections. When they had to check their understanding, they could go directly to the relevant section instead of having to navigate a lengthier case study where they could have missed certain aspects. This organisation of information helps learners organise their thoughts and systematically approach the task, ensuring they consider all the relevant aspects and details of each entity. This aligns with the principles of cognitive load theory, as learners can allocate their limited cognitive resources more efficiently by processing information in smaller, manageable chunks. This can enhance comprehension, analysis, and decision-making during database modelling tasks and is an approach that educators could use when they provide case studies for novices.

Despite integrating aspects of the research instrument to reduce cognitive load, some participants were observed to have difficulties that CLT could explain. The lowest-performing participant, Casey, struggled often and commented on being overwhelmed. This is most likely a result of excessive cognitive load. When learners experience a cognitive load that exceeds their cognitive capacity, their ability to process and understand information is compromised. Longer than normal time on task may also be an indicator of excessive cognitive demand being placed on a learner's mental capacity. Throughout Casey's data, there were instances where they were unable to hold and manipulate information in their working memory, leading to longer completion times. For example, Casey took more than an hour to complete each of the tasks, way beyond the scheduled time of one hour. Casey spent almost four times longer transforming the conceptual model into a logical one than Morgan and Blake did. It is believed that excessive cognitive load leads to frustration where a learner could stop a task in a shorter time, hindering their learning (Kalyuga, 2011b; Kasto, 2016). However, on the whole, Casey was persistent, continuing despite the fact they couldn't progress. This is likely to be due to the excessive cognitive demands for Casey on the tasks. It was evident that none of the introduced mechanisms reduced the cognitive load Casey faced, which in turn resulted in not having enough capacity in their working memory for the germane load. This is further evident when they repeatedly made the same mistake throughout the tasks due to their faulty schema. For a specific example, while developing the business rules, Casey spent almost 30 minutes writing one business rule. This task involved a simple translation from English to taught business rule grammar. The case study provided the information in a structured way and without additional irrelevant information. The unavailability of the necessary schema may have been the main hindrance for Casey in this case. They were unable to adapt to the new knowledge because of the intrinsic load. When given a redirected task to reduce the overall load (intrinsic and extraneous loads), Casey could not progress. This further supports the idea that Casey had limited capacity in their working memory, resulting in difficulties in holding and manipulating information required to develop business rules. Due to being unable to offload information

from working memory more frequently to free up mental resources, Casey could not gain the benefits of the redirected tasks that were developed to reduce the overall cognitive load. The intrinsic load of any database modelling task for a novice is likely to be high.

Even the better-performing participants struggled with using the correct terms, not only weaker-performing and ESL participants. In the fragile knowledge theme (section 7.8.5), this issue is highlighted with evidence from the think-alouds. When students struggle to use the correct terms, it can increase the intrinsic cognitive load of a task, making it harder for students to grasp the concepts and effectively apply them in their problem-solving. It requires additional cognitive effort to bridge the gap between the terms they are using and the correct terms associated with the concepts. These findings agree with those of Mason and Seton (2020). In their study, they provided a glossary of common terms in each student's first language in an examination to level the playing field and reduce the cognitive load imposed by these terms. They found when the glossary was provided, ESL students performed significantly better on final examinations. The findings of this doctoral research align with those of Mason and Seton, suggesting that most novices would benefit from such a glossary at the early stages of learning database modelling.

In deciding the problem description for the database model development problem, a familiar domain was adopted as a deliberate decision to reduce the cognitive load that an unfamiliar domain may bring. While Dhillon and Dasgupta (2011) argue that there is no difference between task performance and whether or not the problem is in a familiar or unfamiliar domain, the observations from this study also suggest that this may be true since in a familiar domain, the learners can possess assumptions based on their existing knowledge and experience which is beneficial but sometimes can also lead to biases and deviations from the presented problem, which was evident.

CLT was also helpful in explaining some of the other observations made in this study. As discussed earlier, the think-alouds were conducted during the COVID-19 pandemic, when the sessions transitioned to online due to lockdowns. In this scenario, both Morgan and Casey used an additional tool, Visual Paradigm, to develop the models. Blake used a tablet which was closer to the pen and paper used during the face-to-face think-alouds. While this was a more circumstantial decision, some interesting facts were observed with regard to tool use. While the conjecture is that tools can aid and enhance productivity, they can also add to the cognitive load in certain circumstances. Both Morgan and Casey had some difficulty using the tool. Casey, in particular, exhibited indications that the tool introduced a multitasking aspect, requiring Casey to consciously switch between doing the task, reading the case study and questions, and using Visual Paradigm. The tool splits their attention (Chandler & Sweller, 1992) and, increases the overall cognitive load, and hinders their learning by not having enough

working memory to deal with the task at hand – “*it’s all so overwhelming, you know, just trying to, you know, get your head around all that. It’s making my head spin with everything.*”

This observation suggests that teaching with CASE tools needs to be carefully considered. Either tool must be taught explicitly and included within the course content or not introduced until later when students have the capacity to cope with the extra load imposed by the tool. Such tools also contain many features that are not used by novices – these redundant elements add to the cognitive load of modelling tasks for novices (Kalyuga et al., 1999). The observations in this research suggest the latter might be the better option. There is the counterargument that when using pen and paper, novices find it difficult to recall the correct symbols and elements of a model. If a tool is introduced, then the issue arises of the stresses between the requirements to cover the core concepts and content in a content-intensive course where learners already exhibit difficulty grasping the concepts and skills (Goldweber et al., 2022), and the additional time needed to cover learning a tool. This research suggests that pen and paper or even a tablet could be a better tool for novices to start with when learning database modelling because it seems to reduce the additional cognitive load imposed by a tool.

On a more positive note, getting the participants to think aloud while attempting a task involves self-explanation. Some research has found that self-explanation can help learners with problem-solving and progress their learning (Aleven & Koedinger, 2002; Aureliano et al., 2016; Chi et al., 1989; Conati, 2016). But work by M.-H. Lin et al. (2015) found that self-explanation did not help novices when they attempted database modelling exercises. Lin et al.’s findings align with *the self-explanation effect* in CLT, which is considered to increase the cognitive load where learning happens (Kalyuga, 2011a), which suggests that explaining thinking aloud should hinder learning – this seems to be true for Casey. However, the best-performing participant, Morgan, said, “*It’s good to talk about what I’m thinking about, which made me sometimes realise ah... that’s wrong*”. These doctoral research findings suggest that the capability of the student affects the usefulness of self-explanation – for those students with some, even fragile, mental schema and knowledge of the basic concepts self-explanation is helpful.

Morgan, the best-performing participant, used all the business rules they developed to effectively develop the conceptual ERD despite having one error that interestingly got transferred from Task 2. This was an interesting observation, as it seemed Morgan directly translated all the business rules without much thought to the final conceptual model. It’s an odd case where they seemed to have an appropriate and systematic process and well-developed schema but automatically processed the new information without conscious or deliberate effort, which resulted in error propagation. If Morgan had controlled the process by critical evaluation, then it is highly likely they would have realised the mistake and corrected it. Morgan’s automatic approach to processing the task meant that they could operate at a lower level of cognition. Controlled processing requires significantly more cognitive resources to

undertake (Shiffrin & Schneider, 1977). It is hard to determine whether this automatic processing for this particular task was because Morgan found the step of shifting from business rules hard and thus followed the process automatically or the cognitive load of thinking aloud meant Morgan fell back on processing in an automatic fashion or that he felt the task was so easy it did not warrant the level of attention that controlled processing requires. Either way, it indicates that by doing this, Morgan reduced the cognitive load associated with the task. However, despite Morgan's limited experience they had with databases, they still don't have the experience of an expert to realise the patterns, and it's expected behaviour from a novice. This is interesting since the educators, including ourselves, emphasise a lot on using the process correctly, but some of them follow the process to the extent that they won't think, and it becomes a manual process. Again, this brings forth the debate as to whether Morgan managed to consolidate their knowledge or whether they are using the surface knowledge gained during the lectures. While Morgan was the best-performing participant, it could be argued that the tasks were isolated simple novice-level tasks, and that Morgan may not be able to transfer their knowledge into a more complex task. This brings the issue of the implications into teaching because, generally, educators are focused on using processes to compensate for the student's lack of experience. This is an example of where they understood the process and developed their own thinking into it but missing the evaluation component.

9.4 Learning Transfer

Learning transfer or transfer theory lies at the intersection of educational and cognitive psychology. It examines the ability to apply learnt knowledge or acquired skills in one context to another. Transfer theory suggests that learning in one situation can influence learning and performance in another situation either positively or negatively. From a learning perspective, transfer theory is concerned with how the learning of one situation influences learning in a new situation. In contrast, from a cognitive perspective, it explains how prior knowledge or experience stored in memory can be retrieved and applied in a new situation.

Learning transfer has been described from several different viewpoints, starting from the traditional view of Thorndike in the early 1900s (Woodworth & Thorndike, 1901). Thorndike's viewpoint was that learning of one context could be transferred to another when both the original and transfer tasks share similar elements and features (stimuli) and where a clear relationship exists between the tasks. Skinner's (1953) view on transfer is somewhat different from other theorists. Skinner believed that transfer occurred because of the generalisation of stimulus-response associations learnt from one context to another. More recently, Gagne et al. (1992) suggested that transfer of learning occurs when prior learning can be applied to new and different situations by activating knowledge in memory networks. According to this theory, when a learner encounters a new situation, they first search their

memory networks for relevant knowledge and experiences. If the learner can identify similarities between a known and new task, then they can activate the relevant memory network to transfer that knowledge or skill to the new context. Gagne et al. (1992) recognised the three types of transfer originally introduced by Thorndike, namely:

- *Positive* transfer – occurs when learners apply prior learning to a new context, resulting in improved performance.
- *Negative* transfer – occurs when the applied prior learning interferes with the learning of a new context, resulting in decreased performance.
- *Zero or Neutral* transfer – occurs when prior learning has no effect on the learning of a new task.

Perkins and Salomon (1992) provided deeper insights into learning transfer by defining the concept and how transfer occurs from an educational psychology perspective. They separated learning transfer into two types *near* and *far* transfer.

- *Near* transfer – describes the process that refers to the application of knowledge or skills from one context or domain to a closely related or similar context or domain.
- *Far* transfer – describes the process that refers to the application of knowledge or skills from one context or domain to a different or distant context or domain.

Recent work on learning computer programming used the notion of near and far transfer. They derived a transfer classification framework based on observations of their learners to define the extent of the knowledge transfer (from positive to failed transfers), which they called *transitions* (Izu & Mirolo, 2021). Their framework was framed around the transfer level between two related programming tasks with a common concrete subgoal. They examined learning transfer in programming at a very granular level, going beyond using just near and far transfers. This fine-grained model is useful when considering novice tasks.

The degree of distance between near and far transfer is open for debate. Perkins and Salomon (1992) consider that the far transfer link is remote or alien by giving an example of applying a chess strategy to investment practices. They give the example of students taking a final exam that is comprised of problems that are the same kinds of problems that they have practised during the course as being near transfer. Taking this into account, all of the tasks participants faced in this study involved near transfer. None of the tasks are outside of those encountered during the course. However, Izu and Mirolo (2021) claim that in the knowledge transfer between their two simple programming exercises, both near and far transfers are possible. For Izu and Mirolo's students at that time in their development, it is possible that far transfer is manifested and that the definition of distance of transfer is dependent on the level of expertise and knowledge and thus manifests differently for each individual learner. In this thesis, the

concepts of near and far are not used, instead, a more fine-grained classification of transfer is used that is adapted from those identified by Izu and Mirolo (2021).

Izu & Mirolo (2021) give the example of writing a function that determines if a number is an Armstrong number or not. The task specifies the range of valid input values to the function. The student solves this problem using a for loop. In a later exercise, the student is asked to write a function that can receive any positive integer and determine whether it contains the digits 3 and 7. The main difference is the parameter, where one problem has a defined range, and the other doesn't. In the later problem, the student uses a while loop, and this solution goes beyond basic transfer because the student realises that a while loop is a better solution when there is no range limit on the parameter. This student is considered to have extended transfer, whereas a student who used a for loop on the later problem was considered to have a consolidation-type transfer. In the context of this study, the tasks were not designed to trigger an extended transfer. However, the other forms of transfer identified by Izu & Mirolo are applicable.

In this thesis, the transfer types have been adapted to the context of novice database modelling, as presented in Table 9.1.

Table 9.1 Transfer types.

Transfer type	Transfer/Non-transfer	Adapted definition
Consolidation	Possible transitions	Consolidation is a positive transfer where students recall and reuse the knowledge from the previous task in a new context or a model. In this study, consolidation happens only when students attempt redirected task and then use that learnt knowledge to solve the main task where the problem domain/context is different or simplified.
Partial		A partial transfer is where the knowledge from a previous task is recalled and/or used but with some errors. For example, when a student recalls using an annotation system when analysing the case study but ends up with some errors in the solution due to inconsistencies.
Failed		A failed transfer is where a poor attempt to recall the prior task but fails or is unable to recall the knowledge from a prior task. For example, when students fail to recall the mapping rules they've used to transform a conceptual model into a logical one.
No transfer	Non-transfer transitions	A no transfer is where, instead of using the approach from the previous task, a different and inadequate approach is selected. The conjecture is that the previous approach was not learnt.
New insight		A new insight is where a suitable approach was used as an alternative to the previous approach. However, in this case, it's harder to conclude the previous approach was not learnt, as they might have replaced it purposely.

Even though the concept of learning transfer has been discussed in other education domains, it's not a widely explored concept among many of the subject areas in computing, except for programming. In their study, Perkins & Martin (1985) discussed novices' *inert knowledge* and how they struggle with tasks that require them to apply programming concepts in new and unfamiliar contexts. More recently,

several researchers have worked on different aspects of transfer related to learning in programming as follows:

- Cognitive factors affecting transfer – (Ginat et al., 2011)
- A classification scheme facilitating different levels of transfer for novice programming tasks –(Kasto, 2016)
- Development of teaching styles to facilitate transfer – (Margulieux et al., 2012; Morrison et al., 2015)
- Skill transfer between programming languages – (Dann et al., 2012; Tshukudu & Cutts, 2020)

While knowledge and skill transfer have been explored in the context of learning programming, the same is not the case when it comes to database modelling education. Only one article was found in the literature: a poster abstract on developing deeper self-directed learning for a database design course (Zyl & Mentz, 2022). They theorised that the mechanisms they used (i.e., restructuring course content holistically, worked examples, cooperative learning and real scenarios in problem-solving) promoted the transfer of knowledge and skills. Further, the authors indicated factors like rushing for time, misconceptions, and not applying problem-solving strategies obstructed knowledge transfer. While there is a lack of literature focused on this area in novice database modelling, the transfer of knowledge is important in database development since the students are required to apply the taught concepts in different scenarios. In the case of novices, typically, tasks involve the *near* transfer of concepts, processes, tactics, and examples taught in class to a new context. Usually, these tasks are framed in the form of a case study, which is designed to be familiar to students in order to remove the burden of additional external domain or context knowledge (Imbulpitiya et al., 2021). Moreover, for novices, the problems are often presented in manageable chunks or subtasks that build on each other sequentially, like the tasks used in this research. These subtasks involve students applying the knowledge developed from completing one subtask to the next subtask in the sequence.

Despite not developing the research instrument to explicitly facilitate and explore knowledge transfer, different types of knowledge transfer were observed during the think-aloud sessions. This section explores the notion of knowledge transfer in the context of this doctoral research with the aim of gaining insights into how knowledge transfer occurs, or not, when novice database modellers of different capability levels attempt relatively simple scaffolded modelling tasks. Several taxonomies of knowledge transfer were explored to define the various types of knowledge/learning transfer that occurred. The transfer types identified in Table 9.1 were all observed in this research. This suggests that the transfer types developed by Izu and Mirolo (2021) for novice programmer tasks can be reinterpreted and generalised and used to interpret novice computer science tasks and student performance on tasks in general, including database modelling.

When considering transfer in this thesis, there were two forms of transfer, one of which is related to the problem domain/context (i.e., a student management system). The other is related to the transfer of knowledge of the discipline, namely, database modelling. In Izu and Mirolo's (2021) work, they ignore the influence of the problem domain and focus solely on the transfer of programming knowledge from one task to another. Arguably, their problems are relatively context-free because the algorithm for the implementation is described in detail, so it does not rely on knowledge of Armstrong numbers. In database modelling, the context of the problem is central to being able to construct a solution. Students need to understand both the problem context and the modelling processes and concepts in order to solve the problem. Looking at the distance for transfer from this perspective means database modelling may involve far transfer in order to understand the problem context and near transfer when the transfer involves modelling knowledge.

Transfer of knowledge with respect to the problem context/domain introduced errors among participants in this study. Instances of this were observed and formed part of the “understanding the domain” theme (see section 7.8.3). This is considered to be a *partial* transfer (see Table 9.1).

There were instances where participants demonstrated *consolidation* transfer when they were given a redirected task. For example, Blake, after attempting a redirected task, was able to progress in the main task of creating business rules. This suggests that there was a transfer of knowledge from the redirected task to the main task, which consolidated their understanding. However, it is important to note that to draw meaningful conclusions and identify the implications for teaching in relation to *consolidation* transfer, a properly developed research instrument that specifically addresses transfer is needed.

The majority of the transfers observed in this research were *partial*. The participants could recall and use an approach they used in class to complete the current task but ended up making mistakes. For example, Blake could recall and use the process for resolving many-to-many relationships and apply it to the current problem, which was different from others they had encountered. On the surface, this may be considered a consolidation transfer, but in reality, the transfer was incomplete. Blake could not recall or use the other processes needed to convert a conceptual ERD into the logical form – meaning the transfer was partial on this task. Morgan, a very capable student, was also only able to transfer their knowledge on this task partially. Blake and Morgan mentioned that they could have recalled the other mapping rules if they had been working on a familiar example they had encountered before. As novices, it seems that they had not seen enough examples to build a schema that was generalised enough to cope with different contexts. This may be the reason for their *partial* transfer. A fully formed schema may have enabled a *consolidation* type transfer. This raised the question as to whether these students are operating at recall level when solving database modelling tasks rather than applying their knowledge as is expected in such exercises.

Since the tasks were sequenced, the participants who performed better managed to identify shared principles and relevant information from one task to the next, transferring the knowledge gained from one task to the next. Morgan showed evidence of this, “*I’m going to use my previous solution. So, I know the entities and their relationships since I sort of developed it and then I’m looking at my business rules when I decide ... what do you call it, multiplicities*”. In this instance, Morgan managed to effectively transfer what they had learnt in Tasks 1 and 2 to develop the conceptual model.

If participants cannot see the commonalities/ patterns between the tasks, they cannot transfer relevant knowledge. This was evident from Casey’s narrative, who often failed to transfer knowledge between tasks. These observations imply that novices need more examples and explicit teaching that highlights patterns and commonalities in tasks. The use of faded worked examples would assist such transfer and reduce cognitive load (Renkl et al., 2004). In addition, using variation theory to inform the design of examples may help. Variation theory suggests highlighting the essential features of a concept by focusing on what remains constant and what changes help to learn (Gu et al., 2004). This involves systematically varying critical aspects of a concept while keeping other aspects constant, providing students with an opportunity to make meaningful connections.

Casey consistently struggled to transfer knowledge effectively, leading to *failed* transfers and even *no transfer* in certain instances. Casey’s failed transfers were often linked to their inability to recall a taught process or tactic. Moreover, Casey could not transfer knowledge from one task to another. For example, Casey, when trying to decompose the relations of the provided data table into third normal form relations, started writing SQL create table statements resorting to a more concrete and known representation. Using this approach was not taught nor useful to the task at hand. Here Casey is transferring inappropriate knowledge in an inadequate attempt to solve the problem. This is classed as a *no transfer* type.

During the normalisation problem, Morgan had their own approach to solving the problem. Morgan typically understood the taught approach, but they purposively replaced it with their own. They expressed that this was because they preferred a more practical, concrete approach closer to specifying the physical database. In retrospective discussions, it emerged that Morgan used that concrete implementation focus, viewing the taught process as more, in their words, “theoretical”. Regardless, Morgan managed to complete the problem and was able to rectify any minor errors they introduced. Arguably, Morgan’s approach could be classed as a *new insight transfer*.

One important observation related to transfer theory was that while it is possible for students to transfer knowledge, the participants in this study commonly transferred incorrect or incomplete information. Simply transferring knowledge without critically examining and verifying its accuracy can lead to

misconceptions and errors. To promote effective knowledge transfer, it is important to encourage students to engage in critical thinking, reflection, and evaluation to help them develop a deeper understanding of the concepts and principles involved.

9.5 Design Thinking

During the exploration of data from various theoretical perspectives, a recurring question arose: can any existing theories effectively explain how novice designers approach their design problems? Numerous studies in the literature examine software development and project management methodologies through the lens of design thinking (e.g., Adikari et al., 2013; Gabrysiak et al., 2011). Additionally, research has used design thinking and computational thinking as ways to gain insights into how individuals tackle programming problems. None of the literature on database modelling examines database designers through the lens of design thinking and design processes. This section attempts to look at the participant observations through this lens.

Kelly and Gero (2021) provide a theoretical foundation for the connection between design thinking and computational thinking as a dual process model for addressing design problems. While their work is a far stretch from pure design thinking, it provides a solid foundation to discuss some of the findings of this study. The concept of design thinking comes from the “design science” domain and sees design problems as “...*problems which are both goal-oriented and constrained and which depend upon a designer’s perception of the context of the problem*” (Kelly & Gero, 2021, p.3). This applies to database modelling since a solution for a database problem always depends on the designer’s perceptions and experiences. Different designers can arrive at different solutions to the same problem in complex problems. The core of design thinking is the “... *understanding that a designer creates the **frame** within which design activity is undertaken*” (Kelly & Gero, 2021, p.3), where the frame is comprised of the designer’s unconscious assembly of interrelated knowledge when presented with a design problem.

Research of designers provides insights into how they frame the problem and their strategies when reframing the solution as it evolves. Effective designers have been found to have an understanding of the problem, problem space, and solution space. Designers typically co-evolve an understanding of the problem, which changes, and the solution, keeping in mind there is more than one solution. The final solution evolves through an understanding and evaluation of the possible solutions within the solution space (Poon & Maher, 1997). However, novice database modelling tasks are typically not large and complex enough to encompass a large solution space or a changing problem space. In this study, the task was constrained in terms of solution space, and there were no changes to the problem, but this does not mean that the participant’s understanding of the problem did not evolve as they worked on solving the problem and arriving at a solution. For example, there were instances where Morgan, the best-

performing participant, may have considered alternatives and evolved his understanding of the problem based on their knowledge in the real world.

Inherent in design is making unexpected discoveries over the course of the design process, which lead to different solutions evolving (Suwa et al., 2000). Architects have been observed not only synthesising requirements but also inventing requirements that capture important aspects of the problem not previously defined (Suwa et al., 2000). The best-performing participant, Morgan, often relied on their existing knowledge of the context or their real-world experience applied in the design process, which sometimes was useful and, at other times, was not unlike in Suwa's study. This implies that, as a novice, they could not differentiate their knowledge of the case study problem domain from what they have been provided cognitively.

Kelly and Gero (2021) argue it is possible for software designers to move seamlessly between design thinking and computational thinking. However, the novices who participated in this study were more in a design thinking phase where their solutions were specific to the problem scenario, and the framing was more general, where they considered every aspect of the case study and even relied on their own experience from the real world without abstracting anything. This could be a reason why, in the study, the novices never claimed that they had seen similar design patterns, even though Kelly and Gero (2021) claimed that designers tend to use patterns as a part of their design thinking. It is likely that novices have not yet developed design patterns. They cannot recognise common patterns when designing database models.

While looking for the potential literature related to design thinking and database modelling, the work of Welch and Lim (2000) emerged. Although their research is focused on physical designs and novices, the similarities between their findings and those of this thesis are worth considering. The parallels are both design tasks involve problem-solving, analysis, and decision-making. Welch and Lim (2000) highlighted that in physical design (i.e., developing a paper tower), the process needs to be grasped as a whole, not as unrelated parts. This is also true for database modelling. For example, students can learn about business rules, modelling constructs, etc., in isolation, but it is important to understand how these individual concepts are interconnected and work together to create a coherent and functional database model. The participants' difficulty in linking and consolidating knowledge during the transfer process raises questions about their overall design process.

In their study, Welch and Lim (2000) developed a framework that includes five steps students use to build a physical design artefact: understanding the problem, generating possible solutions, modelling a possible solution, building a prototype, and evaluation. In this doctoral study, a prototype is not built, though, in the course, students do implement a model based on their own design model as part of the

take-home assessment. The steps we expect to see in this study are problem understanding, generation of solutions, and evaluation of and refinement of solutions.

Both Morgan and Blake followed a process of understanding the problem and generating a solution while learning more about the problem through generating their solution, but they failed to evaluate and validate their solutions. For example, starting with activities centred on reading and comprehending models, followed by critiquing models before eventually engaging the students in the stage of developing the models. This raised the questions of how to teach novices to critique their solution/s and evaluate possible solutions and whether students would do better if all modelling tasks led to a concrete implementation so that they could better understand the implications of different design choices. One solution discussed in the literature was to deliver the course in a spiral model, as discussed by Watson (2006), which is adapted to a certain extent to the course delivery where the participants were enrolled. Given the challenges encountered by the participants despite the way of course delivery, it may be beneficial to think of a delivery approach aligned with skill levels and progressively advance from there.

9.6 Novice Modelling Styles

A recent article⁷⁸ proposed modelling styles as a means of analysing and understanding how novice modellers approach a design problem (Rosenthal et al., 2022). This work characterises a modeller's sequencing of processing based on how they apply the modelling language and its concepts when constructing a model. Rosenthal et al. (2022) observed three different modelling types among database modellers:

- *Entity type-focused modelling* style – modellers who create all the entity types in the model, including the attributes and data types, followed by the relationships and the cardinalities.
- *Linear modelling* style – modellers who follow the sequence of statements provided in the problem description/case study from beginning to end.
- *Structuring modelling* style – modellers who create the entities and the relationship between the entities first and then add the relevant attributes and data types followed by cardinalities.

⁷ Modeling styles in conceptual data modeling: Reflecting observations in a series of multimodal studies.

⁸ Since Rosenthal et al. (2022)'s article was published during the writing of this thesis, the modelling styles of the modellers were not considered while the data was analysed.

It is important to note that they examined the continuum from novice to expert. Here the participants are mapped to these styles.

Morgan and Casey's modelling style tended to be what Rosenthal et al. (2022) defined as *entity-type focused* as they added all the entity types first, including the attributes, then the relationships and cardinalities. Rosenthal et al. (2022) state that most inexperienced modellers use this approach – indeed, all the novice participants in their study used this approach. In contrast, Blake had a linear modelling style, strictly following the sequence of the case study when modelling entity types, relationship types, attributes and cardinalities as they were encountered sequentially in the case study. Rosenthal et al. (2020) claim that this is the approach used by intermediate and expert modellers. However, earlier work found that novice modellers tend to follow a linear process when working on modelling tasks, whereas experts work on one part of a case study at a time to seek clarifications, integrate information, and recognise similar situations before representing the information (Batra & Antony, 1994; Batra & Davis, 1992). While research on expertise and database modelling processes is contradictory, Casey, who adopted an *entity-type focused* process, struggled to navigate and manage the case study. Perhaps for a weaker student like Casey, using a linear modelling style is more useful. This conjecture warrants further investigation.

9.7 Summary

This chapter focused on different theories that could explain the findings and observations identified during the analysis. The theories of Piaget and Sfard can explain some of the learning observed among the participants during their think-alouds. While these theories are useful in explaining knowledge conceptualisation, they do not provide insights into *how* to facilitate schema construction or design activities in such a way as to facilitate learning. In that aspect, Vygotsky's notion of ZPD was more helpful. In this thesis, it informed the development of an intervention model that was able to progress the learning of the participants. The interventions were provided hierarchically to describe the level of support needed by each participant, facilitating personalised assistance based on their struggles. The evidence from this study shows that the concept of ZPD and using interventions is useful to describe and research the learning of novice database modellers. While progressing through the sessions, the intervention protocol allowed the interviewer to predict which type of assistance would be most useful, though this observation is anecdotal.

In this research, CLT was useful to explain and interpret why some participants couldn't progress or had difficulties with certain aspects of learning. The research instrument was intentionally developed in a manner that aimed to reduce intrinsic and extraneous cognitive loads and free the working memory to construct schemas. This was considered important as prior research had noted that the intrinsic load

of database modelling tasks are relatively high due to the higher number of interrelated concepts involved. Moreover, as the participants were novices, it made sense to consider CLT while developing the research instrument. For the participants who had difficulty progressing, it was likely due to the cognitive overload they faced, which led to partial and failed transfer of knowledge. The participant observations and comments suggest that the segmentation of tasks, structured case study and redirected tasks helped them work on the tasks. When the findings of this study are examined through the lens of learning transfer, design thinking and novice modelling styles, insights into what was observed can be useful.

The next chapter presents the conclusion of this doctoral research by revisiting the research questions and highlighting the key findings and contributions. Further, the chapter will discuss the limitations the researchers faced while conducting the study, the implications of their findings in teaching database modelling, and the future avenues this research can be manoeuvred into.

Chapter 10 Conclusions and Future Work

10.1 Overview of Research

This thesis set out to explore difficulties faced by novice database modellers, the mechanisms and tactics they used to overcome the encountered difficulties, and how individuals with different capabilities progress in their learning. To initiate this research, a systematic literature review was conducted as a mapping study. This systematic literature review was conducted to get an in-depth understanding of the published literature in the domain of database modelling education. Subsequently, a question classification framework was developed to gain insights into the nature of the typical examination questions given to novice database modellers. A set of examination questions from several institutions was classified. The classification system included a cognitive dimension using an adaptation of the revised Bloom's taxonomy for classifying novice database modelling questions. Further, database metrics were utilised to predict the task difficulty, and an attempt was made to establish a hierarchy among the database modelling tasks that could be utilised in the research instrument design.

The overarching objective of this doctoral research was to gain an in-depth understanding of how novices learn to model based on their learning processes, developmental stages and how they acquire the required knowledge as novices. This objective was achieved by following novice modellers enrolled in their first database course at AUT, attempting two simple database modelling tasks presented in a scaffolded and sequenced manner. The participants were asked to think aloud and explain what they were doing while solving the problems. The data gathered from the think-alouds was analysed using thematic and narrative analysis. The findings were then related to theories of human conceptualisation in order to attempt to explain the observed learning processes of the novice database modellers in this study.

This chapter starts with a summary of the findings and contributions related to the research questions, followed by a discussion on the threats to validity, the implications of the findings to teaching and the limitations faced while conducting the research. Finally, the chapter concludes with future research directions.

10.2 Review of Research Questions

This research was guided by four main research questions, and this section revisits the research questions formulated in section 1.3.

RQ1: What are the existing focus areas of research in database modelling education?

The main objective of this research question was to investigate the current literature related to the teaching and learning of database modelling by conducting a comprehensive systematic mapping study which was presented in Chapter 2 . This understanding of the existing literature was critical for this doctoral research in order to realise the scope and focus of any existing work and to identify any gaps and trends in the area. This knowledge was needed to be able to develop the research instrument. To this end, a systematic mapping study was conducted covering 30 years of research from 1990 to 2020. One hundred and fifty-five relevant articles related to novice database modelling were identified. Several areas of research were uncovered, namely: *Tools and resources*; *Pedagogy*; *Curricula design*; *Course design*; *Assessment*; and *Teaching/learning challenges*. Among these, the *Assessment* theme was explored further. The intention was to identify various types of tasks that could be given to the participants and to inform the design of questions in an examination to test the notion of a hierarchical structure of novice modelling tasks that could be useful in predicting the difficulty of such tasks. Some of the findings from this literature review later informed decisions made during the research instrument design phase of this research.

At the time of this review, there were no existing systematic reviews found in the domain of database modelling education. Hence, this in-depth review contributes to the body of knowledge by providing a broader picture of the literature in the database modelling education domain⁹. The findings of this review highlighted the need for more research in the area with an emphasis on rigorous empirical evaluation of tools, proposed pedagogies, and assessment designs in order for teachers to build better educational experiences for their students.

RQ2: Can a question classification framework be established that describes the difficulty of novice database modelling tasks?

The main aim of this research question was to investigate whether a question classification framework could be developed that reliably allows the difficulty of novice database modelling tasks to be determined. The intention was to use the framework to inform the design of the tasks used in the later think-aloud study of this thesis. This question was answered by addressing three sub-questions:

- i) RQ2.1: What is the nature of database modelling tasks used to assess novice database modellers?

⁹ Towards the end of this doctoral thesis, a systematic literature review was published (Ishaq et al., 2022), but in comparison, the review in this thesis is more comprehensive and in-depth.

To answer RQ2, it was necessary first to comprehend the nature of typical novice modelling tasks. As a preliminary step, a classification scheme based on question style, concepts involved, and the cognitive levels of the revised Bloom's taxonomy was considered. Bloom's was enhanced with definitions and vignettes directed at novice database tasks (presented in Chapter 4). This classification scheme was built based on a set of novice database modelling examination questions. This was the first attempt to develop an adaptation of Bloom's and provides a system that can be used by future researchers to classify novice database modelling tasks.

The analysis highlighted a major gap in the cognitive levels in the examination questions sourced for this study. None were at the *Analyse* and *Evaluate* levels. There were questions at the *Create* level where it can be assumed the skills of *Analyse* and *Evaluate* are needed to arrive at a viable solution, but these skills are not explicitly assessed.

ii) RQ2.2: What type of measurements can predict the difficulty of database modelling tasks?

The study was expanded to check whether a task's difficulty can be measured using database metrics (see Chapter 5). Several structural conceptual database metrics that could potentially predict the complexity of a novice database modelling task were assessed. However, statistical analysis failed to find any significant correlations between these metrics and question difficulty as measured by student performance. The intellectual complexity classified using the cognitive levels of the revised Bloom's taxonomy also failed to predict question difficulty. These findings may be due to the limitations of the question set, including sample size and question types. The intent had been to expand the data set in a second round of data collection with questions targeting evaluation and analysis of models to be included in an examination. However, due to the pandemic lockdowns, no controlled assessments or examinations were held during the remaining data collection period. In the end, as explained in Chapter 6 the think-aloud tasks were designed based on the number of concepts required to complete a task as a proxy for difficulty, an idea adopted from recent work in programming (Luxton-Reilly et al., 2018).

i) RQ2.3: Does the performance of students in database modelling tasks indicate a hierarchical structure in their mastery of different types of novice modelling tasks?

Another hypothesis tested during this analysis was that there is a hierarchy of skills in novice database modelling tasks. However, no hierarchy was found, at least within the context of this study, even though intuitively, one should exist. This analysis is presented in Appendix K.

RQ3: How do novice database modellers incorporate new modelling concepts and knowledge into their existing understanding of database modelling?

Through the thematic analysis presented in Chapter 7, difficulties faced by the novices were identified, along with the mechanisms/tactics the participants used to overcome these barriers. Seven themes were identified: *Interpreting the case study*; *Problem presentation*; *Understanding the problem domain*; *Verification of design decisions*; *Fragile knowledge*; *Problem recall*; and *Tools*. The following questions were considered through this thematic analysis:

- i) RQ3.1: What difficulties do students face when attempting novice database modelling tasks?

The most common difficulty faced by the participants, regardless of capability, was their fragile or incomplete knowledge of foundation database modelling. Consequently, the participants' barriers to solving a task often revolved around fragile understandings of the relevant conceptual knowledge, syntactic knowledge and/or terminology.

The participants also had difficulty transferring their knowledge from one task to another. This is a necessary skill since developing a logical database model involves following several steps in the DBLC. To solve the think-aloud tasks, participants needed to transfer their existing knowledge gained in class to the new task context of this study. This transfer involves the ability to generalise or abstract. Additionally, the participants struggled to understand the case study context. While this was sometimes evident with the participant whose first language was not English, interestingly, participants whose first language was English also struggled with the narrative of the case study in instances where the sentences were long and connected with conjunctions.

Both the model development and normalisation problems were designed and presented in a manner to reduce the cognitive load that comes from interrelated concepts required in developing the solutions. Despite the intention, all participants exhibited cognitive overload, particularly the weaker two participants. The participants, regardless of ability, also showed a lack of attention to detail.

This study contributes to the body of knowledge as one of the few studies of novice modellers that use a thematic analysis method. The thematic analysis identified the difficulties and challenges faced by novices while attempting the modelling tasks (see section 7.9.2) and the tactics they used to overcome these issues they had (see section 7.9.1). Some of the findings identified confirmed the findings of those reported in the literature. One new insight in this doctoral thesis analysis was the lack of ability of the participants to transfer knowledge from one task to another. This was not an area which had been explored to any great extent in the published literature. A further contribution is the themes developed, which can be further explored and adopted by other researchers to investigate their learners.

- ii) RQ3.2: What common tactics do students apply when attempting novice database modelling tasks?

Several tactics were found to be used by the novices through the thematic analysis: annotating the case study; doodling; making assumptions; applying real-world knowledge; one-to-one translation of the case study into modelling constructs; and verifying the design decisions. Annotating the case study was a common tactic used to analyse the case study text. While not all participants used a consistent annotation system, this was a taught process on par with the noun-verb analysis taught in class. Making assumptions when information was considered inadequate to make informed decisions was another taught tactic. However, the participants who attempted this were unable to use appropriate external knowledge and made assumptions about aspects of the problem that were actually detailed in the case study. Some participants used doodling to visualise key aspects of the case study, which yielded partial diagrams that were helpful in later tasks to develop their models. The participants who used doodling performed better. Less common tactics seen were one-on-one translation of the case study, summarising the understanding, and verifying the decisions before advancing further during the analysis of the case study. Among the participants in the study, it was observed that only those who performed well took the initiative to validate their comprehension of the decisions they made before progressing to the subsequent sections of the case study.

While there was published literature related to the difficulties and different tactics used by novices to overcome these issues, none highlighted the importance of interpreting a case study and annotating as a useful tactic during this process. Another new insight was the use of doodling while attempting modelling tasks, which seemed to be favoured by the better-performing participants. Additionally, this finding aligns with existing literature highlighting the value of visualisations. Another rare tactic was verifying design decisions. This finding resonates with prior research suggesting that novices struggle with assessing model quality. Overall, the new insights identified via the thematic analysis offer researchers a set of tactics to investigate further. Moreover, knowing which tactics are effective and when they can be usefully applied would improve the teaching and learning of novice database modellers.

- iii) RQ3.3: How does the learning of novice database modellers of different capabilities develop over time?

In order to gain a deeper understanding of how novices develop their knowledge and progress in the context of database modelling tasks, the think-aloud data were subjected to a narrative analysis that followed three selected participants' learning progression, as detailed in Chapter 8. It was evident from the narratives that the best and middle-performing participants showed clear progress in learning and

developed their knowledge as they progressed through the tasks. Overall, they could recall and apply their existing knowledge to the problem at hand and transfer newly developed knowledge to solve the next task in the sequence with assistance. Thus, while they struggled and faced barriers, they persisted and could successfully complete the tasks with the interviewer's assistance. The lowest-performing participant exhibited fragile knowledge, which led to an unmanageable cognitive load. They showed no real learning progress or knowledge development.

This is the first study, to the researcher's knowledge, that has attempted to understand how novice modellers learn to model as they engage in a series of interconnected tasks spanning multiple sessions. This approach deviates from the existing work in the published literature, which often focuses on isolated individual tasks. Further, the visual representation of the learning progression and trajectory contributes to the body of knowledge by introducing a format for quick interpretation and comparison of student progress.

Based on this analysis, various learning and cognitive theories were considered, as discussed in Chapter 9 to interpret further the findings of the qualitative analysis. Several theories related to knowledge construction, cognitive load, knowledge transfer, and design thinking were considered during this discussion. Of the theories and many facets of the theories discussed, certain ones provided a more useful and worthy of future more targeted research.

Three knowledge construction theories were examined. Between Piaget's and Vygotsky's work, it was evident that Vygotsky's notion of ZPD was more helpful in understanding novice modellers. Designing learning experiences like the ones used as interventions in this study proved to be helpful for students who have, to some degree, mastered the basic concepts. However, those who do not need help to consolidate their understanding of concepts before being ready to attempt even simple modelling tasks.

Vygotsky's notion of ZPD provided a lens to help identify when a participant could complete tasks independently, with the assistance of an MKO, or simply could not, despite the assistance. This helps to establish the readiness for learning and what interventions might advance knowledge.

Sfard's theory aligns with Piaget and Vygotsky however, it is not ideally suited for time-constrained studies like this thesis. Based on the outcomes of this research, this is a theory worthwhile delving into if the researchers are planning for a more longitudinal approach and increasing the level of concepts and difficulty in tasks where there would be enough time for the concept development to be established. Otherwise, Vygotsky's ZPD emerges as an effective theory that future researchers can adopt when working with novice learners.

Considering CLT was found to be useful for shaping research instruments. This research, informed how the instrument was designed to mitigate intrinsic and extraneous cognitive loads. Several effects were used that were found to give notable support to the students, including segmentation of tasks, structured case study, redirected tasks, and sequenced sessions. These elements can not only aid researchers in informing their studies but also have the potential to improve course and assessment design.

Transfer theory has not had much attention, though it holds particular relevance in the domain of database modelling, as novices are required to transfer their knowledge across two critical dimensions, domain-specific knowledge and the conceptual understanding of database modelling, in the literature. This thesis presents a novel taxonomy of transfer types for novice database modelling tasks founded on the literature and the thesis findings. This is a taxonomy that could be verified and used by future researchers. It also provides a means for considering the complexity of simple database tasks that can be used by educators to inform the design of such tasks.

Overall, these insights serve as guidelines for other researchers and educators, providing insight into which theories they might adopt in their research and teaching. From an examination of the literature, it is evident that nobody has looked at database modelling tasks and how students are progressing, relating them through multiple theories as done in this thesis. Thus, this work forms a new contribution to the body of existing knowledge.

10.3 Online Think-aloud Method

The emergency shift to online think-alouds, as presented in section 7.3.3, required a new data-gathering protocol to be established. A literature review was conducted to inform this change, and a protocol with guidelines for conducting think-alouds with tertiary CS students was established. Ethical considerations for online think-alouds were established that can be used by other researchers wishing to conduct such studies. This was an unexpected contribution to this doctoral study.

Working virtually and simultaneously engaging in think-aloud activities can indeed present challenges, particularly when dealing with multiple tools and technologies. However, in this study, no significant issues were encountered, which may be attributed to our participants' proficiency in technology, as they were CS students. There were initial issues with sharing screens on MS Teams, but these were easily overcome and highlighted the need for practice sessions to deal with any technical issues. Other issues also emerged with managing all the information on a screen and using the Visual Paradigm modelling tool, highlighting necessary considerations for future researchers.

Online fatigue was noted by the participants, but they commented that what they missed most was the face-to-face interaction. When designing online think-alouds, breaks need to be scheduled every 15 minutes to relieve this.

Considering the well-being of participants is important. In this study, the interviewer and participants had already interacted face-to-face, making the transition smoother than it might otherwise have been. The participants sought more reassurance and connection with the interviewer during the online format, in part due to the online format and fatigue but also due to the isolation imposed by lengthy lockdowns. Establishing rapport and allocating extra time before and after the sessions became crucial in the online mode. The interviewer did not experience notable additional fatigue when moving from in-person to online sessions. Overall, the study highlighted the importance of personal connection and support in the online setting to ensure participants' comfort and engagement during the research. Further, from the researcher's perspective, the virtual sessions allowed increased flexibility in scheduling and saved participants' time and money by eliminating the need for travel.

The participants demonstrated clear appreciation for the additional time and support provided by the interviewer in the online format. Some of them expressed thinking aloud online was more comfortable, possibly due to reduced performance pressure compared to the in-person mode. Interestingly, the overall transition to the online format seemed to have little impact on the participants, even for those who preferred the in-person mode. This suggests that the online format was well-received and effective in facilitating the think-aloud sessions.

10.4 Implications for Teaching

10.4.1 Designing novice modeller tasks to promote learning

It is well established that database modelling tasks pose challenges for novice learners due to their abstract nature and the presence of multiple interconnected concepts. These factors contribute to the potential for increased cognitive load, making the task more difficult for individuals who are still acquiring foundational knowledge and skills in this domain. The intervention/redirected tasks, which reduced the number of concepts a task involved, proved to be effective. The framework of intervention tasks is another contribution of this study. However, these interventions were found to be helpful only when the participants had knowledge of the foundation concepts in the task, even if that knowledge was weak. Further, it was evident that students cannot successfully start a task until they have this foundation knowledge. It is a challenge to develop tasks that are focused on atomic concepts, but for educators, it's suggested to design exercises isolating or minimising interrelated concepts until those early concepts are mastered then new concepts can be introduced in a scaffolded manner. This study also provided evidence that segmentation of a task into manageable chunks also helped in reducing the cognitive load

that arises from interrelated concepts this is also an approach that would lead to the design of more effective learning and assessment tasks.

Another important aspect is the way in which the case study or problem context is dealt with. In this research, the problem context was familiar to the participants, and this was intentional to avoid the cognitive load that could arise from an unfamiliar context. However, participants still encountered difficulties related to understanding the context of the problem and made incorrect or unnecessary assumptions. Providing a structured case study proved to be an effective strategy which facilitated participant engagement by enabling them to concentrate on individual sections in the case study. For the best-performing participant, this structured presentation helped to realise an assumption they had made was unnecessary. However, incorrect and unneeded assumptions were still made despite the focus brought through the structured case study. One potential teaching approach to mitigate such issues would be to have students work in pairs and foster an environment that encourages them to question assumptions (or decisions) made. Another way to support learning might be to provide exercises that ask students to critically evaluate and justify design decisions.

This research found that fully worked examples were not needed to progress the learning of the stronger students, while weaker students did not ultimately benefit from fully worked examples. In such cases, employing faded worked examples (Renkl et al., 2004) might be worth investigating. This study showed that the novices involved struggled with terminology regardless of their performance level. As suggested by Mason and Seton (2020, 2021), using a glossary of terms in each student's native language is worth considering to mitigate this issue.

In this research, none of the participants made use of additional resources even though they were free to use the internet or other resources to check their understanding. This reflects findings from a study of novices in which only one participant made use of the programming language documentation (Whalley et al., 2023).

10.4.2 Self-explanation for learning

Engaging students in thinking aloud about a problem using a formalised online protocol incorporates the practice of self-explanation – a cognitive process whereby individuals articulate their thoughts, reasoning and problem-solving strategies (Chi et al., 1989; Conati, 2016). In fields other than database modelling, self-explanations have been found to be associated with learning gains (see section 9.3). However, the only study involving students learning database modelling found that self-explanation did not lead to improved learning M.-H. Lin et al. (2015).

However, there is some evidence that the participants in this study experienced benefits from articulating their thought processes to the interviewer as they developed their database models. Thus, the findings of this thesis suggest that introducing opportunities for and encouraging students to adopt self-explanation as a learning strategy for comprehension monitoring, making inferences and connecting new information to prior knowledge might improve learner outcomes. Even very high-achieving students can benefit from self-explanation (Chamberland et al., 2011). It should be noted, however, that the effectiveness of self-explanation is believed to vary depending on a number of factors, including the specific task, learner characteristics, and instructional context.

10.4.3 The need, or not, for CASE tools

In this research, richer data was obtained when students used pen and paper or a tablet rather than Visual Paradigm, and it was evident that using the Visual Paradigm imposed a significant cognitive load on the participants who used it. This raises an important question for educators to consider: Is the incorporation of a CASE tool necessary at this stage, or is it worth delaying its introduction? Does the use of a tool hinder the learning of modelling at early stages? When is it appropriate to introduce a CASE tool? Is there adequate time in the course to teach the tool and the modelling concepts? Ultimately, this is a decision that educators must carefully consider based on their course, students, and learning objectives.

10.4.4 Use of diagrammatic representation

It was observed that during the normalisation task, the participants were able to identify their errors when they were directed to use a visual dependency diagram instead of an algebraic representation of data dependencies. This is in line with the findings of Katz (2018b) that students using a visually guided database design approach made fewer errors. It can be argued that providing students with normalisation problems that require them to use a dependency diagram is beneficial, and redirected tasks using this representation may help to consolidate knowledge. Moreover, providing students with many examples is recommended to give them the opportunity to form a generalised abstraction of this diagrammatic representation.

10.4.5 Course design

Two participants relied on concrete examples during the study to understand the modelling process. They had some but limited experience in database development from a high school class, so they were naturally falling back on past concrete and familiar examples. The other participants had no prior experience and, therefore, had no idea of physical databases or how to implement a database. Hence, they had nothing to fall back on. Subsequently, they sometimes found the tasks more difficult than the

other participants with some prior exposure to database implementation. However, the participants who always relied on concrete examples did better than the others.

The participants who relied on concrete and familiar examples performed better than the others. In this study, the learners found it easier to understand conceptual and abstract knowledge required in modelling when they had concrete examples to relate to. This observation indicates that courses should not be designed first to teach modelling and then teach implementation, but rather these topics should be taught concurrently in an approach that teaches bite-sized design concepts and the matching implementation in an iterative manner, and this “spiral approach” was first suggested by Watson (2006). But, the question remains as to whether this approach is sufficient to overcome the novice difficulties observed in this thesis.

Considering cognitive apprenticeship (Collins, 2006) as a pedagogy might be worthwhile. Cognitive apprenticeship is a model of instruction in which novice learners work closely with skilled experts to acquire knowledge and skills. Cognitive apprenticeship strategies fall into three groups: (1) *modelling*, *coaching* and *scaffolding*, providing students with a space to acquire an integrated set of skills through observation and guided practice, (2) *articulation* and *reflection*, methods that allow the students to observe expert problem-solving strategies to reflect on their own strategies, and (3) *exploration* allow learners to explore and experiment with tasks and concepts to encourage learner autonomy. These strategies facilitate a learner’s observation, acquisition, and externalisation of implicit processes and techniques during learning. Many of these strategies were followed during this research study and helped to progress learning therefore, it might be worthwhile to consider this model as a holistic framework for designing future database modelling courses.

10.5 The Impact of COVID-19

The occurrence of COVID-19 had a significant influence on the decisions made during this doctoral research. The timing of the PhD research, which began in early 2020, coincided with the first lockdown in New Zealand. Consequently, the study design and methodologies had to be adapted to accommodate the lockdown restrictions. Originally, this research was going to focus solely on a quantitative research approach, collecting naturally occurring data from controlled assessments. However, during the first phase of data collection, it became evident that the data obtained was insufficient to meet the research objectives, and a second data collection phase was planned that added new questions to encompass the missing *Analyse* and *Evaluate* levels. However, due to the uncertainty of the trajectory of the pandemic, AUT and the school (ECMS) made the decision to offer only uncontrolled assessments for the foreseeable future. As a result, a decision was made to revise the research methodology and incorporate an observation protocol that would be suitable for a smaller number of students, with a research study

instrument designed specifically for this doctoral thesis while keeping the quantitative data with the intention of triangulating the data to understand how novice modellers learn to model. This change required submitting another ethics application. One positive was that this change allowed for a more in-depth exploration of the research topic and provided valuable insights that would have been missed with a purely quantitative approach.

10.6 Limitations

Only examination questions were considered when developing the question classification framework and attempting to establish measures of task difficulty. Exams are not the only assessments in a typical database course, and different types of questions might have been identified if assignments and other assessments had been considered. However, because of the need to interpret student performance as experiencing difficulty, it was appropriate to restrict this data to controlled assessments only. While the number of participants was sufficient, the number of questions and type of questions was limited, and the questions did not include ones that required analysis or evaluation of a model. It is possible that with a more comprehensive data set, a metric and a hierarchy of skills or task types may have been found.

As discussed earlier, a curriculum change happened between the data collection phases. Due to that, more advanced questions from the pilot study could not be used in the full study. In the end, this did not impact the findings due to the level at which participants were from the full data collection phase. None of the students were so advanced as to complete the simpler tasks easily, and the extended task would most likely not have added to the findings.

Qualitative analysis methods (thematic analysis and narrative analysis) have many strengths but are limited in terms of the lack of generalisability of the findings. Further, reflexive thematic analysis involves the researcher making a subjective interpretation of the data, which can introduce personal biases. However, this thesis acknowledges these limitations, and several steps are taken to mitigate such issues, as discussed in section 7.6. An obvious limitation of using the think-aloud method, which is already discussed in the thesis, is that it may interfere with the main task the participant has to perform. Precautions were taken to make sure that the participants were comfortable doing both simultaneously. For example, they were provided with a practice think-aloud where they got an opportunity to practice the method and ask questions to get familiar with the data collection method.

Another limitation that needs to be acknowledged is the use of data from a single institution and within the context of one course's pedagogy and assessments. It is possible that students taking different courses might experience different difficulties and use different tactics, and their learning might progress differently. However, many of the difficulties seen in this study are reported in the literature.

The ethics requirements for the in-person think-aloud interviews prohibited the researcher from recording participants, which posed a challenge during online interviews. In the online format, the interviewer was unable to see the participants' facial expressions, unlike in the in-person format, making it more difficult to gauge when intervention and assistance were needed. The absence of visual cues forced the interviewer to rely solely on audio cues. Additionally, not being able to observe participants' offline activities was a limitation.

10.7 Future Work

In the future, the question classification framework could be improved through the examination of a larger data set that encompasses assessments other than just examinations. Further, including database modelling tasks from contemporary textbooks might also enhance the classification scheme. Moreover, the study did not investigate the relationship between learning outcomes and the alignment of exams with the corresponding learning outcomes of the courses. Further gaps in the assessments of database modelling courses may have been identified if this was examined. The presented question classification framework did not incorporate the revised Bloom's knowledge dimension since this was the first attempt to develop such a framework in database modelling. A possible avenue for improvement lies in incorporating the adaptation of Bloom's knowledge dimension proposed by Bogdanova and Snoeck (2019) in their analysis of OO analysis and design tasks. It is posited that their approach can be applied to database modelling assessment tasks, thus enriching the classification framework and providing a more comprehensive evaluation of students' cognitive levels and knowledge.

In the quantitative analysis, several database metrics were considered as potential predictors to measure the difficulty of novice database modelling tasks to develop a reliable predictor for task difficulty. However, this hypothesised predictor was not developed since the conducted statistical measures didn't hold any significant correlations between the task difficulty and the calculated metrics. This is worthwhile to consider further as such predictors have proven valuable in developing tasks in novice programming, which intuitively could be equally useful for educators in developing novice database modelling tasks. One of the exercises the students attempted in their natural coursework was identifying errors in an ERD. A potential area of interest is exploring whether reading activities prior to the creation process could enhance student performance. This thesis presents an initial approach to investigate the relationship between reading, understanding, and creating models. A hierarchy was not observed, possibly due to the limitations of the data. Thus, it would be worth undertaking this analysis again with better data. Intuitively, a hierarchy should exist, and if it does, it could help develop better learning tasks and assessments. If the difficulty of tasks can be predicted and tasks placed in a hierarchy of skills, then future researchers would have the tools to develop tasks to collect data for further in-depth qualitative studies of novice database modellers.

References

- Abid, A., Farooq, M. S., Raza, I., Farooq, U., & Abid, K. (2015). Variants of Teaching First Course in Database Systems. *Bulletin of Education & Research*, 37(2), 1–17.
- Abreu, P. H., Silva, D. C., & Gomes, A. (2019). Multiple-Choice Questions in Programming Courses: Can We Use Them and Are Students Motivated by Them? *ACM Transactions on Computing Education*, 19(1), 1–16. <https://doi.org/10.1145/3243137>
- ACM Data Science Task Force. (2021). Computing competencies for undergraduate data science curricula. In *Computing competencies for undergraduate data science curricula* (Issue January). ACM. <https://doi.org/10.1145/3453538>
- Adikari, S., McDonald, C., & Campbell, J. (2013). Reframed Contexts: Design Thinking for Agile User Experience Design. In A. Marcus (Ed.), *Design, User Experience, and Usability. Design Philosophy, Methods, and Tools*. (pp. 3–12). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-39229-0_1
- Aleven, V. A. W. M. M., & Koedinger, K. R. (2002). An effective metacognitive strategy: learning by doing and explaining with a computer-based Cognitive Tutor. *Cognitive Science*, 26(2), 147–179. https://doi.org/10.1207/s15516709cog2602_1
- Altman, D. G. (1999). *Practical statistics for medical research*. Chapman & Hall/CRC Press. <https://doi.org/10.1038/096483a0>
- Amin, M., Romney, G. W., Dey, P., & Sinha, B. (2019). Teaching Relational Database Normalization in an Innovative Way. *J. Comput. Sci. Coll.*, 35(2), 48–56.
- Amrit, C., & Tax, N. (2014). Towards understanding the understandability of UML models. *Proceedings of the 6th International Workshop on Modeling in Software Engineering - MiSE 2014*, 49–54. <https://doi.org/10.1145/2593770.2593779>
- Anderson, L. W., Krathwohl, D. R., Airasian, P. W., Cruikshank, K. A., Mayer, R. E., Pintrich, P. R., Raths, J., & Wittrock, M. C. (2001). *A Taxonomy for Learning, Teaching and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Addison-Wesley Longman, Inc. <https://www.uky.edu/~rsand1/china2018/texts/Anderson-Krathwohl - A taxonomy for learning teaching and assessing.pdf>
- Antony, S., & Mellarkod, V. (2004). A Methodology for using data-modeling patterns. *Symposium on*

- Aureliano, V. C. O., Tedesco, P. C. D. A. R., & Caspersen, M. E. (2016). Learning programming through stepwise self-explanations. *Iberian Conference on Information Systems and Technologies, CISTI, 2016-July*. <https://doi.org/10.1109/CISTI.2016.7521457>
- Badia, A., & Lemire, D. (2011). A call to arms: revisiting database design. *ACM SIGMOD Record*, 40(3), 61–69. <https://doi.org/10.1145/2070736.2070750>
- Banerjee, M., Capozzoli, M., McSweeney, L., & Sinha, D. (1999). Beyond kappa: A review of interrater agreement measures. *Canadian Journal of Statistics*, 27(1), 3–23. <https://doi.org/10.2307/3315487>
- Bastian, M., Heymann, S., & Jacomy, M. (2009). Gephi: An Open Source Software for Exploring and Manipulating Networks. *Proceedings of the International AAAI Conference on Web and Social Media*, 3(1), 361–362. <https://doi.org/10.1609/icwsm.v3i1.13937>
- Batra, D., & Antony, S. R. (1994). Novice errors in conceptual database design. *European Journal of Information Systems*, 3(1), 57–69. <https://doi.org/10.1057/ejis.1994.7>
- Batra, Dinesh. (2005). Cognitive complexity factors in data modeling. *Association for Information Systems - 11th Americas Conference on Information Systems, AMCIS 2005: A Conference on a Human Scale*, 7, 3460–3467.
- Batra, Dinesh. (2007). Cognitive complexity in data modeling: Causes and recommendations. *Requirements Engineering*, 12(4), 231–244. <https://doi.org/10.1007/s00766-006-0040-y>
- Batra, Dinesh, & Davis, J. G. (1992). Conceptual data modelling in database design: similarities and differences between expert and novice designers. *International Journal of Man-Machine Studies*, 37(1), 83–101. [https://doi.org/10.1016/0020-7373\(92\)90092-Y](https://doi.org/10.1016/0020-7373(92)90092-Y)
- Batra, Dinesh, & Sein, M. K. (1994). Improving conceptual database design through feedback. *International Journal of Human-Computer Studies*, 40(4), 653–676. <https://doi.org/10.1006/ijhc.1994.1030>
- Batra, Dinesh, & Wishart, N. A. (2004). Comparing a rule-based approach with a pattern-based approach at different levels of complexity of conceptual data modelling tasks. *International Journal of Human Computer Studies*, 61(4), 397–419. <https://doi.org/10.1016/j.ijhcs.2003.12.019>
- Becker, J., Niehaves, B., & Pfeiffer, D. (2008). Ontological Evaluation of Conceptual Models.

- Scandinavian Journal of Information Systems*, 20(2), 1–28. <http://aisel.aisnet.org/sjis/vol20/iss2/4>
- Berry, W., & Feldman, S. (1985). *Multiple Regression in Practice*. SAGE Publications, Inc. <https://doi.org/10.4135/9781412985208>
- Biggs, J. (1996). Enhancing teaching through constructive alignment. *Higher Education*, 32(3), 347–364. <https://doi.org/10.1007/BF00138871>
- Biggs, J. B., & Collis, K. F. (1982). *Evaluating the quality of learning : the SOLO taxonomy (Structure of the Observed Learning Outcome)*. Academic Press.
- Blondel, V. D., Guillaume, J.-L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10), P10008. <https://doi.org/10.1088/1742-5468/2008/10/P10008>
- Bloom, B. S., Engelhart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. (1956). *Taxonomy of educational objectives : The Classification of Educational Goals*. Longmans, Green and Co Ltd.
- Bock, D. B., & Ryan, T. (1996). Modeling ternary relationships. *Journal of Computer Information Systems*, 36(4), 60–65. <https://doi.org/https://doi.org/10.1080/08874417.1996.11647228>
- Bock, D. B., & Yager, S. E. (2005). Using the Data Modeling Worksheet to Improve Novice Data Modeler Performance. *Journal of Information Systems Education*, 16(3), 341–350.
- Bogdanova, D., & Snoeck, M. (2017). Domain modelling in bloom: Deciphering how we teach It. *Lecture Notes in Business Information Processing*, 305(November 2017), 3–17. https://doi.org/10.1007/978-3-319-70241-4_1
- Bogdanova, D., & Snoeck, M. (2019). CaMeLOT: An educational framework for conceptual data modelling. *Information and Software Technology*, 110(December 2017), 92–107. <https://doi.org/10.1016/j.infsof.2019.02.006>
- Borodzhieva, A. N. (2022). Using Project-Based Learning in the Course “Databases” on the Topic “Relational Algebra.” *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 1369–1374. <https://doi.org/10.23919/MIPRO55190.2022.9803494>
- Börstler, J., Caspersen, M. E., & Nordström, M. (2007). *Beauty and the Beast: Toward a Measurement Framework for Example Program Quality* (Issue January).

- Börstler, Jürgen, Caspersen, M. E., & Nordström, M. (2016). Beauty and the Beast: on the readability of object-oriented example programs. *Software Quality Journal*, 24(2), 231–246. <https://doi.org/10.1007/s11219-015-9267-5>
- Boyd, A., Lawson, T., & Guimaraes, M. (2005). Courseware to aid database design. *Proceedings of the 43rd Annual Southeast Regional Conference on - ACM-SE 43*, 1, 262. <https://doi.org/10.1145/1167350.1167426>
- Boyeena, M., & Goteti, P. (2010). Promoting active learning through case driven approach: An empirical study on database course. *Proceedings of the 2010 IEEE Students' Technology Symposium, April*, 191–195. <https://doi.org/10.1109/TECHSYM.2010.5469181>
- Branch, J. L. (2000). Investigating the Information-Seeking Processes of Adolescents. *Library & Information Science Research*, 22(4), 371–392. [https://doi.org/10.1016/S0740-8188\(00\)00051-7](https://doi.org/10.1016/S0740-8188(00)00051-7)
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77–101. <https://doi.org/10.1191/1478088706qp063oa>
- Braun, V., & Clarke, V. (2012). Thematic analysis. In H. Cooper, P. . Camic, L. Long, A. . Panter, D. Rindskopf, & K. J. Sher (Eds.), *APA handbook of research methods in psychology, Vol 2: Research designs: Quantitative, qualitative, neuropsychological, and biological*. (pp. 57–71). American Psychological Association. <https://doi.org/10.1037/13620-004>
- Braun, V., & Clarke, V. (2013). *Successful Qualitative Research A Practical Guide for Beginners* (1st ed.). Sage publications.
- Braun, V., & Clarke, V. (2019). Reflecting on reflexive thematic analysis. In *Qualitative Research in Sport, Exercise and Health* (Vol. 11, Issue 4, pp. 589–597). <https://doi.org/10.1080/2159676X.2019.1628806>
- Braun, V., & Clarke, V. (2021). Can I use TA? Should I use TA? Should I not use TA? Comparing reflexive thematic analysis and other pattern-based qualitative analytic approaches. *Counselling and Psychotherapy Research*, 21(1), 37–47. <https://doi.org/10.1002/capr.12360>
- Braun, V., & Clarke, V. (2022). *Thematic Analysis A Practical Guide* (A. Maher (ed.)). SAGE Publications Ltd.
- Braun, V., Clarke, V., Hayfield, N., & Terry, G. (2019). Thematic Analysis. In *Handbook of Research Methods in Health Social Sciences* (pp. 843–860). Springer Singapore. https://doi.org/10.1007/978-981-10-5251-4_103

- Brereton, P., Kitchenham, B. A., Budgen, D., Turner, M., & Khalil, M. (2007). Lessons from applying the systematic literature review process within the software engineering domain. *The Journal of Systems and Software*, 80(4), 571–583. <https://doi.org/10.1016/j.jss.2006.07.009>
- Briand, L. C., Wüst, J., & Lounis, H. (2001). Replicated Case Studies for Investigating Quality Factors in Object-Oriented Designs. *Empirical Software Engineering*, 6(1), 11–58. <https://doi.org/10.1023/A:1009815306478>
- Brown, S., & Klein, B. (2003). Small Steps Using Cooperative Learning Techniques in the Database Management Course: Some Preliminary Results. *AMCIS 2003 Proceedings*, Paper 396. <http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1862&context=amcis2003>
- Bunch, J. M. (2009). An Approach to Reducing Cognitive Load in the Teaching of Introductory Database Concepts. *Journal of Information Systems Education*, 20(3), 269–275. <https://aisel.aisnet.org/jise/vol20/iss3/1>
- Byrne, D. (2022). A worked example of Braun and Clarke’s approach to reflexive thematic analysis. In *Quality and Quantity* (Vol. 56, Issue 3, pp. 1391–1412). <https://doi.org/10.1007/s11135-021-01182-y>
- Calero, C., Piattini, M., & Genero, M. (2001a). Database Complexity Metrics. *Proc. 4th International Conference on the Quality of Information and Communications Technology*, 79–85.
- Calero, C., Piattini, M., & Genero, M. (2001b). Empirical validation of referential integrity metrics. *Information and Software Technology*, 43(15), 949–957. [https://doi.org/10.1016/S0950-5849\(01\)00202-6](https://doi.org/10.1016/S0950-5849(01)00202-6)
- Cardell-Oliver, R. (2011). How Can Software Metrics Help Novice Programmers? *Proceedings of the Thirteenth Australasian Computing Education Conference - Volume 114*, 55–62. <https://doi.org/10.5555/2459936.2459943>
- Castro, F. E. V., & Fisler, K. (2017). Designing a multi-faceted SOLO taxonomy to track program design skills through an entire course. *ACM International Conference Proceeding Series*, 10–19. <https://doi.org/10.1145/3141880.3141891>
- Catania, B., Guerrini, G., & Traversaro, D. (2022). Collaborative Learning in an Introductory Database Course: A Study with Think-Pair-Share and Team Peer Review. *1st International Workshop on Data Systems Education*, 60–66. <https://doi.org/10.1145/3531072.3535330>
- Catrambone, R. (1998). The subgoal learning model: Creating better examples so that students can solve

- novel problems. *Journal of Experimental Psychology: General*, 127(4), 355–376. <https://doi.org/10.1037/0096-3445.127.4.355>
- CC2020 Task Force. (2020). *Computing Curricula 2020: Paradigms for Global Computing Education*. Association for Computing Machinery. <https://doi.org/10.1145/3467967>
- Chandler, P., & Sweller, J. (1992). The split-attention effect as a factor in the design of instruction. *British Journal of Educational Psychology*, 62(2), 233–246. <https://doi.org/10.1111/j.2044-8279.1992.tb01017.x>
- Chen, Z., Li, N., & Zhang, H. (2012). Exploration of teaching model of the database course based on constructivism learning theory. *2012 2nd International Conference on Consumer Electronics, Communications and Networks, CECNet 2012 - Proceedings*, 1808–1811. <https://doi.org/10.1109/CECNet.2012.6201855>
- Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-Explanations: How Students Study and Use Examples in Learning to Solve Problems. *Cognitive Science*, 13(2), 145–182. https://doi.org/10.1207/s15516709cog1302_1
- Chilton, M. A. (2006). Data Modeling Using Entity Relationship Diagrams: A Step-Wise Method. *Journal of Information Systems Education*, 17(4), 385–394.
- Clear, T., Bouvier, D. J., Jacková, J., Robbins, P., Lister, R., Carter, P., Lopez, M., Seppälä, O., Simon, Eckerdal, A., McCartney, R., & Thompson, E. (2009). Naturally occurring data as research instrument: Analyzing examination responses to study the novice programmer. *SIGCSE Bulletin Inroads*, 41(4), 156–173. <https://doi.org/10.1145/1709424.1709460>
- Collins, A. (2006). Cognitive Apprenticeship. In *The Cambridge handbook of: The learning sciences*. (pp. 47–60). Cambridge University Press.
- Conati, C. (2016). Commentary on: “Toward Computer-Based Support of MetaCognitive Skills: a Computational Framework to Coach Self Explanation.” *International Journal of Artificial Intelligence in Education*, 26(1), 183–192. <https://doi.org/10.1007/s40593-015-0074-8>
- Connolly, T., & Begg, C. (2006). A constructivist-based approach to teaching database analysis and design. *Journal of Information Systems Education*, 17(1), 43.
- Copple, C., & Bredekamp, S. (2009). *Developmentally appropriate practice in early childhood programs serving children from birth through age 8* (3rd ed.). National Association for the Education of Young Children. <https://www.naeyc.org/sites/default/files/globally->

shared/downloads/PDFs/resources/position-statements/PSDAP.pdf

- Coronel, C., & Morris, S. (2016). *Database systems : design, implementation, and management* (12th ed.). Cengage Learning.
- Costantini, U., Lonati, V., & Morpurgo, A. (2020). How Plans Occur in Novices' Programs. *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 852–858. <https://doi.org/10.1145/3328778.3366870>
- Creswell, W. J. (2009). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (3rd ed.). SAGE Publications Inc.
- Culligan, N., & Casey, K. (2020). Exploring the Coding Behaviour of Successful Students in Programming by Employing Neo-Piagetian Theory. *Proceedings of the 31st Annual Workshop of the Psychology of Programming Interest Group (PPIG)*, 57–65.
- Currim, S., Ram, S., Durcikova, A., & Currim, F. (2014). Using a knowledge learning framework to predict errors in database design. *Information Systems*, 40, 11–31. <https://doi.org/10.1016/j.is.2013.08.001>
- Dann, W., Cosgrove, D., Slater, D., Culyba, D., & Cooper, S. (2012). Mediated transfer : Alice 3 to Java. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, 141–146. <https://doi.org/10.1145/2157136.2157180>
- Davies, M., & Fleiss, J. L. (1982). Measuring Agreement for Multinomial Data. *Biometrics*, 38(4), 1047. <https://doi.org/10.2307/2529886>
- Descript. (2021). *Descript* (20.1.4). <https://www.descript.com/>
- Dhillon, M. K., & Dasgupta, S. (2011). Individual Differences and Conceptual Modeling Task Performance: Examining the Effects of Cognitive Style, Self-efficacy, and Application Domain Knowledge. In T. Halpin, N. Selmin, J. Krogstie, P. Soffer, E. Proper, R. Schmidt, & B. Ilia (Eds.), *Enterprise, Business-Process and Information Systems Modeling, Lecture Notes in Business Information Processing*. (Vol. 81, pp. 483–496). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-21759-3_35
- Dietrich, S. W., & Goelman, D. (2017). Formative self-assessment for customizable database visualizations: Checkpoints for learning. *ASEE Annual Conference and Exposition, Conference Proceedings, 2017-June*. <https://doi.org/10.18260/1-2--27853>

- Dietrich, S. W., & Urban, S. D. (1996). Database theory in practice. *ACM SIGCSE Bulletin*, 28(1), 112–116. <https://doi.org/10.1145/236462.236520>
- Drake, J. M. (1998). Class to survey application programming topics. *SIGCSE Bulletin (Association for Computing Machinery, Special Interest Group on Computer Science Education)*, 30(1), 161–165. <https://doi.org/10.1145/274790.273183>
- Duran, R., Sorva, J., & Leite, S. (2018). Towards an analysis of program complexity from a cognitive perspective. *ICER 2018 - Proceedings of the 2018 ACM Conference on International Computing Education Research*, 21–30. <https://doi.org/10.1145/3230977.3230986>
- Duran, R., Zavgorodniaia, A., & Sorva, J. (2022). Cognitive Load Theory in Computing Education Research: A Review. *ACM Transactions on Computing Education*, 22(4), 1–27. <https://doi.org/10.1145/3483843>
- Elnaffar, S. (2016). Using software metrics to predict the difficulty of code writing questions. *2016 IEEE Global Engineering Education Conference (EDUCON), April*, 513–518. <https://doi.org/10.1109/EDUCON.2016.7474601>
- Ericsson, K. A. (2017). Protocol Analysis. In W. Bechtel & G. Graham (Eds.), *A Companion to Cognitive Science* (pp. 425–432). Blackwell Publishing Ltd. <https://doi.org/10.1002/9781405164535.ch33>
- Ericsson, K. A., & Simon, H. A. (1993). *Protocol Analysis: Verbal Reports as Data*. The MIT Press. <https://doi.org/10.7551/mitpress/5657.001.0001>
- Etlinger, H. A. (2013). Adding a contributing student pedagogy component to an introductory database course. *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*, 299–304. <https://doi.org/10.1145/2445196.2445288>
- Fang, A.-D., Chen, G.-L., Cai, Z.-R., Cui, L., & Harn, L. (2017). Research on Blending Learning Flipped Class Model in Colleges and Universities Based on Computational Thinking — “Database Principles” for Example. *EURASIA Journal of Mathematics, Science and Technology Education*, 13(8), 5747–5755. <https://doi.org/10.12973/eurasia.2017.01024a>
- Festinger, L. (1957). A theory of cognitive dissonance. In *A theory of cognitive dissonance*. Stanford University Press.
- Fincher, S., & Petre, M. (2004). *Computer Science Education Research*. Taylor & Francis.

- Flesch, R. (1948). A new readability yardstick. In *Journal of Applied Psychology* (Vol. 32, Issue 3, pp. 221–233). American Psychological Association. <https://doi.org/10.1037/h0057532>
- Fredborg, L. (2021). *DocTools ExtractData* (1.5). <https://doi.org/https://wordaddins.com/products/extract-acronyms-bookmarks-tracked-changes-and-comments-from-word/>
- Fuller, R. M., Murthy, U., & Schafer, B. A. (2010). The effects of data model representation method on task performance. *Information and Management*, 47(4), 208–218. <https://doi.org/10.1016/j.im.2009.06.008>
- Fuller, U., Johnson, C. G., Ahoniemi, T., Cukierman, D., Hernán-Losada, I., Jackova, J., Lahtinen, E., Lewis, T. L., Thompson, D. M. G., Riedesel, C., & Thompson, E. (2007). Developing a computer science-specific learning taxonomy. *ITiCSE-WGR 2007 - Working Group Reports on ITiCSE on Innovation and Technology in Computer Science Education*, 152–170. <https://doi.org/10.1145/1345443.1345438>
- Gabrysiak, G., Giese, H., & Seibel, A. (2011). Towards Next Generation Design Thinking: Scenario-Based Prototyping for Designing Complex Software Systems with Multiple Users. In C. Meinel, L. Leifer, & H. Plattner (Eds.), *Design Thinking: Understand -- Improve -- Apply* (pp. 219–236). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-13757-0_13
- Gagne, R. M., Briggs, L. J., & Wager, W. W. (1992). *Principles of Instructional Design* (Fourth). Harcourt Brace College Publishers. <https://doi.org/10.1525/9780520341302-010>
- Gauvain, M., & Richert, R. (2023). Cognitive development. In H. S. Friedman & C. H. Markey (Eds.), *Encyclopedia of Mental Health* (3rd ed., pp. 430–437). Elsevier. <https://doi.org/10.1016/B978-0-323-91497-0.00048-5>
- Genero, M., Poels, G., & Piattini, M. (2002). Defining and validating measures for conceptual data model quality. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2348(October 2013), 724–727. https://doi.org/10.1007/3-540-47961-9_54
- Genero, M., Poels, G., & Piattini, M. (2008). Defining and validating metrics for assessing the understandability of entity-relationship diagrams. *Data and Knowledge Engineering*, 64(3), 534–557. <https://doi.org/10.1016/j.datak.2007.09.011>
- Ginat, D., & Menashe, E. (2015). SOLO taxonomy for assessing novices' algorithmic design. *SIGCSE*

- 2015 - *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, 452–457. <https://doi.org/10.1145/2676723.2677311>
- Ginat, D., Shifroni, E., & Menashe, E. (2011). Transfer, Cognitive Load, and Program Design Difficulties. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 7013 LNCS* (pp. 165–176). https://doi.org/10.1007/978-3-642-24722-4_15
- Glaser, B. G., & Strauss, A. L. (1968). The Discovery of Grounded Theory: Strategies for Qualitative Research. *American Journal of Sociology*, 73(6), 773–774. <https://doi.org/10.1086/224572>
- Glassman, E. L. (2014). Interacting with massive numbers of student solutions. *Proceedings of the Adjunct Publication of the 27th Annual ACM Symposium on User Interface Software and Technology - UIST'14 Adjunct*, 17–20. <https://doi.org/10.1145/2658779.2661167>
- Gluga, R., Kay, J., Lister, R., Kleitman, S., & Lever, T. (2012). Over-confidence and confusion in using bloom for programming fundamentals assessment. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, 147–152. <https://doi.org/10.1145/2157136.2157181>
- Goelman, D., & Dietrich, S. W. (2018). A Visual Introduction to Conceptual Database Design for All. *Proceedings of the 49th ACM Technical Symposium on Computer Science Education, 2018-Janua*, 320–325. <https://doi.org/10.1145/3159450.3159555>
- Goldweber, M., Wei, M., Aly, S., Raj, R. K., & Mokbel, M. (2022). The 2022 undergraduate database course in computer science: What to teach? *ACM Inroads*, 13(3), 16–21. <https://doi.org/10.1145/3549545>
- Granda, R. X., Echeverria, V., Chiluiza, K., & Wong-Villacres, M. (2015). Supporting the Assessment of Collaborative Design Activities in Multi-tabletop Classrooms. *2015 Asia-Pacific Conference on Computer Aided System Engineering*, 270–275. <https://doi.org/10.1109/APCASE.2015.54>
- Gray, L., Wong-Wylie, G., Rempel, G., & Cook, K. (2020). Expanding Qualitative Research Interviewing Strategies: Zoom Video Communications. *The Qualitative Report*, 25(5), 1292–1301. <https://doi.org/10.46743/2160-3715/2020.4212>
- Gu, L., Huanf, R., & Marton, F. (2004). Teaching with Variation: A Chinese Way of Promoting Effective Mathematics Learning. In *How Chinese learn mathematics: Perspectives from insiders* (pp. 309–347). World Sceientific Publishing Co. Pte. Ltd. https://doi.org/10.1142/9789812562241_0012

- Gunning, R. (1952). *The technique of clear writing*. McGraw-Hill.
- Hall, L., & Gordon, A. (1998). A virtual learning environment for entity relationship modelling. *SIGCSE Bulletin (Association for Computing Machinery, Special Interest Group on Computer Science Education)*, 30(1), 345–349. <https://doi.org/10.1145/274790.274327>
- Hamzah, M. L., Rukun, K., Rizal, F., Purwati, A. A., Hamzah, & Zarnelly. (2019). A review of increasing teaching and learning database subjects in computer science. *Espacios*, 40(26).
- Harris, L. R., Adie, L., & Wyatt-Smith, C. (2022). Learning Progression–Based Assessments: A Systematic Review of Student and Teacher Uses. *Review of Educational Research*, 92(6), 996–1040. <https://doi.org/10.3102/00346543221081552>
- Harrison, G. (2004). Computer-Based Assessment Strategies in the Teaching of Databases at Honours Degree Level 1. In H. Williams & L. MacKinnon (Eds.), *Key Technologies for Data Management* (pp. 257–264). Springer. https://doi.org/10.1007/978-3-540-27811-5_23
- Hazzan, O., Dubinsky, Y., Eidelman, L., Sakhnini, V., & Teif, M. (2006). Qualitative research in computer science education. *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education - SIGCSE '06*, 408. <https://doi.org/10.1145/1121341.1121469>
- Higgins, C. A., & Bligh, B. (2006). Formative computer based assessment in diagram based domains. *Proceedings of the 11th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, 38(3), 98. <https://doi.org/10.1145/1140124.1140152>
- Hoffer, J., Venkataraman, R., & Topi, H. (2016). *Modern Database Management* (12th ed.). Pearson Higer Educatio.
- Hogarty, K. Y., Hines, C. V., Kromrey, J. D., Perron, J. M., & Mumford, A. K. R. (2005). The quality of factor solutions in exploratory factor analysis: The influence of sample size, communality, and overdetermination. *Educational and Psychological Measurement*, 65(2), 202–226. <https://doi.org/10.1177/0013164404267287>
- Holden, E. P. (2008). Assessment of an introductory database course: A case study. *SIGITE'08: Proceedings of the 9th ACM SIG-Information Technology Education Conference*, 131–138. <https://doi.org/10.1145/1414558.1414595>
- Holmes, A. G. D. (2020). Researcher positionality: a consideration of its influence and place in qualitative research. *Shanlax International Journal of Education*, 8(4), 1–10.

- Hussain, T. (2017). Teaching entity-relationship models effectively. *Proceedings - 2016 International Conference on Computational Science and Computational Intelligence, CSCI 2016*, 264–269. <https://doi.org/10.1109/CSCI.2016.0058>
- Hussain, T. (2016). Teaching Entity-Relationship Models Effectively. *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, 264–269. <https://doi.org/10.1109/CSCI.2016.0058>
- IBM Corp. (2021). *IBM SPSS Statistics for Windows* (Version 28.0). IBM Corp.
- Imbulpitiya, A., Whalley, J., & Senapathi, M. (2021). Examining the Exams: Bloom and Database Modelling and Design. *Proceedings of the Twenty-Third Australasian Conference on Computing Education*, 21–29. <https://doi.org/10.1145/3441636.3442301>
- Ishaq, M., Abid, A., Farooq, M. S., Manzoor, M. F., Farooq, U., Abid, K., & Helou, M. A. (2022). Advances in database systems education: Methods, tools, curricula, and way forward. *Education and Information Technologies*. <https://doi.org/10.1007/s10639-022-11293-0>
- Ishikawa, K. (1976). *Guide to quality control*. Asian Productivity Organization.
- Izu, C., & Mirolo, C. (2021). Learning Transfer in Novice Programmers: A Preliminary Study. *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1, I(1)*, 178–184. <https://doi.org/10.1145/3430665.3456336>
- Izu, C., Schulte, C., Aggarwal, A., Cutts, Q., Duran, R., Gutica, M., Heinemann, B., Kraemer, E., Lonati, V., Mirolo, C., & Weeda, R. (2019a). Fostering program comprehension in novice programmers - learning activities and learning trajectories. *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, 27–52. <https://doi.org/10.1145/3344429.3372501>
- Izu, C., Schulte, C., Aggarwal, A., Cutts, Q., Duran, R., Gutica, M., Heinemann, B., Kraemer, E., Lonati, V., Mirolo, C., & Weeda, R. (2019b). Program Comprehension: Identifying Learning Trajectories for Novice Programmers. *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education, February 2020*, 261–262. <https://doi.org/10.1145/3304221.3325531>
- Izu, C., Weerasinghe, A., & Pope, C. (2016). A Study of Code Design Skills in Novice Programmers using the SOLO taxonomy. *Proceedings of the 2016 ACM Conference on International Computing Education Research*, 251–259. <https://doi.org/10.1145/2960310.2960324>

- Jenks, C. J. (2011). *Transcribing talk and interaction : issues in the representation of communication data*. John Benjamins Pub. Co. <http://site.ebrary.com/id/10495929>
- Jiang, B., Zhao, W., Zhang, N., & Qiu, F. (2019). Programming trajectories analytics in block-based programming language learning. *Interactive Learning Environments*, 0(0), 1–14. <https://doi.org/10.1080/10494820.2019.1643741>
- Johnson, B. E. (2011). The speed and accuracy of voice recognition software-assisted transcription versus the listen-and-type method: a research note. *Qualitative Research*, 11(1), 91–97. <https://doi.org/10.1177/1468794110385966>
- Joint Task Force on Computing Curricula, A. for C. M. (ACM) and I. C. S. (2013). *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science* (ACM Computing Curricula Task Force (ed.)). Association for Computing Machinery. <https://doi.org/10.1145/2534860>
- Jordan, B., & Henderson, A. (1995). Interaction Analysis: Foundations and Practice. *Journal of the Learning Sciences*, 4(1), 39–103. https://doi.org/10.1207/s15327809jls0401_2
- Kalyuga, S. (2011a). *Cognitive Load Theory : How Many Types of Load Does It Really Need ?* 1–19. <https://doi.org/https://doi.org/10.1007/s10648-010-9150-7>
- Kalyuga, S. (2011b). Cognitive load theory: Implications for affective computing. *Proceedings of the 24th International Florida Artificial Intelligence Research Society, FLAIRS - 24*, 105–110.
- Kalyuga, S., Chandler, P., & Sweller, J. (1999). Managing split-attention and redundancy in multimedia instruction. *Applied Cognitive Psychology*, 13(4), 351–371. [https://doi.org/10.1002/\(SICI\)1099-0720\(199908\)13:4<351::AID-ACP589>3.0.CO;2-6](https://doi.org/10.1002/(SICI)1099-0720(199908)13:4<351::AID-ACP589>3.0.CO;2-6)
- Kasto, N. (2016). *Learning to Program: The Development of Knowledge in Novice Programmers* [Auckland University of Technology]. <http://hdl.handle.net/10292/10377>
- Kasto, N., & Whalley, J. (2013). Measuring the difficulty of code comprehension tasks using software metrics. *Conferences in Research and Practice in Information Technology Series*, 136(59), 59–65.
- Kasto, N., Whalley, J., Philpott, A., & Whalley, D. (2014). Solution Spaces. *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148*, 133–137.
- Katz, A. (2018a). Visualizing the syntax of gradual transitions in teaching database modeling.

Proceedings of the 11th IADIS International Conference Information Systems 2018, IS 2018, April, 228–232.

Katz, A. (2018b). Applying TSSL in database schema modeling: Visualizing the syntax of gradual transitions. *Journal of Technology and Science Education*, 8(4), 238. <https://doi.org/10.3926/jotse.365>

Katz, A. (2020). Improved Teaching of Database Schema Modeling by Visualizing Changes in Levels of Abstraction. *Journal of Information Systems Education*, 31(4), 294–311. <http://jise.org/Volume31/n4/JISEv31n4p294.html>

Katz, A., & Shmallo, R. (2015). Improving relational data modelling through learning from errors. *Proceedings of the European Conference on Data Mining 2015, ECDM 2015 and International Conferences on Intelligent Systems and Agents 2015, ISA 2015 and Theory and Practice in Modern Computing 2015, TPMC 2015 - Part of the Multi Conference on Computer Sci, 1972*, 198–202.

Katz, A., & Shmallo, R. (2016). Learning from errors as a pedagogic approach for reaching a higher conceptual level in database modeling. *Lecture Notes in Business Information Processing*, 249, 93–102. https://doi.org/10.1007/978-3-319-39564-7_9

Kaur, H., & Chopra, V. (2010). Creation of web based tutor to enhance student learning of normalization. *ICIME 2010 - 2010 2nd IEEE International Conference on Information Management and Engineering*, 4, 182–185. <https://doi.org/10.1109/ICIME.2010.5477527>

Kawash, J., Jarada, T., & Moshirpour, M. (2020). Group Exams as Learning Tools. *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 626–632. <https://doi.org/10.1145/3328778.3366889>

Kawash, J., & Sailunaz, K. (2020). Learning By Creating Instructional Videos: An Experience Report from a Database Course. *2020 IEEE Global Engineering Education Conference (EDUCON)*, 1536–1541. <https://doi.org/10.1109/EDUCON45650.2020.9125218>

Kellam, N. N., Gerow, K. S., & Walther, J. (2015). Narrative analysis in engineering education research: Exploring ways of constructing narratives to have resonance with the reader and critical research implications. *ASEE Annual Conference and Exposition, Conference Proceedings, 122nd ASEE (122nd ASEE Annual Conference and Exposition: Making Value for Society)*. <https://doi.org/10.18260/p.24521>

- Kelly, N., & Gero, J. S. (2021). Design thinking and computational thinking: A dual process model for addressing design problems. *Design Science, Ennis 1993*, 1–15. <https://doi.org/10.1017/dsj.2021.7>
- Kesh, S. (1995). Evaluating the quality of entity relationship models. *Information and Software Technology*, 37(12), 681–689. [https://doi.org/10.1016/0950-5849\(96\)81745-9](https://doi.org/10.1016/0950-5849(96)81745-9)
- Khomokhoana, P. J. (2022). Mapping the Problem-Solving Strategies of Novice Programmers to Polya's Framework : SWOT Analysis as a Bottleneck Identification Tool. In *Communications in Computer and Information Science* (pp. 132–148). Springer International Publishing. <https://doi.org/10.1007/978-3-030-95003-3>
- Kim, N., Lee, D., & Moon, S. (2007). Behavior-inductive data modeling for enterprise information systems. *Journal of Computer Information Systems*, 48(1), 105–116. <https://doi.org/10.1080/08874417.2007.11645999>
- Kintsch, W., & van Dijk, T. A. (1978). Toward a model of text comprehension and production. *Psychological Review*, 85(5), 363–394. <https://doi.org/10.1037/0033-295X.85.5.363>
- Knorr, E. M. (2020). Worked examples, cognitive load, and exam assessments in a senior database course. *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, 612–618. <https://doi.org/10.1145/3328778.3366915>
- Kouznetsova, S. (2009). No more freeloading: using individual assignments to improve team-based learning outcomes. *Journal of Computing Sciences in Colleges*, 24(5), 30–36.
- Kui, X., Du, H., Zhong, P., & Liu, W. (2018). Research and Application of Flipped Classroom in Database Course. *2018 13th International Conference on Computer Science & Education (ICCSE)*, 1–5. <https://doi.org/10.1109/ICCSE.2018.8468744>
- Kung, H.-J., & Kung, L. (2013). An Interactive Tool to Improve Learning of Data Modeling: A Survey Study. *J. Comput. Sci. Coll.*, 28(4), 11–18. <https://doi.org/10.5555/2458539.2458541>
- Kwan, A. C. M. (2013). Towards a minimalist approach to lesson planning of an introductory database course. *Proceedings of 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering, TALE 2013, August*, 782–785. <https://doi.org/10.1109/TALE.2013.6654544>
- Landis, J. R., & Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data Published by : International Biometric Society Stable URL : <http://www.jstor.org/stable/2529310>.

Biometrics, 33(1), 159–174.

- Lee, A., & Chan, H. (2001). An Exploratory Analysis of Semantic Network Complexity for Data Modeling Performance. *PACIS 2001 Proceedings*, 702–716. <https://aisel.aisnet.org/pacis2001/49>
- Leinonen, J., Denny, P., & Whalley, J. (2021). Exploring the Effects of Contextualized Problem Descriptions on Problem Solving. *ACM International Conference Proceeding Series*, 30–39. <https://doi.org/10.1145/3441636.3442302>
- Lim, B. B. L., & Hunter, R. (1992). DBTool: A graphical database tool for an introductory database course. *ACM SIGCSE Bulletin*, 24(1), 24–27. <https://doi.org/10.1145/135250.134516>
- Lin, M.-H., Chen, M.-P., & Chen, C.-F. (2015). Effects of Question Prompts and Self-explanation on Database Problem Solving in a Peer Tutoring Context. In N. Nguyen, B. Trawinski, & R. Kosala (Eds.), *Intelligent Information and Database Systems: Lecture Notes in Computer Science* (Vol. 9012, pp. 180–189). Springer, Cham. https://doi.org/10.1007/978-3-319-15705-4_18
- Lin, M. H., Chen, M. P., & Chen, C. F. (2013). Exploring peer scaffolding opportunities on experiential problem solving learning. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8083 LNAI, 572–581. https://doi.org/10.1007/978-3-642-40495-5_57
- Lina, W. (2018). Teaching Design of “Access Database Technology and Application” Based on Flipped Classroom. *2nd International Conference on Data Science and Business Analytics (ICDSBA)*, 251–254. <https://doi.org/10.1109/ICDSBA.2018.00052>
- Lister, R. (2001). Objectives and objective assessment in CS1. *ACM SIGCSE Bulletin*, 33(1), 292–296. <https://doi.org/10.1145/366413.364605>
- Lister, R. (2000). On blooming first year programming, and its blooming assessment. *Proceedings of the Australasian Conference on Computing Education - ACSE '00*, 158–162. <https://doi.org/10.1145/359369.359393>
- Lister, R., Adams, E. S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M., McCartney, R., Moström, J. E., Sanders, K., & Seppälä, O. (2007). A Multi-National Study of Reading and Tracing Skills in Novice Programmers. *SIGCSE Bulletin*, 36(4), 119–150. <https://doi.org/10.1145/1044550.1041673>
- Livescribe Inc. (2015a). *Echo Desktop* (Version 3.0). [mobile app]. <http://www.livescribe.com/en-gb/media/pdf/support/EchoDesktopUserGuide.pdf>

- Livescribe Inc. (2015b). *Livescribe™ Echo 2 smartpen*. <https://us.livescribe.com/products/echo-2™-smartpen>
- Lo Iacono, V., Symonds, P., & Brown, D. H. K. (2016). Skype as a Tool for Qualitative Research Interviews. *Sociological Research Online*, 21(2), 103–117. <https://doi.org/10.5153/sro.3952>
- Lobato, J., & Walters, D. C. (2017). A Taxonomy of Approaches to Learning Trajectories and Progressions. In J. Cai (Ed.), *Compendium for Research in Mathematics Education* (pp. 74–101). National Council of Teachers of Mathematics.
- Loksa, D., & Ko, A. J. (2016). The Role of Self-Regulation in Programming Problem Solving Process and Success. *Proceedings of the 2016 ACM Conference on International Computing Education Research*, 83–91. <https://doi.org/10.1145/2960310.2960334>
- Long, H. B., McCrary, K., & Ackerman, S. (1979). Adult Cognition: Piagetian Based Research Findings. *Adult Education*, 30(1), 3–18. <https://doi.org/10.1177/074171367903000101>
- Lopez, M., Whalley, J., Robbins, P., & Lister, R. (2008). Relationships between reading, tracing and writing skills in introductory programming. *ICER'08 - Proceedings of the ACM Workshop on International Computing Education Research*, 101–111. <https://doi.org/10.1145/1404520.1404531>
- Lorenzo-Seva, U., & Ferrando, P. J. (2021). Not Positive Definite Correlation Matrices in Exploratory Item Factor Analysis: Causes, Consequences and a Proposed Solution. *Structural Equation Modeling*, 28(1), 138–147. <https://doi.org/10.1080/10705511.2020.1735393>
- Luxton-Reilly, A., Becker, B. A., Cao, Y., McDermott, R., Mirolo, C., Mühling, A., Petersen, A., Sanders, K., Simon, & Whalley, J. (2018). Developing Assessments to Determine Mastery of Programming Fundamentals. *Proceedings of the 2017 ITiCSE Conference on Working Group Reports, 2018-Janua*, 47–69. <https://doi.org/10.1145/3174781.3174784>
- Luxton-Reilly, A., Denny, P., Kirk, D., Tempero, E., & Yu, S.-Y. (2013). On the differences between correct student solutions. *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE '13*, 177. <https://doi.org/10.1145/2462476.2462505>
- MacLean, L. M., Meyer, M., & Estable, A. (2004). Improving accuracy of transcripts in qualitative research. *Qualitative Health Research*, 14(1), 113–123. <https://doi.org/10.1177/1049732303259804>
- Maiorana, F. (2015). Extending the database curriculum: From design principles to web and mobile

- programming. *Communications in Computer and Information Science*, 510, 258–271.
https://doi.org/10.1007/978-3-319-25768-6_17
- Margulieux, L. E., Guzdial, M., & Catrambone, R. (2012). Subgoal-labeled instructional material improves performance and transfer in learning to develop mobile applications. *Proceedings of the Ninth Annual International Conference on International Computing Education Research*, 71–78.
<https://doi.org/10.1145/2361276.2361291>
- Marjanovic, O. (2013). Organizational design of innovative education - Insights from a combined design and action research project. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7939 LNCS, 212–227. https://doi.org/10.1007/978-3-642-38827-9_15
- Marjanovic, O. (2016). Designing Innovative Education through Action Design Research: Method and Application for Teaching Design Activities in Large Lecture Environments. *Journal of Information Technology Theory and Application JITTA*, 17(1), 1.
- Martínez-González, M. M., & Duffing, G. (2007). Teaching databases in compliance with the European dimension of higher education: Best practices for better competences. *Education and Information Technologies*, 12(4), 211–228. <https://doi.org/10.1007/s10639-007-9047-3>
- Martinez, A. (2012). Using JITT in a database course. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education - SIGCSE '12*, 367–372.
<https://doi.org/10.1145/2157136.2157245>
- Masapanta-Carrión, S., & Velázquez-Iturbide, J. Á. (2018). A Systematic Review of the Use of Bloom's Taxonomy in Computer Science Education. *Proceedings of the 49th ACM Technical Symposium on Computer Science Education, 2018-Janua*, 441–446.
<https://doi.org/10.1145/3159450.3159491>
- Mason, R., & Seton, C. (2020). Assessing International Students - The Role of Cognitive Load. *Proceedings of the Twenty-Second Australasian Computing Education Conference*, 53(9), 160–166. <https://doi.org/10.1145/3373165.3373183>
- Mason, R., & Seton, C. (2021). Leveling the playing field for international students in IT courses. *Proceedings of the Australasian Computing Education Conference*, 138–146.
<https://doi.org/10.1145/3441636.3442316>
- Mason, R., Seton, C., & Cooper, G. (2016). Applying cognitive load theory to the redesign of a

- conventional database systems course. *Computer Science Education*, 26(1), 68–87.
<https://doi.org/10.1080/08993408.2016.1160597>
- Matusov, E., & Hayes, R. (2000). Sociocultural critique of Piaget and Vygotsky. *New Ideas in Psychology*, 18(2–3), 215–239. [https://doi.org/10.1016/S0732-118X\(00\)00009-X](https://doi.org/10.1016/S0732-118X(00)00009-X)
- McCallum, A. K. (1996). Learning to Use Selective Attention and Short-Term Memory in Sequential Tasks. *Proceedings of 4th International Conference in Simulation of Adaptive Behavior*, 315–324.
<https://doi.org/10.7551/mitpress/3118.003.0039>
- McLaughlin, G. H. (1969). SMOG grading: A new readability formula. *Journal of Reading*, 12(8), 639–646.
- McLoughlin, C. (1999). Scaffolding: Applications to learning in technology supported environments. In B. Collis & R. Oliver (Eds.), *Proceedings of ED-MEDIA 1999--World Conference on Educational Multimedia, Hypermedia & Telecommunications* (pp. 1826–1831). Association for the Advancement of Computing in Education (AACE).
<https://www.learntechlib.org/primary/p/7213/>
- Meng, F., & Chen, W. (2021). Exploration of Mixed Teaching Based on BOPPPS Taking Database Courses as an Example. *2021 2nd International Conference on Artificial Intelligence and Education (ICAIE)*, 349–352. <https://doi.org/10.1109/ICAIE53562.2021.00079>
- Mengel, S. A., & Yerramilli, V. (1999). A case study of the static analysis of the quality of novice student programs. *SIGCSE 1999 - Proceedings of the 13th SIGCSE Technical Symposium on Computer Science Education*, 78–82. <https://doi.org/10.1145/299649.299689>
- Milik, N., Marshall, M., & Mitrovic, A. (2006a). Responding to Free-Form Student Questions in ERM-Tutor. In M. Ikeda, K. . Ashley, & T.-W. Chan (Eds.), *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 4053 LNCS* (pp. 707–709). Springer, Berlin, Heidelberg.
https://doi.org/10.1007/11774303_75
- Milik, N., Marshall, M., & Mitrovic, A. (2006b). Teaching logical database design in ERM-Tutor. *Proceedings of ITS 2006*, 707–709.
- Milik, N., Mitrovic, A., & Grimley, M. (2008). Investigating the Relationship between Spatial Ability and Feedback Style in ITSs. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5091 LNCS, 281–290.

https://doi.org/10.1007/978-3-540-69132-7_32

- Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63(2), 81–97. <https://doi.org/10.1037/h0043158>
- Mishra, S., Aisuryalaxmi, E., & Mall, R. (2012). Estimating Database Size and Its Development Effort at Conceptual Design Stage. In *Communications in Computer and Information Science: Vol. 270 CCIS* (Issue PART II, pp. 120–127). https://doi.org/10.1007/978-3-642-29216-3_14
- Mitchell, A. (2020). Four Heads Are Better Than One: Using Team-Based Learning to Learn Database Technology. *SIGITE 2020 - Proceedings of the 21st Annual Conference on Information Technology Education*, 384–389. <https://doi.org/10.1145/3368308.3415359>
- Mithun, S., & Luo, X. (2019). Incorporate Cross-Course Knowledge Integration into Computing Education. *Proceedings - Frontiers in Education Conference, FIE, 2019-Octob.* <https://doi.org/10.1109/FIE43999.2019.9028588>
- Mitrovic, A. (2002). NORMIT: a Web-enabled tutor for database normalization. *International Conference on Computers in Education, 2002. Proceedings., 1*, 1276–1280. <https://doi.org/10.1109/CIE.2002.1186210>
- Mitrovic, Antonija. (2006). Large-Scale Deployment of Three Intelligent Web-based Database Tutors. *28th International Conference on Information Technology Interfaces*, 135–140. <https://doi.org/10.1109/ITI.2006.1708466>
- Mitrovic, Antonija, Mathews, M., & Holland, J. (2012). Exploring two strategies for teaching procedures. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7315 LNCS, 499–504. https://doi.org/10.1007/978-3-642-30950-2_64
- Mitrovic, Antonija, & Suraweera, P. (2016). Teaching Database Design with Constraint-Based Tutors. *International Journal of Artificial Intelligence in Education*, 26(1), 448–456. <https://doi.org/10.1007/s40593-015-0084-6>
- Mitrovic, Antonija, Suraweera, P., Martin, B., & Weerasinghe, A. (2004). DB-suite: Experiences with Three Intelligent, Web-based Database Tutors. *Journal of Interactive Learning Research*, 15(4), 409–432.
- Mohtashami, M., & Scher, J. M. (2000). Application of Bloom ’ s Cognitive Domain Taxonomy to Database Design. *The Proceedings of the Information Systems Education Conference*.

<http://proc.edsig.org/2000/918/ISECON.2000.Mohtashami.pdf>

- Moody, D. D. L. (2003). Measuring the Quality of Data Models: An Empirical Evaluation of the Use of Quality Metrics in Practice. *ECIS 2003 Proceedings*. <https://doi.org/https://aisel.aisnet.org/ecis2003/78>
- Moores, T. T., Chang, J. C.-J., & Smith, D. K. (2006). Clarifying the role of self-efficacy and metacognition as predictors of performance. *ACM SIGMIS Database: The DATABASE for Advances in Information Systems*, 37(2–3), 125–132. <https://doi.org/10.1145/1161345.1161360>
- Morgan, D. L. (2007). Paradigms Lost and Pragmatism Regained: Methodological Implications of Combining Qualitative and Quantitative Methods. *Journal of Mixed Methods Research*, 1(1), 48–76. <https://doi.org/10.1177/2345678906292462>
- Morrison, B. B., Margulieux, L. E., & Guzdial, M. (2015). Subgoals, Context, and Worked Examples in Learning Computing Problem Solving. *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, 21–29. <https://doi.org/10.1145/2787622.2787733>
- Muñoz Barón, M., Wyrich, M., & Wagner, S. (2020). An Empirical Validation of Cognitive Complexity as a Measure of Source Code Understandability. *Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, 1–12. <https://doi.org/10.1145/3382494.3410636>
- Murray, M., & Guimaraes, M. (2008a). Animated database courseware: using animations to extend conceptual understanding of database concepts. *Journal of Computing Sciences in Colleges*, 24(2), 144–150. <http://dl.acm.org/citation.cfm?id=1409823.1409855>
- Murray, M., & Guimaraes, M. (2008b). Recent upgrades to an animated database courseware (ADbC). *Proceedings of the 46th Annual Southeast Regional Conference on XX, ACM-SE 46*, 515–516. <https://doi.org/10.1145/1593105.1593242>
- Nattassha, R., & Azizah, F. N. (2016). Database analysis and design learning tool based on problem/project-based learning. *Proceedings of 2015 International Conference on Data and Software Engineering, ICODSE 2015*, 65–68. <https://doi.org/10.1109/ICODSE.2015.7436973>
- Newell, A., & Simon, H. A. (1972). *Human problem solving*. Prentice-Hall Englewood Cliffs, NJ.
- Newman, M. E. J. (2001). The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences*, 98(2), 404–409. <https://doi.org/10.1073/pnas.98.2.404>

- Omar, M., & Baryannis, G. (2020). Semi-automated development of conceptual models from natural language text. *Data and Knowledge Engineering*, 127(December 2019), 101796. <https://doi.org/10.1016/j.datak.2020.101796>
- Pahl, C., Barrett, R., & Kenny, C. (2004). Supporting active database learning and training through interactive multimedia. *SIGCSE Bulletin (Association for Computing Machinery, Special Interest Group on Computer Science Education)*, 36(3), 27–31. <https://doi.org/10.1145/1026487.1008007>
- Pamela Neely, M. (2007). Mastery level learning and the art of database design. *Association for Information Systems - 13th Americas Conference on Information Systems, AMCIS 2007: Reaching New Heights*, 2, 818–826.
- Paterson, J., Wilson, J. N., & Leimich, P. (2012). *Uses of Peer Assessment in Database Teaching and Learning* (pp. 135–146). https://doi.org/10.1007/978-3-642-25704-9_17
- Patton, M. . (2002). *Qualitative Research & Evaluation Methods* (3rd ed.). SAGE Publications, Inc.
- Pavlić, M., Kaluža, M., & Vrčec, N. (2008). Database Complexity Measuring Method. *Proceedings of the 19th Central European Conference on Information and Intelligent Systems*.
- Pelchen, T., Mathieson, L., & Lister, R. (2020). On the evidence for a learning hierarchy in data structures exams. *ACE 2020 - Proceedings of the 22nd Australasian Computing Education Conference, Held in Conjunction with Australasian Computer Science Week*, 122–131. <https://doi.org/10.1145/3373165.3373179>
- Perkins, D., & Martin, F. (1985). *Fragile Knowledge and Neglected Strategies in Novice Programmers*.
- Perkins, D. N., & Salomon, G. (1992). Transfer of Learning. In T. Husén & T. . Postlethwaite (Eds.), *International Encyclopedia of Education* (2nd ed., pp. 425–441). Pergamon Press.
- Petersen, A., Craig, M., & Denny, P. (2016). Employing Multiple-Answer Multiple Choice Questions. *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, 252–253. <https://doi.org/10.1145/2899415.2925503>
- Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M. (2008). Systematic mapping studies in software engineering. *12th International Conference on Evaluation and Assessment in Software Engineering, EASE 2008*, 1–10. <https://doi.org/10.14236/ewic/ease2008.8>
- Petersen, K., Vakkalanka, S., & Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64, 1–18.

<https://doi.org/10.1016/j.infsof.2015.03.007>

- Pettit, R., Homer, J., Gee, R., Mengel, S., & Starbuck, A. (2015). An Empirical Study of Iterative Improvement in Programming Assignments. *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, 410–415. <https://doi.org/10.1145/2676723.2677279>
- Philpott, A., Robbins, P., & Whalley, J. L. (2007a). Accessing The Steps on the Road to Relational Thinking. *20th Annual Conference of the National Advisory Committee on Computing Qualifications, 2006*, 286. <http://citrenz.ac.nz/conferences/2007/286.pdf>
- Philpott, A., Robbins, P., & Whalley, J. L. (2007b). Accessing The Steps on the Road to Relational Thinking. *20th Annual Conference of the National Advisory Committee on Computing Qualifications, 2006*, 286.
- Piaget, J. (1929). *The child's conception of the world*. Routledge & K. Paul.
- Piaget, J. (1952). *The origins of intelligence in children*. International Universities Press.
- Piaget, J. (1971). *Genetic Epistemology*. W. W. Norton & Company.
- Piaget, J. (1985). *The equilibrium of cognitive structures*. University of Chicago Press.
- Piaget, J. (2003). *The Psychology of Intelligence*. Routledge. <https://doi.org/10.4324/9780203164730>
- Pittges, J., Phillips, R. H., & Uppuluri, P. (2009). Learning outcomes for an introductory database project. *ITNG 2009 - 6th International Conference on Information Technology: New Generations*, 981–986. <https://doi.org/10.1109/ITNG.2009.308>
- Podeschi, R. J. (2016). Building I.S. Professionals through a Real-World Client Project in a Database Application Development Course. *Information Systems Education Journal (ISEDJ)*, 14(6), 14.
- Polkinghorne, D. (1995). Narrative Configuration in Qualitative Analysis. *International Journal of Qualitative Studies in Education*, 8, 5–23.
- Poon, J., & Maher, M. L. (1997). Co-evolution and emergence in design. *Artificial Intelligence in Engineering*, 11(3), 319–327. [https://doi.org/10.1016/S0954-1810\(96\)00047-7](https://doi.org/10.1016/S0954-1810(96)00047-7)
- Prados, F., Boada, I., Soler, J., & Poch, J. (2005). An automatic correction tool for relational database schemas. *ITHET 2005: 6th International Conference on Information Technology Based Higher Education and Training, 2005*, 9–14. <https://doi.org/10.1109/ITHET.2005.1560331>

- Prados, F., Boada, I., Soler, J., & Poch, J. (2006). A web-based tool for entity-relationship modeling. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3980 LNCS, 364–372. https://doi.org/10.1007/11751540_39
- Protogeros, N., & Tzafilkou, K. (2015). Simple-talking database development: Let the end-user design a relational schema by using simple words. *Computers in Human Behavior*, 48, 273–289. <https://doi.org/10.1016/j.chb.2015.02.002>
- Rae, J. (1990). Getting to grips with database design: A step by step approach. *Computers & Education*, 14(6), 481–488. [https://doi.org/10.1016/0360-1315\(90\)90106-H](https://doi.org/10.1016/0360-1315(90)90106-H)
- Reeves, W. W. (1996). *Cognition and complexity: the cognitive science of managing complexity*. Scarecrow Press.
- Ren, J., & Nan, Z. (2012). Taking database design as trunk line of database courses. *Proceedings - 4th International Conference on Computational and Information Sciences, ICCIS 2012*, 767–769. <https://doi.org/10.1109/ICCIS.2012.310>
- Renkl, A., Atkinson, R. K., & Große, C. S. (2004). How Fading Worked Solution Steps Works – A Cognitive Load Perspective. *Instructional Science*, 32(1/2), 59–82. <https://doi.org/10.1023/B:TRUC.0000021815.74806.f6>
- Ridjanovic, D. (2003). Pedagogical Tools for Database Design , Development and Use. *Proceedings of the 2nd International Conference on Principles and Practice of Programming in Java*, 121–124. <https://dl.acm.org/doi/10.5555/957289.957326>
- Riessman, C. K. (2008). Narrative methods for the human sciences. In *Narrative methods for the human sciences*. Sage Publications, Inc.
- Robbert, M. A. (2000). Enhancing the value of a project in the database course. *SIGCSE Bulletin (Association for Computing Machinery, Special Interest Group on Computer Science Education)*, 36–40. <https://doi.org/10.1145/331795.331807>
- Rojas, J. M., Fraser, G., & Arcuri, A. (2015). Automated unit test generation during software development: a controlled experiment and think-aloud observations. *Proceedings of the 2015 International Symposium on Software Testing and Analysis*, 338–349. <https://doi.org/10.1145/2771783.2771801>
- Rosenthal, K., & Strecker, S. (2019). Toward a taxonomy of modeling difficulties: A multi-modal study

- on individual modeling processes. *40th International Conference on Information Systems, ICIS 2019*, 1–25. https://aisel.aisnet.org/icis2019/learning_environ/learning_environ/12/
- Rosenthal, K., Strecker, S., & Pastor, O. (2020). *Supplementary material for “Modeling Difficulties in Data Modeling: Similarities and Differences between Experienced and Non-experienced Modelers.”* <https://doi.org/https://doi.org/10.5281/zenodo.3992737>
- Rosenthal, K., Ternes, B., & Strecker, S. (2020). Understanding individual processes of conceptual modeling. In D. Bork, D. Karagiannis, & H. C. Mayr (Eds.), *Modellierung 2020* (pp. 77–92). Gesellschaft für Informatik e.V. <http://dl.gi.de/handle/20.500.12116/31848>
- Rosenthal, K., Wagner, J., & Strecker, S. (2022). Modeling styles in conceptual data modeling : Reflecting observations in a series of multimodal studies. *Proceedings of European Conference of Information Systems 2022*. https://aisel.aisnet.org/ecis2022_rp/73
- Ryan, S. D., Bordoloi, B., & Harrison, D. A. (2000). Acquiring Conceptual Data Modeling Skills: The Effect of Cooperative Learning and Self-Efficacy on Learning Outcomes. *Data Base for Advances in Information Systems*, 31(4), 9–24. <https://doi.org/10.1145/506760.506762>
- Saiedian, H., & Farhat, H. (1991). A team-oriented, project-intensive database course. *Proceedings of the Twenty-Second SIGCSE Technical Symposium on Computer Science Education - SIGCSE '91*, 192–198. <https://doi.org/10.1145/107004.107038>
- Scalabrino, S., Bavota, G., Vendome, C., Linares-Vasquez, M., Poshyvanyk, D., & Oliveto, R. (2021). Automatically Assessing Code Understandability. *IEEE Transactions on Software Engineering*, 47(3), 595–613. <https://doi.org/10.1109/TSE.2019.2901468>
- Schaeffer, D., & Olson, P. (2011). Adding the web to database class. *Journal of Computing Sciences in Colleges*, 26(3), 98–101.
- Schober, P., & Schwarte, L. A. (2018). Correlation coefficients: Appropriate use and interpretation. *Anesthesia and Analgesia*, 126(5), 1763–1768. <https://doi.org/10.1213/ANE.0000000000002864>
- Schwamborn, A., Thillmann, H., Opfermann, M., & Leutner, D. (2011). Cognitive load and instructionally supported learning with provided and learner-generated visualizations. *Computers in Human Behavior*, 27(1), 89–93. <https://doi.org/https://doi.org/10.1016/j.chb.2010.05.028>
- Scorce, R. (2010). Perspectives concerning the utilization of service learning projects for a computer science course. *Journal of Computing Sciences in Colleges*, 25(3), 75–81.

- Sedrakyan, G., Snoeck, M., & Poelmans, S. (2014). Assessing the effectiveness of feedback enabled simulation in teaching conceptual modeling. *Computers and Education*, 78, 367–382. <https://doi.org/10.1016/j.compedu.2014.06.014>
- Sfard, A. (1991). On the dual nature of mathematical conceptions: Reflections on processes and objects as different sides of the same coin. *Educational Studies in Mathematics*, 22(1), 1–36. <https://doi.org/10.1007/BF00302715>
- Sfard, A. (1998). On two metaphors for learning and the dangers of choosing just one. *Educational Researcher*, 27(2), 4–13. <https://doi.org/10.3102/0013189X027002004>
- Shackelford, R., Lunt, B., McGettrick, A., Sloan, R., Topi, H., Davies, G., Kamali, R., Cross, J., Impagliazzo, J., & LeBlanc, R. (2006). Computing Curricula 2005: The Overview Report. *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education - SIGCSE '06*, 456. <https://doi.org/10.1145/1121341.1121482>
- Shankaranarayanan, G., Stoddard, D., Gilleran, R., Beitelspacher, L. S., & Bravo, S. (2021). Experiential Learning of Information Systems in Functional Contexts: The Digital Brand Strategy Project. *Communications of the Association for Information Systems*, 49(1), 38–56. <https://doi.org/10.17705/1CAIS.04902>
- Sheard, J., Carbone, A., Lister, R., Simon, B., Thompson, E., & Whalley, J. L. (2008). Going SOLO to assess novice programmers. *Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education - ITiCSE '08*, 61(3), 209. <https://doi.org/10.1145/1384271.1384328>
- Sheard, J., Simon, Carbone, A., Chinn, D., Clear, T., Corney, M., D'Souza, D., Fenwick, J., Harland, J., Laakso, M. J., & Teague, D. (2013). How difficult are exams? A framework for assessing the complexity of introductory programming exams. *Conferences in Research and Practice in Information Technology Series*, 136, 146–154.
- Sheard, J., Simon, S., Hamilton, M., & Lönnberg, J. (2009). Analysis of research into the teaching and learning of programming. *Proceedings of the Fifth International Workshop on Computing Education Research Workshop - ICER '09*, 93. <https://doi.org/10.1145/1584322.1584334>
- Shebaro, B. (2018). Using Active Learning Strategies in Teaching Introductory Database Courses. *J. Comput. Sci. Coll.*, 33(4), 28–36.
- Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing: II.

- Perceptual learning, automatic attending and a general theory. *Psychological Review*, 84(2), 127–190. <https://doi.org/10.1037/0033-295X.84.2.127>
- Si-Said Cherfi, S., Akoka, J., & Comyn-Wattiau, I. (2002). Conceptual modeling quality-from EER to UML schemas evaluation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2503(October), 414–428. https://doi.org/10.1007/3-540-45816-6_38
- Simanjuntak, H. (2015). Proposed Framework for Automatic Grading System of ER Diagram. *2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE)*, 141–146. <https://doi.org/10.1109/ICITEED.2015.7408930>
- Simon. (2020a). Twenty-Four Years of ITiCSE Authors. *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, 205–211. <https://doi.org/10.1145/3341525.3387387>
- Simon. (2020b). Twenty-two years of Ace. *ACE 2020 - Proceedings of the 22nd Australasian Computing Education Conference, Held in Conjunction with Australasian Computer Science Week*, 203–210. <https://doi.org/10.1145/3373165.3373188>
- Simon, M. A. (1995). Reconstructing Mathematics Pedagogy from a Constructivist Perspective. *Journal for Research in Mathematics Education*, 26(2), 114. <https://doi.org/10.2307/749205>
- Simon, & Sheard, J. (2020). Twenty-Four Years of ITiCSE Papers. *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, 4, 5–11. <https://doi.org/10.1145/3341525.3387407>
- Simon, Sheard, J., Carbone, A., Chinn, D., Laakso, M. J., Clear, T., de Raadt, M., D’Souza, D., Lister, R., Philpott, A., Skene, J., & Warburton, G. (2012). Introductory programming: Examining the exams. *Proceedings of the Fourteenth Australasian Computing Education Conference*, 123, 61–70.
- Sindre, G. (2020). Code Writing vs Code Completion Puzzles: Analyzing Questions in an E-exam. *2020 IEEE Frontiers in Education Conference (FIE)*, 1–9. <https://doi.org/10.1109/FIE44824.2020.9273919>
- Sinha, A. P., & Vessey, I. (1999). An Empirical Investigation of Entity-Based and Object-Oriented Data Modeling: A Development Lifecycle Approach. *20th International Conference on Information Systems, 1991*, 229–244. <https://aisel.aisnet.org/icis1999/21>

- Sinha, B., Romney, G., Dey, P., & Amin, M. (2014). Estimation of database complexity from modeling schemas. *Journal of Computing Sciences in Colleges*, 30(2), 95–104.
- Skinner, B. F. (1953). Science and human behavior. In *Science and human behavior*. Macmillan.
- Soloway, E. (1986). Learning to program = learning to construct mechanisms and explanations. *Communications of the ACM*, 29(9), 850–858. <https://doi.org/10.1145/6592.6594>
- Someren, M. van, Barnard, Y., & Sandberg, J. (1994). The think aloud method: A practical guide to modelling cognitive processes. In *Knowledge based systems* (p. 208). Academic Press. <https://hdl.handle.net/11245/1.103289>
- Springsteel, F., Robbert, M. A., & Ricardo, C. M. (2000). The next decade of the database course. *Proceedings of the Thirty-First SIGCSE Technical Symposium on Computer Science Education - SIGCSE '00*, 41–45. <https://doi.org/10.1145/330908.331808>
- Starsinic, K. (1998). Perl Style. *The Perl Journal*, Fall 1998, 3(3).
- Storey, V. C., Dey, D., Ullrich, H., & Sundaresan, S. (1998). An ontology-based expert system for database design. *Data and Knowledge Engineering*, 28(1), 31–46. [https://doi.org/10.1016/S0169-023X\(98\)00012-3](https://doi.org/10.1016/S0169-023X(98)00012-3)
- Sugumaran, V., & Storey, V. C. (2006). The role of domain ontologies in database design: An ontology management and conceptual modeling environment. *ACM Transactions on Database Systems*, 31(3), 1064–1094. <https://doi.org/10.1145/1166074.1166083>
- Sugumaran, V., & Storey, V. C. (2003). Supporting Database Designers in Entity-Relationship Modeling: An Ontology- Based Approach. *ICIS 2003 Proceedings*, 59–71. <https://aisel.aisnet.org/icis2003/6>
- Suraweera, P., & Mitrovic, A. (2002). KERMIT: A Constraint-Based Tutor for Database Modeling. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 2363, pp. 377–387). https://doi.org/10.1007/3-540-47987-2_41
- Suraweera, P., & Mitrovic, A. (2004). An intelligent tutoring system for entity relationship modelling. *International Journal of Artificial Intelligence in Education*, 14(3–4), 375–417.
- Suwa, M., Gero, J., & Purcell, T. (2000). Unexpected discoveries and S-invention of design requirements: important vehicles for a design process. *Design Studies*, 21(6), 539–567.

[https://doi.org/10.1016/S0142-694X\(99\)00034-4](https://doi.org/10.1016/S0142-694X(99)00034-4)

- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2), 257–285. [https://doi.org/10.1016/0364-0213\(88\)90023-7](https://doi.org/10.1016/0364-0213(88)90023-7)
- Sweller, J. (2010). Element interactivity and intrinsic, extraneous, and germane cognitive load. *Educational Psychology Review*, 22(2), 123–138. <https://doi.org/10.1007/s10648-010-9128-5>
- Sweller, J., Van Merriënboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive Architecture and Instructional Design. *Educational Psychology Review*, 10(3), 251–296. <https://doi.org/10.1023/A:1022193728205>
- Taipalus, T., & Seppänen, V. (2020). SQL Education: A Systematic Mapping Study and Future Research Agenda. *ACM Transactions on Computing Education*, 20(3), 1–33. <https://doi.org/10.1145/3398377>
- Tashakkori, R., & Andrews, Z. W. (2018). A team software process approach to database course. *Proceedings of the ACMSE 2018 Conference, 2018-Janua*, 1–7. <https://doi.org/10.1145/3190645.3190676>
- Teague, D. (2015). Neo-Piagetian Theory and the Novice Programmer [Queensland University of Technology Faculty]. In *Thesis*. <https://eprints.qut.edu.au/86690/>
- Teague, D., Corney, M., Ahadi, A., & Lister, R. (2013). A Qualitative Think Aloud Study of the Early Neo-Piagetian Stages of Reasoning in Novice Programmers}. *Proceedings of the Fifteenth Australasian Computing Education Conference*, 136, 87–95.
- Teddle, C., & Tashakkori, A. (2010). Overview of Contemporary Issues in Mixed Methods Research. In *SAGE Handbook of Mixed Methods in Social & Behavioral Research* (pp. 1–42). SAGE Publications, Inc. <https://doi.org/10.4135/9781506335193.n1>
- Tempero, E., Denny, P., Luxton-Reilly, A., & Ralph, P. (2018). Objects count so count objects! *ICER 2018 - Proceedings of the 2018 ACM Conference on International Computing Education Research, June*, 187–195. <https://doi.org/10.1145/3230977.3230985>
- Thomas, P. (2012). Online Automatic Marking of Diagrams. *Systemic Practice and Action Research*, 26, 349–359. <https://doi.org/10.1007/s11213-012-9273-5>
- Thomas, P. G., Smith, N., & Waugh, K. G. (2007). Computer assisted assessment of diagrams. *Proceedings of the 12th Annual SIGCSE Conference on Innovation and Technology in Computer*

Science Education - ITiCSE '07, 68. <https://doi.org/10.1145/1268784.1268806>

- Thomas, P., Waugh, K., & Smith, N. (2015). Generalised Diagramming Tools with Automatic Marking. *Innovation in Teaching and Learning in Information and Computer Sciences*, 10(1), 22–34. <https://doi.org/10.11120/ital.2011.10010022>
- Thomas, P., Waugh, K., & Smith, N. (2006). Using patterns in the automatic marking of ER-diagrams. *Working Group Reports on ITiCSE on Innovation and Technology in Computer Science Education 2006*, 83–87. <https://doi.org/10.1145/1140124.1140149>
- Thompson, E., Luxton-Reilly, A., Whalley, J. L., Hu, M., & Robbins, P. (2008). Bloom’s taxonomy for CS assessment. *Proceedings of the Tenth Conference on Australasian Computing Education*, 78(January), 155–161.
- Thonggoom, O., Song, I., & An, Y. (2011). Semi-automatic Conceptual Data Modeling Using Entity and Relationship Instance Repositories. In J. M., D. L., & L. T.W (Eds.), *Conceptual Modeling – ER 2011* (pp. 219–232). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-24606-7_17
- Thota, N., Berglund, A., & Clear, T. (2012). Illustration of paradigm pluralism in computing education research. *Conferences in Research and Practice in Information Technology Series*, 123, 103–112.
- Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker, J. F., Sipior, J. C., & de Vreede, G. J. (2010). IS 2010: Curriculum guidelines for undergraduate degree programs in information systems. *Communications of the Association for Information Systems*, 26(1), 359–428. <https://doi.org/10.17705/1cais.02618>
- Trate, J. M., Teichert, M. A., Murphy, K. L., Srinivasan, S., Luxford, C. J., & Schneider, J. L. (2020). Remote interview methods in chemical education research. *Journal of Chemical Education*, 97(9), 2421–2429. <https://doi.org/10.1021/acs.jchemed.0c00680>
- Tshukudu, E., & Cutts, Q. (2020). Understanding Conceptual Transfer for Students Learning New Programming Languages. *Proceedings of the 2020 ACM Conference on International Computing Education Research*, 227–237. <https://doi.org/10.1145/3372782.3406270>
- Tuttle, S. (2002). Practical lessons from experience with the database design course project. *Journal of Computing Sciences in Colleges*, 32–42. <http://dl.acm.org/citation.cfm?id=771330>
- Tversky, A. (1977). Features of similarity. - 1977 - Tversky.pdf. *Psychological Review*, 84(4), 327–352.

- Udoh, E. (2006). *Teaching database in an integrated oracle environment*. 38(4), 71. <https://doi.org/10.1145/1189215.1189174>
- van Solingen, R., Basili, V., Caldiera, G., & Rombach, H. D. (2002). Goal Question Metric (GQM) Approach. In *Encyclopedia of Software Engineering*. John Wiley & Sons, Inc. <https://doi.org/10.1002/0471028959.sof142>
- Venable, J. R. (1996). Teaching novice conceptual data modellers to become experts. *Proceedings - 1996 International Conference Software Engineering: Education and Practice, SEEP 1996*, 50–56. <https://doi.org/10.1109/SEEP.1996.533980>
- Vessey, I. (1991). Cognitive Fit: A Theory-Based Analysis of the Graphs Versus Tables Literature. *Decision Sciences*, 22(2), 219–240. <https://doi.org/10.1111/j.1540-5915.1991.tb00344.x>
- Visual Paradigm. (2009). *Comparing Database Schemas Visually*. Visual Pa. <https://www.visual-paradigm.com/tutorials/comparedbschemasvisually.jsp>
- Visual Paradigm. (2022). *Visual Paradigm Community Edition* (No. 17). <https://www.visual-paradigm.com/download/community.jsp>
- Vygotsky, L. (1978a). Interaction between learning and development. In Mary Gauvain & M. Cole (Eds.), *Mind and Society* (2nd ed.). Harvard University Press.
- Vygotsky, L. (1978b). The development of higher psychological processes. In M. Cole, V. John-Steiner, S. Scribner, & E. Souberman (Eds.), *Mind in society*. Harvard University Press.
- Wadsworth, B. J. (2004). *Piaget's theory of cognitive and affective development* (5th ed). Pearson. <https://doi.org/LK> - <https://worldcat.org/title/474587380>
- Wagner, P. (2005). Teaching Data Modeling: Process and Patterns. *Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education - ITiCSE '05*, 168. <https://doi.org/10.1145/1067445.1067493>
- Wang, J., & Chen, H. (2014). Research and Practice on the Teaching Reform of Database Course. *Proceedings of the 2014 International Conference on Education Reform and Modern Management*, 35–37. <https://doi.org/10.2991/ermm-14.2014.63>
- Wang, N., & Ma, C. (2017). Teaching reform on database course for science laboratory class. *2017 12th International Conference on Computer Science and Education (ICCSE), Iccse*, 310–313. <https://doi.org/10.1109/ICCSE.2017.8085508>

- Wang, Y., McCoey, M., & Blum, T. (2020). Teaching Database for Freshmen: A Two-Thread Model. *J. Comput. Sci. Coll.*, 35(8), 33–43.
- Watson, R. T. (2006). The Essential Skills of Data Modeling. *Journal of Information Systems Education*, 17(1), 39.
- Wagh, K., Thomas, P., & Smith, N. (2004). Towards the automated assessment of entity-relationship diagrams. In: *Second Workshop of the Learning and Teaching Support Network - Information and Computer Science) TLAD (Teaching, Learning and Assessment of Databases)*. <https://oro.open.ac.uk/2455/>
- Weerasinghe, A., & Mitrovic, A. (2002). Enhancing learning through self-explanation. *Proceedings - International Conference on Computers in Education, ICCE 2002*, 244–248. <https://doi.org/10.1109/CIE.2002.1185914>
- Weerasinghe, A., & Mitrovic, A. (2006). Studying human tutors to facilitate self-explanation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4053 LNCS, 713–715. https://doi.org/10.1007/11774303_77
- Weerasinghe, A., Mitrovic, A., Thomson, D., Mogin, P., & Martin, B. (2011). Evaluating a general model of adaptive tutorial dialogues. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6738 LNAI, 395–402. https://doi.org/10.1007/978-3-642-21869-9_51
- Weitl-Harms, S. (2022). Database Service-learning Projects: Addressing Community Needs While Measuring and Meeting Computer Science Learning Outcomes. *2022 10th International Conference on Information and Education Technology (ICIET)*, 239–243. <https://doi.org/10.1109/ICIET55102.2022.9779002>
- Welch, M., & Lim, H. S. (2000). The Strategic Thinking of Novice Designers: Discontinuity Between Theory and Practice. *Journal of Technology Studies*, 26(2), 34–44. <https://doi.org/10.21061/jots.v26i2.a.6>
- Whalley, J., & Kasto, N. (2014a). How difficult are novice code writing tasks? A software metrics approach. *Conferences in Research and Practice in Information Technology Series*, 148, 105–112.
- Whalley, J., & Kasto, N. (2014b). A qualitative think-aloud study of novice programmers' code writing strategies. *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science*

Education - ITiCSE '14, 279–284. <https://doi.org/10.1145/2591708.2591762>

- Whalley, J. L., & Lister, R. (2009). The BRACElet 2009 . 1 (Wellington) Specification. *Proceedings of the Eleventh Australasian Conference on Computing Education*, 95, 9–18.
- Whalley, J. L., Lister, R., Thompson, E., Clear, T., Robbins, P., Ajith Kumar, P. K., & Prasad, C. (2006). An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies. *Conferences in Research and Practice in Information Technology Series*, 52(January 2006), 243–252.
- Whalley, J., Settle, A., & Luxton-Reilly, A. (2023). A Think-Aloud Study of Novice Debugging. *ACM Trans. Comput. Educ.*, 23(2). <https://doi.org/10.1145/3589004>
- Whalley, J., Settle, A., & Luxton-Reilly, A. (2021). Novice Reflections on Debugging. *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, 73–79. <https://doi.org/10.1145/3408877.3432374>
- Wieringa, R., Maiden, N., Mead, N., & Rolland, C. (2006). Requirements engineering paper classification and evaluation criteria: A proposal and a discussion. *Requirements Engineering*, 11(1), 102–107. <https://doi.org/10.1007/s00766-005-0021-6>
- Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. *ACM International Conference Proceeding Series*. <https://doi.org/10.1145/2601248.2601268>
- Wood, D., Bruner, J. S., & Ross, G. (1976). The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry*, 17(2), 89–100. <https://doi.org/10.1111/j.1469-7610.1976.tb00381.x>
- Woodford, K., & Bancroft, P. (2005). Multiple Choice Questions Not Considered Harmful. *Proceedings of the 7th Australasian Conference on Computing Education - Volume 42*, 109–116.
- Woodworth, R. S., & Thorndike, E. L. (1901). The influence of improvement in one mental function upon the efficiency of other functions. (I). *Psychological Review*, 8(3), 247–261. <https://doi.org/10.1037/h0074898>
- Worsley, M., & Blikstein, P. (2013). Programming Pathways: A Technique for Analyzing Novice Programmers' Learning Trajectories. In H. C. Lane, K. Yacef, J. Mostow, & P. Pavlik (Eds.), *Artificial Intelligence in Education, Lecture Notes in Computer Science* (Vol. 7926, pp. 844–847). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-39112-5_127

- Wurman, R. S. (1989). *Information anxiety* (1st ed.). Doubleday.
- Xu, S., Pollacia, L., & Lai, S. (2021). Analysis of students' database design skills in capstone projects. *Proceedings of the 2021 ACM Southeast Conference*, 199–203. <https://doi.org/10.1145/3409334.3452081>
- Yang, H. (2003). Comparing relational database designing approaches: some managerial implications for database training. *Industrial Management & Data Systems*, 103(3), 150–166. <https://doi.org/10.1108/02635570310465634>
- You, J., Wang, Z., Zhao, X., Kou, W., Zhang, Y., & Zhang, T. (2022). Online and offline hybrid teaching mode based on scenario cases: a case study of MySQL databases. *Fourth International Conference on Computer Science and Educational Informatization (CSEI 2022)*, 169–175. <https://doi.org/10.1049/icp.2022.1468>
- Zakharov, K., Mitrovic, A., & Johnston, L. (2008). Towards emotionally-intelligent pedagogical agents. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5091 LNCS, 19–28. https://doi.org/10.1007/978-3-540-69132-7_7
- Zhang, J., Luxton-Reilly, A., Denny, P., & Whalley, J. (2021). Scientific Collaboration Network Analysis for Computing Education Conferences. *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, 582–588. <https://doi.org/10.1145/3430665.3456385>
- Zhou, S., Xu, C., Wu, H., Zhang, J., Lin, Y., Wang, J., Gray, J., & Bryant, B. (2004). E-R modeler: A database modeling toolkit for eclipse. *Proceedings of the Annual Southeast Conference*, 160–165. <https://doi.org/10.1145/986537.986575>
- Zyl, S. Van, & Mentz, E. (2022). Developing deeper self-directed learning in database design: Factors that influence knowledge transfer. *Proceedings of the 17th Workshop in Primary and Secondary Computing Education*, 1(1), 1–2. <https://doi.org/10.1145/3556787.3556873>

Appendix A

Ethics Approval Letters

To collect the naturally occurring data from novice database modellers:



Auckland University of Technology Ethics Committee (AUTEC)

Auckland University of Technology
D-88, Private Bag 92006, Auckland 1142, NZ
T: +64 9 921 9999 ext. 8316
E: ethics@aut.ac.nz
www.aut.ac.nz/researchethics

31 July 2020

Jacqueline Whalley
Faculty of Design and Creative Technologies

Dear Jacqueline

Re Ethics Application: **20/167 Assessing Novice Data Modelling**

Thank you for providing evidence as requested, which satisfies the points raised by the Auckland University of Technology Ethics Committee (AUTEC).

Your ethics application has been approved for three years until 31 July 2023.

Standard Conditions of Approval

1. The research is to be undertaken in accordance with the [Auckland University of Technology Code of Conduct for Research](#) and as approved by AUTEC in this application.
2. A progress report is due annually on the anniversary of the approval date, using the EA2 form.
3. A final report is due at the expiration of the approval period, or, upon completion of project, using the EA3 form.
4. Any amendments to the project must be approved by AUTEC prior to being implemented. Amendments can be requested using the EA2 form.
5. Any serious or unexpected adverse events must be reported to AUTEC Secretariat as a matter of priority.
6. Any unforeseen events that might affect continued ethical acceptability of the project should also be reported to the AUTEC Secretariat as a matter of priority.
7. It is your responsibility to ensure that the spelling and grammar of documents being provided to participants or external organisations is of a high standard and that all the dates on the documents are updated.

AUTEC grants ethical approval only. You are responsible for obtaining management approval for access for your research from any institution or organisation at which your research is being conducted and you need to meet all ethical, legal, public health, and locality obligations or requirements for the jurisdictions in which the research is being undertaken.

Please quote the application number and title on all future correspondence related to this project.

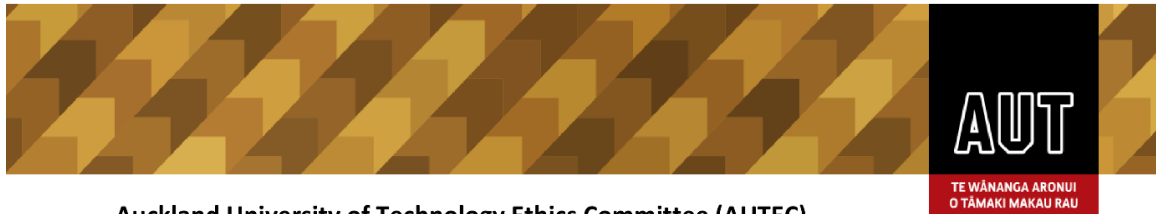
For any enquiries please contact ethics@aut.ac.nz. The forms mentioned above are available online through <http://www.aut.ac.nz/research/researchethics>

(This is a computer-generated letter for which no signature is required)

The AUTEC Secretariat
Auckland University of Technology Ethics Committee

Cc: asanthika.imbulpitiya@aut.ac.nz; Mali Senapathi

To collect the database modelling exam scripts:



Auckland University of Technology Ethics Committee (AUTC)

Auckland University of Technology
D-88, Private Bag 92006, Auckland 1142, NZ
T: +64 9 921 9999 ext. 8316
E: ethics@aut.ac.nz
www.aut.ac.nz/researchethics

15 December 2020

Jacqueline Whalley
Faculty of Design and Creative Technologies

Dear Jacqueline

Re Ethics Application: **20/167 Assessing Novice Data Modelling**

Thank you for providing evidence as requested, which satisfies the points raised by the Auckland University of Technology Ethics Committee (AUTC).

Your ethics application has been approved for three years until 14 December 2023.

Non-Standard Conditions of Approval

1. In the 'how will these discomforts and risks be alleviated' section of the Information Sheet please replace the first iteration of anonymous with confidential.

Non-standard conditions must be completed before commencing your study. Non-standard conditions do not need to be submitted to or reviewed by AUTC before commencing your study.

Standard Conditions of Approval

1. The research is to be undertaken in accordance with the [Auckland University of Technology Code of Conduct for Research](#) and as approved by AUTC in this application.
2. A progress report is due annually on the anniversary of the approval date, using the EA2 form.
3. A final report is due at the expiration of the approval period, or, upon completion of project, using the EA3 form.
4. Any amendments to the project must be approved by AUTC prior to being implemented. Amendments can be requested using the EA2 form.
5. Any serious or unexpected adverse events must be reported to AUTC Secretariat as a matter of priority.
6. Any unforeseen events that might affect continued ethical acceptability of the project should also be reported to the AUTC Secretariat as a matter of priority.
7. It is your responsibility to ensure that the spelling and grammar of documents being provided to participants or external organisations is of a high standard and that all the dates on the documents are updated.

AUTC grants ethical approval only. You are responsible for obtaining management approval for access for your research from any institution or organisation at which your research is being conducted and you need to meet all ethical, legal, public health, and locality obligations or requirements for the jurisdictions in which the research is being undertaken.

Please quote the application number and title on all future correspondence related to this project.

For any enquiries please contact ethics@aut.ac.nz. The forms mentioned above are available online through <http://www.aut.ac.nz/research/researchethics>

(This is a computer-generated letter for which no signature is required)

The AUTC Secretariat
Auckland University of Technology Ethics Committee

Cc: asanthika.imbulpitaya@aut.ac.nz; Mali Senapathi

To conduct think-aloud sessions:



Auckland University of Technology Ethics Committee (AUTEC)

Auckland University of Technology
D-88, Private Bag 92006, Auckland 1142, NZ
T: +64 9 921 9999 ext. 8316
E: ethics@aut.ac.nz
www.aut.ac.nz/researchethics

11 March 2021

Jacqueline Whalley
Faculty of Design and Creative Technologies

Dear Jacqueline

Re Ethics Application: **21/24 Assessing Novice Data Modelling**

Thank you for providing evidence as requested, which satisfies the points raised by the Auckland University of Technology Ethics Committee (AUTEC).

Your ethics application has been approved for three years until 11 March 2024.

Standard Conditions of Approval

1. The research is to be undertaken in accordance with the [Auckland University of Technology Code of Conduct for Research](#) and as approved by AUTEC in this application.
2. A progress report is due annually on the anniversary of the approval date, using the EA2 form.
3. A final report is due at the expiration of the approval period, or, upon completion of project, using the EA3 form.
4. Any amendments to the project must be approved by AUTEC prior to being implemented. Amendments can be requested using the EA2 form.
5. Any serious or unexpected adverse events must be reported to AUTEC Secretariat as a matter of priority.
6. Any unforeseen events that might affect continued ethical acceptability of the project should also be reported to the AUTEC Secretariat as a matter of priority.
7. It is your responsibility to ensure that the spelling and grammar of documents being provided to participants or external organisations is of a high standard and that all the dates on the documents are updated.

AUTEC grants ethical approval only. You are responsible for obtaining management approval for access for your research from any institution or organisation at which your research is being conducted and you need to meet all ethical, legal, public health, and locality obligations or requirements for the jurisdictions in which the research is being undertaken.

Please quote the application number and title on all future correspondence related to this project.

For any enquiries please contact ethics@aut.ac.nz. The forms mentioned above are available online through <http://www.aut.ac.nz/research/researchethics>

(This is a computer-generated letter for which no signature is required)

The AUTEC Secretariat
Auckland University of Technology Ethics Committee

Cc: asanthika.imbulpitaya@aut.ac.nz; Mali Senapathi

Appendix B

Related Documents – Naturally Occurring Data

Participant Information Sheet:

Participation Information Sheet

Start of Block: PIS

Hello,

I'm Asanthika Imbulpitiya and I am a PhD student at AUT. I would like to extend an invitation for you to participate in my doctoral research in the area of computer science education. As an undergraduate student, I found data modelling and database design challenging and this is why I am interested in learning about how we can more effectively assess and teach database design and modelling. I would like your consent to analyse your answers to your INFS601 (Logical Database Design) assessments, quizzes and exams. All you need to do is fill out a short questionnaire about your knowledge of software design and modelling

I will be conducting my research in collaboration with and under the supervision of Associate Professor Jacqueline Whalley and Dr. Mali Senapathi.

You can access the information related to the project using the below link. Once you have read the information, if you are happy to participate, you will be asked to indicate your consent to participate and then you will be taken to the prior knowledge survey.

The survey should take no longer than 10-15 minutes to complete. Your valuable participation will contribute to the future direction of how database design and modelling will be taught and assessed.

If you have any questions about this project please feel free to contact us either using the Blackboard discussion forum or by emailing me at asanthika.imbulpitiya@aut.ac.nz

Thank you

Asanthika

Q1 Participation information sheet

Date Information Sheet Produced:

8 th June 2020

Project Title

Assessing Novice Data Modelling

An Invitation

My name is Asanthika Imbulpitiya and I am a PhD student at AUT. I would like to extend an invitation to you to participate in my doctoral research in the area of computer science education. As an undergraduate student, I found data modelling and database design challenging and this is why I am interested in learning more about how we can more effectively assess and teach database design and modelling. I will be conducting my research in collaboration with and under the supervision of Associate Professor Jacqueline (Jacqui) Whalley and Dr. Mali Senapathi.

What is the purpose of this research?

No research has looked at the nature of data modelling questions, like the ones you are given in INFS601, or how difficult these tasks are. I believe that there is a hierarchy of data modelling skills and this is the main focus of my research. Earlier work has established that there is a hierarchy of skills in learning to program and intuitively it makes sense that there is some sort of dependency in building models to describe the technical requirements of software including databases. This research aims to investigate whether or not such a hierarchy of skills also exists for novices learning data and software analysis, design, and modelling. The findings of this research may be used for academic publications and presentations and will be included in my PhD thesis.

How was I identified and why am I being invited to participate in this research?

You are being invited to participate because you are taking INFS601 (Logical Database & Design) paper this semester. All students in INFS601 are being extended this invitation to participate.

How do I agree to participate in this research?

You will need to complete the online consent form. After you have indicated your agreement to participate you will then see a prior knowledge survey which you will complete as part of your agreement to participate. The survey is simply asking you a few questions about what you already know about software modelling and design and the extent of your knowledge and how confident you are in different aspects of data modelling if you do have some experience. Because INFS601 assumes no prior knowledge I expect that most of you will not know much about this topic so don't worry if you can't answer all the questions in the survey. Your participation in this research is voluntary (it is your choice) and whether or not you choose to participate will neither advantage nor disadvantage you. You are able to withdraw from the study at any time. If you choose to

you choose to withdraw from the study, then you will be offered the choice between having any data that is identifiable as belonging to you removed or allowing it to continue to be used. However, once the findings have been produced, the removal of your data may not be possible.

What will happen in this research?

In this research, you will be asked to provide us with permission to analyse the artefacts that you produce as part of your study in INFS601. These artefacts may include your answers and solutions to lab exercises, assessments, quizzes, tests and exams. Apart from the prior knowledge survey all other data you give me permission to collect and analyse is from your normal course work and assessments. Your data will be combined and analysed along with the data provided by other students in order to establish things like how difficult the tasks were or how the skills and knowledge required to answer different types of problems relate to each other. Sometimes images of your database designs or models may be used in publications arising from this research with your prior permission. You can choose whether or not your database designs/models to be used in the publications in the consent form.

What are the discomforts and risks?

It is unlikely that you will feel any risk or discomfort however it is possible that you may experience some discomfort when filling in the prior knowledge survey and not like revealing either a lack of experience or substantial experience in the area. Two of the researchers of this study are involved in delivering the INFS601 paper and there is the potential for you to perceive that participating in the study and/or any data you provide might affect your grades and be a source of conflict. The study has been carefully designed to ensure that this potential risk is alleviated.

How will these discomforts and risks be alleviated?

As part of this study, you will be asked to complete a prior knowledge survey about your experiences and knowledge of data modelling before studying INFS601. Jacqui will be the only researcher who will know who you are and the data you provide will be summarised so that no-one other than herself will be able to identify who you in any reports or in my thesis. Because one of my supervisors Dr. Mali Senapathi is a lecturer on INFS601 and myself is a teaching assistant I have carefully designed my research study so that both of us will not know whether or not you have participated in this research and we will only be able to see the data provided by you once all your data has been anonymised. In addition, we will not start analysing your data until the paper is over and your results have been finalised.

What are the benefits?

As a participant of this study, you will be contributing to the design, development and improvement of the assessment and overall learning experiences in INFS601. These improvements will benefit future students studying this subject both within and outside of AUT. I will benefit from this research as your data will enable my PhD study. I and my

collaborators will gain a deeper understanding of the current INFS601 assessments and exercises and how they impact student learning. We will gain a deeper understanding of the source of student difficulties in data modelling. This improved understanding will result in improved, fairer and better structured assessments for all students. The dissemination of the results as peer reviewed papers and at conferences will enable educators to create a better-structured learning experience for novice data modellers like yourself.

How will my privacy be protected?

In order to protect your privacy, so that you are not identifiable to others, only Jacqui will have access to information related to your identity and this information will be kept confidential in a secured location. The other named researchers will only have access to your anonymised survey, lab exercises and assessment data. In any reports or academic papers arising from this research, you will not be named or identifiable. Further, none of your data will be shared with other parties or used for any other purposes apart from this study.

What are the costs of participating in this research?

None, apart from the time taken to go view the invitation to participate, read the information sheet and provide the consent, and complete the prior knowledge survey. It is expected that in total most of you will spend about 5- 10 minutes to complete the prior knowledge survey. Data for the research will be derived from the activities and assessments related to INFS601 paper.

What opportunity do I have to consider this invitation?

You will have six weeks to consider this invitation.

Will I receive feedback on the results of this research?

If you would like to receive feedback on the results of this research you can indicate this when completing the consent form and provide an email address for it to be sent to you.

What do I do if I have concerns about this research?

Any concerns regarding the nature of this project should be notified in the first instance to the Project Supervisor, Associate Professor Jacqueline Whalley, jwhalley@aut.ac.nz , (+649) 921 9999 ext 5203.

Concerns regarding the conduct of the research should be notified to the Executive Secretary of ATEC, ethics@aut.ac.nz , (+649) 921 9999 ext 6038.

Whom do I contact for further information about this research?

Please keep this Information Sheet and a copy of the Consent Form for your future reference. You are also able to contact the research team as follows:

Researcher Contact Details:

Asanthika Imbulpitiya, asanthika.imbulpitiya@aut.ac.nz

Dr. Mali Senapathi, mali.senapathi@aut.ac.nz , (+64) 921 9999 extension 5213

Project Supervisor Contact Details:

Associate Professor Jacqueline Whalley, jwhalley@aut.ac.nz, (+649) 921 9999 ext 5203.

Q2 Confirmation

☐ I confirm that I have read the information provided in the Participation Information Sheet. (4)

Skip To: End of Survey If Confirmation = I confirm that I have read the information provided in the Participation Information Sheet.

End of Block: PIS

Consent Form:

Consent Form

Start of Block: Default Question Block

Project title: **Assessing Novice Data Modelling**
Project Supervisor: **Associate Professor Jacqueline Whalley**
Researcher: **Asanthika Imbulpitiya**

- I have read and understood the information provided about this research project in the Information Sheet dated 8th June 2020.
- I have had an opportunity to ask questions and to have them answered.
- I understand that taking part in this study is voluntary (my choice) and that I may withdraw from the study at any time without being disadvantaged in any way.
- I understand that images of my database designs/models may be used in publications arising from this research.
- I understand that if I withdraw from the study then I will be offered the choice between having any data that is identifiable as belonging to me removed or allowing it to continue to be used. However, once the findings have been produced, removal of my data may not be possible.

Q1 Name of the participant: (Please enter your name as on your AUT student id)

Q2 Student ID number of the participant:

Q3 Would you like to receive a summary of the research findings?

☐ Yes (1)

☐ No (2)

Display This Question:

If Would you like to receive a summary of the research findings? = Yes

Q4

Email address to which we should send a summary of the research findings:



Q5 I give permission for my database designs/models to be reproduced in publications arising from this research.

☐ Yes (1)

☐ No (2)

Q6 I have read the above information and I consent to participate.

☐ Yes (1)

☐ No (2)

Skip To: End of Survey If I have read the above information and I consent to participate. = Yes

End of Block: Default Question Block

Prior Knowledge Survey:

Prior Knowledge Survey

Start of Block: Personal Information

Personal Information

Q1 What is your area of specialization (major) in the BCIS degree program?

- ☐ Analytics (1)
 - ☐ Computational Intelligence (2)
 - ☐ Computer Science (3)
 - ☐ IT Service Science (4)
 - ☐ Network and Security (5)
 - ☐ Software Development (6)
-

Q2 Is English your 1st language?

- ☐ Yes (1)
- ☐ No (2)

End of Block: Personal Information

Start of Block: Background in Software Modelling

Background in Software Modelling

Q1 Have you completed Data and Process Modelling paper (INFS600) ?

☐ Yes (1)

☐ No (2)

Q2 Are you familiar with UML modelling?

☐ Yes (1)

☐ No (2)

Skip To: End of Block If Q2 = No

Display This Question:

If Q2 = Yes

Q2.1 Which type of UML diagrams are you familiar with?

☐

Use Case Diagram (1)

☐

Activity Diagram (2)

☐

Interaction Diagram (Sequence or Communication) (3)

☐

State Chart Diagram (4)

☐

Class Diagram (5)

☐

Component Diagram (6)

☐

Deployment Diagram (7)

Display This Question:

If Q2 = Yes

Q2.2 How confident are you in creating these diagrams?

	Not at all Confident (1)	Not Very Confident (2)	Neither (3)	Fairly Confident (4)	Very Confident (5)
Use Case Diagram (1)	☐	☐	☐	☐	☐
Activity Diagram (2)	☐	☐	☐	☐	☐
Interaction Diagram (3)	☐	☐	☐	☐	☐
State Chart Diagram (4)	☐	☐	☐	☐	☐
Class Diagram (5)	☐	☐	☐	☐	☐
Component Diagram (6)	☐	☐	☐	☐	☐
Deployment Diagram (7)	☐	☐	☐	☐	☐

End of Block: Background in Software Modelling

Start of Block: Block 2

Q3 Do you have any prior experience in database modelling and design?

☐ Yes (1)

☐ No (2)

Skip To: Q3.1 If Q3 = Yes

Skip To: End of Survey If Q3 = No

Q3.1 If so how confident are you about your database modelling and design skills?

	Not at all Confident (1)	Not Very Confident (2)	Neither (3)	Fairly Confident (4)	Very Confident (5)
confidence in database design and modelling (1)	☐	☐	☐	☐	☐

Q3.2 Where did you first learn about data modelling?

- ☐ As a part of formal qualification (e.g. secondary, tertiary) (1)
- ☐ Any other course that is not part of a formal education (e.g. short courses with a teacher) (2)
- ☐ On the job (e.g. workplace training) (3)
- ☐ Self-taught (e.g. youtube, online courses such as coursera, edex etc.) (4)
-

Q3.3 What type of data modelling have you learned before?

- ☐ Conceptual Modelling (e.g. UML diagrams, ER diagrams, EER diagrams) (1)
- ☐ Logical Modelling (e.g. Relational models, Hierarchical models) (2)
- ☐ Physical Modelling (e.g. Oracle, SQL server) (3)
-

Q3.4 If you have learned data modelling before how well do you understand the below concepts?

	Not well at all (1)	Slightly well (2)	Moderately well (3)	Very well (4)	Extremely well (5)
Concept of entities (1)	☐	☐	☐	☐	☐
Concept of attributes (2)	☐	☐	☐	☐	☐
Concept of relationships (3)	☐	☐	☐	☐	☐
Concept of keys (e.g. primary key, foreign key) (4)	☐	☐	☐	☐	☐

Q3.5 Is there anything else you would like to tell us about your experiences with or of data modelling and database design?

End of Block: Block 2

Appendix C

Related Documents – Exam Script Collection

Participant Information Sheet:



Participant Information Sheet

Date Information Sheet Produced:

9th September 2020

Project Title

What are we assessing in data modelling?

An Invitation

My name is Asanthika Imbulpititiya and I'm a PhD student at Auckland University of Technology, New Zealand. I'm conducting my research in Computer Science Education (CSEd) under the supervision of Associate Professor Jacqui Whalley and Dr Mali Senapathi. I would like to extend an invitation to you to participate in part of my doctoral research in which I am looking at different questions asked in introductory database courses at the tertiary level. I am ultimately interested in understanding how to better assess database modelling skills and knowledge.

What is the purpose of this research?

Very little research has been undertaken that looks at how to assess novice data modellers (and database design) at the tertiary level. There is no doubt that this is an important skill for graduates. Work examining the difficulties for novice modellers has shown that students have difficulty understanding relationships, semantics and representation in database design documents. Most research in this area was undertaken nearly 20 years ago even though databases are still important today. This research aims to examine how and what contemporary database courses are assessing. It is our conjecture that little has changed in the field despite the importance of the subject area and the need to improve on the assessment.

The research also forms part of a larger PhD topic and the findings will be published in the thesis and disseminated through relevant conferences/journals in the discipline.

How was I identified and why am I being invited to participate in this research?

You are being invited to participate because you are subscribed to this specialist mailing list/institutional newsletter. All members subscribed to the specialist mailing list/institutional newsletter are being extended this invitation to participate.

How do I agree to participate in this research?

When you receive the invitation to participate you can go through the attached participation information sheet. Once you have read and understood the provided information sheet and if you are willing to participate you can fill out the consent form section of the email to send it back as a reply. Once the consent is received we will send a link to a submission form which provides the facility to upload exams and module/paper descriptors via the request file feature of OneDrive. This feature will only allow you to upload your files. No edit or view permission is given to the OneDrive to further enable the security of data.

Your participation in this research is voluntary (it is your choice) and whether or not you choose to participate will neither advantage nor disadvantage you.

What will happen in this research?

In this research, you will be asked to provide us with exam scripts from your undergraduate database course(s) preferably from the mid and/or final exam scripts along with the course/module outlines/descriptors indicating the learning outcomes and the course content.

The questions asked in each examination, that are relevant to database design and modelling, will be analysed using a framework that includes Bloom's level, Topic and Task Type. The results will be collated and analysed using simple descriptive statistics to give an overall picture of the nature of the assessments currently being used in database design and modelling. The questions in the exam will be mapped to the corresponding learning outcomes to see how well they correlate.

What are the discomforts and risks?

It is unlikely that you will feel any level of discomfort or risk considering the facts that you are academics and inherently reflective practitioners and you will be only required to provide exam scripts and course content information.

However, while highly unlikely, there is the possibility that the information you provide may be attributed back to you or your institution and this could potentially affect some facets of your role as an academic. The study has been carefully designed to ensure that this potential risk is alleviated.

How will these discomforts and risks be alleviated?

Your participation will be confidential. The data you supply will be anonymised at the institution level and reported in an aggregated manner. Some questions may be reproduced as exemplars and I will take care to remove any identifiable information within a question. For example, a question might say something like “a student in INFS601 has designed the ERD below and would like your advice on how to improve their design”, in such a case the INFS601 course identifier would be removed from the question stem.

What are the benefits?

As a participant of this study, you will have a voice in furthering understanding of the assessment of database design and modelling topics. You will also benefit from the outcomes of the study, using the results to inform the design of future examinations and improve the assessments for your students.

I will benefit from this research as your data will enable my PhD study.

I and my collaborators will gain a better understanding of how to assess database modelling. This improved understanding will result in improved and fairer assessments for students studying databases. The dissemination of the results as peer-reviewed papers and at conferences will enable educators to create a better-structured learning experience for novice data modellers.

How will my privacy be protected?

In order to protect your privacy, the raw data will be anonymised at the point of collection. The other named researchers will only have access to your anonymised data for the purposes of analysis. In any reports or academic papers arising from this study, you will not be named or identifiable. Further, none of your data will be shared with other parties or used for any other purposes apart from this study.

All the provided data will be downloaded to a password-protected external storage device and will be securely stored in my office. At the end of the study, all the data will be destroyed using a tool like DBAN.

What are the costs of participating in this research?

None, apart from the time taken to go view the invitation to participate, read the information sheet, provide the consent and upload the data. At most, we anticipate that you will spend about 30 minutes to gather the data required and upload to the onedrive folder.

What opportunity do I have to consider this invitation?

You will have eight weeks to consider this invitation.

Will I receive feedback on the results of this research?

If you would like to receive feedback on the results of this research, you can indicate this when completing the consent form and the feedback will be sent to the same email address you used when providing the consent.

What do I do if I have concerns about this research?

Any concerns regarding the nature of this project should be notified in the first instance to the Project Supervisor, Associate Professor Jacqueline Whalley, jwhalley@aut.ac.nz, (+649) 921 9999 ext 5203.

Concerns regarding the conduct of the research should be notified to the Executive Secretary of AUTC, ethics@aut.ac.nz, (+649) 921 9999 ext 6038.

Whom do I contact for further information about this research?

Please keep this Information Sheet and a copy of the Consent Form for your future reference. You are also able to contact the research team as follows:

Researcher Contact Details:

Asanthika Imbulpitiya, asanthika.imbulpitiya@autuni.ac.nz,

Dr. Mali Senapathi, mali.senapathi@aut.ac.nz, (+64) 921 9999 extension 5213

Project Supervisor Contact Details:

Associate Professor Jacqueline Whalley, jwhalley@aut.ac.nz, (+649) 921 9999 ext 5203.

Consent Form (embedded to the invitation email):

Consent Form

Email subject line: Invitation to participate in the study - "What are we assessing in data modelling?"

Email body:

Hello,

I'm Asanthika Imbulpitiya, a PhD student from Auckland University of Technology, New Zealand. I am looking at different assessment questions in Database Modelling and Design to understand how better assess database modelling skills and knowledge. As a preliminary study, I am looking at the nature of novice level database modelling questions and identifying what types of problems students are asked to solve and how difficult these problems are. I will be using taxonomies (such as Bloom), concepts assessed, and metrics to classify the problems.

I'm conducting my research in collaboration with and under the supervision of Associate Professor Jacqui Whalley and Dr Mali Senapathi.

Information related to the study can be accessed via [Participation Information Sheet](#) and the same is attached to the email as well. Once you have read the information, if you are happy to participate, please fill the below-given consent form and send it as a reply to this email. We will be sending you a link to submit the materials once we received your consent for participation.

If you have any questions about this project, please feel free to contact me via this email.

Thank you

Asanthika

Consent Form

Project title: **What are we assessing in data modelling?**
Project Supervisor: **Associate Professor Jacqueline Whalley**
Researcher: **Asanthika Imbulpitiya**

- I have read and understood the information provided about this research project in the Information Sheet dated 9th September 2020.
 - I have had an opportunity to ask questions and to have them answered.
 - I understand that taking part in this study is voluntary and that I may withdraw from the study at any time without being disadvantaged in any way.
 - I understand that questions of my exam scripts may be reproduced as exemplars in publications arising from this research.
 - I have my institute's approval to share the exam scripts with you.
1. What is your name?
 2. What is the institute you are currently affiliated with?
 3. Would you like to receive a summary of the research findings? Yes No (Please strikethrough the incorrect option)
 4. I have read the above information and I consent to participate. Yes No (Please strikethrough the incorrect option)
-

Material submission form:

Material Submission Form

Start of Block: Default Question Block

Q1 Please upload your course outline/ course descriptor by clicking the below link.

[Click Here](#)

Q2 Please upload your exam scripts by clicking the below link.

[Click Here](#)

Q3 Is there any other additional information you would like to tell us about your course?

End of Block: Default Question Block

Appendix D

Related Documents – Think-aloud Data

Participant Information Sheet:



Participant Information Sheet

Date Information Sheet Produced:

11th January 2021

Project Title

Assessing Novice Data Modelling

An Invitation

My name is Asanthika Imbulpitiya and I am a PhD student at AUT. I would like to extend an invitation to you to participate in my doctoral research which looks at how we can more effectively assess and teach database design and modelling. I will am conducting my research under the supervision of Associate Professor Jacqueline Whalley and Dr. Mali Senapathi.

What is the purpose of this research?

I am interested in finding out how you learn database modelling, what challenges you face when designing a database and what strategies you use to solve a database design problem.

The findings of this research may be used for academic publications and presentations and will be included in my PhD thesis.

How was I identified and why am I being invited to participate in this research?

You are being invited to participate because you are taking INFS601 or COMP508 paper this semester. In a situation where we will get more than the required no of participants, we will select a representative sample to participate in the study.

How do I agree to participate in this research?

If you agree to participate in this research, you need to complete the Consent Form provided to you. You can get a hard copy of the form in your lab class or download it as a softcopy using the Blackboard course page. Once you have completed the consent form you can either photograph or scan or digitally sign the form and email it to me. Alternatively, you can email me and arrange to give me a hard copy in person after completing the form.

Your participation in this research is voluntary (it is your choice) and whether or not you choose to participate will neither advantage nor disadvantage you. You are able to withdraw from the study at any time. If you choose to withdraw from the study, then you will be offered the choice between having any data that is identifiable as belonging to you removed or allowing it to continue to be used. However, once the findings have been produced, the removal of your data may not be possible.

What will happen in this research?

In this research, you will be asked to attend four 40-60 minute sessions during the semester. These sessions will be scheduled out of your class time in consultation with you and at your convenience. The sessions will be scheduled after you have covered the required concepts in your lecture so that you will be familiar with the modelling tasks that you are asked to undertake.

The study involves you attempting some database modelling tasks while telling me what you are thinking as you are solving them. Your voice and your pen and paper drawings and notes (and/or computer screen) will be recorded.

Recordings will be made using a smartpen, that records your voice and anything you write on the paper (if you answer the questions on a paper) or using a tool like Camtasia if the modelling task is done using Visual Paradigm.

Your data will be transcribed and you will be given the opportunity to verify the data prior to it being analysed. When I analyse your data I will be trying to establish things like how difficult the tasks were or what are the skills and knowledge required to answer different types of problems related to each other. Further, your data will be used to create a learning trajectory to understand how your learning progresses and which exercises help you to extend your knowledge.

What are the discomforts and risks?

Sometimes you may feel slight discomfort or embarrassment if you find it difficult to answer the modelling tasks and not like revealing either a lack of experience or substantial experience in the area.

One of the researchers is also a teacher on the INFS601/COMP508 paper and there is the potential for you to perceive that participating in the study and/or any data you provide might affect your grades and be a source of conflict. The study has been carefully designed to ensure that this potential risk is alleviated.

If you are a repeat student taught INFS601 labs by me in 2020 please note that I have no involvement in this years course and its outcomes.

How will these discomforts and risks be alleviated?

I will be the only researcher on the team who will know who you are and I will assist you throughout the sessions easing you into the process and will guide you when you are completing the tasks if you come across any issues. Also please note that I will not provide any negative evaluation or criticise your current approach but rather if you want me to I can provide you with constructive feedback at the end of the sessions which will help you in your learning process.

Once the audio recording of the think aloud sessions are transcribed, you will get a copy of the transcription (if you wish) to check and verify. You will be given two weeks to verify the transcriptions and inform me of any corrections before the beginning of the analysis.

Because one of my supervisors, Dr. Mali Senapathi, is a lecturer on INFS601/COMP508 I have carefully designed my research study so that she will not know whether or not you have participated in this research and she will only be able to see the data provided by you once all your data has been anonymised.

What are the benefits?

If you decide to participate in the study, as well as benefit from extra practice undertaking tasks that support your learning, and you will play an important part in improving the way we teach and assess data modelling!

At the end of each session, you will have the opportunity to benefit from post session feedback that can help your learning and understanding of some of the topics being taught in INFS601 and COMP508.

I will benefit from this research as your data will enable my PhD study.

I and my collaborators will gain a deeper understanding of the current data modelling assessments and exercises and how they impact student learning. We will gain a deeper understanding of the source of student difficulties in data modelling. This improved understanding will result in improved, fairer and better structured assessments for all students. The dissemination of the results as peer reviewed papers and at conferences will enable educators to create a better-structured learning experience for novice data modellers like yourself.

How will my privacy be protected?

In order to protect your privacy, so that you are not identifiable to others, only I will have access to information related to your identity and this information will be kept confidential. The other named researchers will only have access to your anonymised interview data. In any reports or academic papers arising from this research, you will not be named or identifiable.

What are the costs of participating in this research?

You will have to participate in four one hour sessions during the semester totalling up to four hours followed by a brief feedback session which can be considered as a personal tutoring session as a form of koha (a Maori tradition of giving a gift as a thank you) for your contribution to the research which is focused on the tasks you completed.

What opportunity do I have to consider this invitation?

You will have 2 weeks after the brief in-class session which will be conducted to introduce the study.

Will I receive feedback on the results of this research?

If you would like to receive feedback on the results of this research you can indicate this when completing the consent form and provide an email address for it to be sent to you.

What do I do if I have concerns about this research?

Any concerns regarding the nature of this project should be notified in the first instance to the Project Supervisor, Associate Professor Jacqueline Whalley, jwhalley@aut.ac.nz, (+649) 921 9999 ext 5203.

Concerns regarding the conduct of the research should be notified to the Executive Secretary of AUTC, ethics@aut.ac.nz, (+649) 921 9999 ext 6038.

Whom do I contact for further information about this research?

Please keep this Information Sheet and a copy of the Consent Form for your future reference. You are also able to contact the research team as follows:

Researcher Contact Details:

Asanthika Imbulpitiya, asanthikaimbulpitiya@autuni.ac.nz,

Dr. Mali Senapathi, mali.senapathi@aut.ac.nz, (+64) 921 9999 extension 5213.

Project Supervisor Contact Details:

Associate Professor Jacqueline Whalley, jwhalley@aut.ac.nz, (+649) 921 9999 ext 5203.

Approved by the Auckland University of Technology Ethics Committee on *11th March 2021*, AUTC Reference number *21/24*.

Consent Form:



Consent Form

Project title: **Assessing Novice Data Modelling**

Project Supervisor: **Associate Professor Jacqueline Whalley**

Researcher: **Asanthika Imbulpitiya**

- ☐ I have read and understood the information provided about this research project in the Information Sheet dated 11th January 2021.
- ☐ I have had an opportunity to ask questions and to have them answered.
- ☐ I understand that notes will be taken during the think aloud interviews while performing modelling tasks and that they will also be audio-taped and transcribed, and I will get an opportunity to verify the transcriptions if I wish to do so.
- ☐ I understand that images of my database designs/models may be used in publications arising from this research and that these will not be identifiable to me.
- ☐ I understand that taking part in this study is voluntary (my choice) and that I may withdraw from the study at any time prior to the completion of data analysis without being disadvantaged in any way.
- ☐ I understand that if I withdraw from the study then I will be offered the choice between having any data that is identifiable as belonging to me removed or allowing it to continue to be used. However, once the findings have been produced, the removal of my data may not be possible.
- ☐ I agree to take part in this research. Yes ☐ No ☐
- ☐ I wish to receive a summary of the research findings (please tick one): Yes ☐ No ☐

Participant's signature:

Participant's name:

Participant's email address (if you wish to receive a summary of the findings):

.....

Date:

Approved by the Auckland University of Technology Ethics Committee on 11th March 2021 AUTEK Reference number 21/24

Note: The Participant should retain a copy of this form.

Appendix E

Principal Component Analysis

Principal Component Analysis (PCA) is a variable reduction technique that maximises the amount of variance accounted for in the observed variables by a smaller group of variables called components. In this study, PCA was used to simplify the common variance among the set of possible variables, the database metrics discussed in Chapter 5. In this instance, PCA was conducted as an attempt to develop a predictive model that could predict the difficulty of *model creation* and *model transformation* tasks.

PCA for *model creation* tasks

Before conducting the PCA, an R-matrix was calculated to determine variables that do not correlate with any other variables (or very few) or variables that are highly correlated with other variables. Having highly correlated variables are known as a singularity (perfectly correlated variables) or multicollinearity (highly correlated variables) in a data set which needs to be eliminated before a PCA as it could increase the standard error of a coefficient which in turn can make a variable statistically insignificant where they should be significant.

Figure 1 provides the correlation matrix for the nine metrics discussed in Chapter 5 for *model creation* tasks, where the output indicates the pairwise Pearson's correlation coefficient between the metrics.

Correlation Matrix ^{a,b}									
	Number_of_E	Number_of_R	Number_of_A	Number_of_MVA	Number_of_ISAR	Number_of_1MR	Number_of_MNR	Readability_FKGL	Number_of_CA
Correlation	Number_of_E	1.000	.778	.775	-.447	.000	.485	.433	.430
	Number_of_R	.778	1.000	.989	.063	.253	.909	-.122	.324
	Number_of_A	.775	.989	1.000	-.008	.379	.927	-.187	.366
	Number_of_MVA	-.447	.063	-.008	1.000	-.200	.217	-.387	-.697
	Number_of_ISAR	.000	.253	.379	-.200	1.000	.542	-.775	.406
	Number_of_1MR	.485	.909	.927	.217	.542	1.000	-.525	.312
	Number_of_MNR	.433	-.122	-.187	-.387	-.775	-.525	1.000	-.082
	Readability_FKGL	.430	.324	.366	-.697	.406	.312	-.082	1.000
	Number_of_CA	.372	.246	.284	.055	.055	.120	-.519	1.000
a. Determinant= .000									
b. This matrix is not positive definite.									

Figure 1 Correlation matrix between the complexity metrics for model creation tasks

Based on the above output, it indicates that the matrix is not positive definite, which usually occurs when there are too many variables and too few cases of data/observations. In this case, variables were the metrics and cases/observations were the number of database modelling tasks. Further, Lorenzo-Seva & Ferrando (2021) discussed the conditions which should be violated for a correlation matrix to be not positive definite as follows,

- The number of observations from which R is computed is larger than the number of variables.

- All the correlations are based on the same number of observations.
- There are no linear dependencies among the variables.
- There are no items with zero variance.

Considering the data set in Table 1, it is evident that the number of observations (questions data are gathered from) is less than the number of variables (metrics). However, Hogarty et al. (2005) noted the following in their study related to the sample size and the quality of the factor solutions, “*Consistent with the general findings of the MacCallum et al. (1999) study, our results show that there was not a minimum level of N or $N:p$ ratio to achieve good factor recovery across conditions examined*” (p. 222). Considering this fact, we decided to conduct the PCA after screening the data. The calculated values for metrics NMVA and the NISAR almost had zero variance since, in all the questions, the metrics were 0 apart from questions 2 and 3 for NMVA and NISAR, respectively. Due to this reason, it was decided to remove these metrics from the analysis altogether.

Table 1 Questions and related metrics in model creation tasks

	Q 1	Q 2	Q 3	Q4	Q5	Q6
Difficulty (%) - based on top quartile	33.33	28.57	31.08	25	33.33	30
Number of Entities (NE)	8	7	8	8	7	10
Number of Attributes (NA)	36	42	50	36	31	58
Number of Composite Attributes (NCA)	10	9	9	10	6	9
Number of Multi-valued Attributes (NMVA)	0	1	0	0	0	0
Number of Relationships (NR)	5	7	8	5	4	11
Number of ISA Relationships (NISAR)	0	0	2	0	0	0
Number of one-to-many Relationships (N1:MR)	2	6	8	2	2	8
Number of many-to-many Relationships (NM:NR)	3	1	0	3	2	3
Readability metric – Flesch Kincaid grade level	7.83	5.95	11.05	7.83	11.16	11.22

After removing the two variables the correlation matrix was as given in Figure 2. Despite the screening of data, the correlation matrix was not a positive definite. Further, we looked into highly correlated variables ($R > 0.9$) that might be causing the issue and identified four variables with high correlation ((NA:NR) - 0.989, (N1:MR, NR) - 0.909, (N1:MR, NA) - 0.927). However, removing these variables consistently did not make the matrix to be a positive definite one to carry out the PCA. The outcome was similar when applied to *model transformation* tasks as well. A possible reason for this might have been the less variance among the collected data points due to having similar questions along with insufficient data points.

Correlation Matrix ^{a,b}								
		Number_of_E	Number_of_CA	Number_of_R	Number_of_A	Number_of_MNR	Number_of_1MR	Readability_FKGL
Correlation	Number_of_E	1.000	.372	.778	.775	.433	.485	.430
	Number_of_CA	.372	1.000	.246	.284	.215	.120	-.519
	Number_of_R	.778	.246	1.000	.989	-.122	.909	.324
	Number_of_A	.775	.284	.989	1.000	-.187	.927	.366
	Number_of_MNR	.433	.215	-.122	-.187	1.000	-.525	-.082
	Number_of_1MR	.485	.120	.909	.927	-.525	1.000	.312
	Readability_FKGL	.430	-.519	.324	.366	-.082	.312	1.000
a. Determinant = .000								
b. This matrix is not positive definite.								

Figure 2 Correlation matrix between the complexity metrics for model creation tasks after screening

Considering the limitation we had in conducting a PCA to identify the components to develop a predictive model to measure the difficulty of database modelling tasks, we decided to look into it further by applying a well-known feature selection technique in machine learning to achieve the same. Nonetheless, the outcome remained the same with the data in hand.

Appendix F

Copyright Clearance for Figures and Tables

1. Figure 4.1: ©Leslie Owen Wilson. Reprinted with the permission of the author. The permission provided to use the image is provided below:

Leslie Wilson <thesecondprinciple@gmail.com>      
To: Asanthika Imbulpitiya Thu 23/03/2023 8:09 AM

First off, mega-congrats at being close to the completion of your program!

Secondly, if you read my fine print on the [usage page](#) there is a caveat for non-commercial use of my materials that clearly states, "Because I believe that we can actively work toward creating a better, kinder world, users are duty bound to commit at least one random act of beauty or kindness for each concept used — no kidding!!!"

As long as you agree to those terms, we are good and you have filled my requirement, so use away.

And while I do not know how seriously New Zealander engineering/mathematical types take the concept of karma, I do know that this old lady takes it quite seriously. So, go out in the world and create something wonderful on your way. But most of all remember that, no matter what happens, **you** are always in charge of your own attitude. Be joyful!

And again congratulations.

Namaste,

Leslie

Leslie Owen Wilson, Ed.D.
Professor Emerita, School of Education
University of Wisconsin – Stevens Point
e-mails: thesecondprinciple@gmail.com
Website: <http://thesecondprinciple.com>

Asanthika Imbulpitiya     
To: thesecondprinciple@gmail.com Tue 21/03/2023 2:04 PM

Dear Leslie,

I'm a Doctoral student at Auckland University of Technology and am writing a thesis on "*Learn to Model Relational Databases: Understanding the Novice Database Designer*" for a Doctor of Philosophy in Computer and Information Sciences.

I am writing to request permission for the following work, which you hold the copyright, to be included in my thesis:
Figure: Diagram 1.1, Bloom's Taxonomy Revised
From: <https://thesecondprinciple.com/essential-teaching-skills/blooms-taxonomy-revised/>

A digital copy of the thesis will be made available online via the University's digital repository [Tuwhera](#). This is an open-access research repository for scholarly work intended to make research accessible to as wide an audience as possible. The material will be fully and correctly referenced per your instructions.

If you agree, I would be very grateful if you would email me so I can add your permission to my thesis.

Thank you for your assistance and I look forward to hearing from you.

Best Regards,
Asanthika

Asanthika Imbulpitiya MISM, BSc (Hons)
PhD Candidate / Research Assistant / Teaching Assistant
Software Engineering Research Centre
School of Engineering, Computer & Mathematical Sciences

2. Figure 4.2: CC-BY CC0 1.0 indicates no copyright, and the material is in the public domain. Appropriate credit is provided with the figure.

3. Table 4.1 – 4.3 and Figure 4.3 – 4.6: © 2021 Copyright held by the owner/author(s). Reprinted from the author's own published work. The permission to reproduce their own work published in the ACM digital library is given below. Author rights are explained here: <https://authors.acm.org/author-resources/author-rights>.

Reuse

Authors can reuse any portion of their own work in a new work of their own (and no fee is expected) as long as a citation and DOI pointer to the Version of Record in the ACM Digital Library are included.

- Contributing complete papers to any edited collection of reprints for which the author is not the editor; requires permission and usually a republication fee.
- Authors can include partial or complete papers of their own (and no fee is expected) in a dissertation as long as citations and DOI pointers to the Versions of Record in the ACM Digital Library are included. Authors can use any portion of their own work in presentations and in the classroom (and no fee is expected).
- Commercially produced course-packs that are sold to students require permission and possibly a fee.

4. Table 5.2: Copyright © 2007 Elsevier B.V. Reproduced with Elsevier's permission, the clearance licence is given below.

02/06/2022, 11:14

RightsLink - Your Account

ELSEVIER LICENSE TERMS AND CONDITIONS

Jun 01, 2022

This Agreement between Ms. Asanthika Imbulpititiya ("You") and Elsevier ("Elsevier") consists of your license details and the terms and conditions provided by Elsevier and Copyright Clearance Center.

License Number	5320450952006
License date	Jun 01, 2022
Licensed Content Publisher	Elsevier
Licensed Content Publication	Data & Knowledge Engineering
Licensed Content Title	Defining and validating metrics for assessing the understandability of entity-relationship diagrams
Licensed Content Author	Marcela Genero, Geert Poels, Mario Piattini
Licensed Content Date	Mar 1, 2008
Licensed Content Volume	64
Licensed Content Issue	3
Licensed Content Pages	24
Start Page	534
End Page	557
Type of Use	reuse in a thesis/dissertation
Portion	figures/tables/illustrations
Number of figures/tables/illustrations	1
Format	both print and electronic
Are you the author of this Elsevier article?	No
Will you be translating?	No
Title	Learn to model relational databases: Understanding the novice database modeller
Institution name	Auckland University of Technology
Expected presentation date	Mar 2023
Order reference number	1988_1
Portions	Table 2 on page 539
Requestor Location	Ms. Asanthika Imbulpititiya 12k/113 Vincent Street Auckland Central, Auckland 1010 New Zealand Attn: Asanthika Imbulpititiya
Publisher Tax ID	GB 494 6272 12
Billing Type	Invoice
Billing Address	Asanthika Imbulpititiya 12k/113 Vincent Street Auckland Central, New Zealand 1010 Attn: Asanthika Imbulpititiya
Total	0.00 AUD

5. Figure 7.2 and Figure 7.3: © 2021 Copyright held by the authors and publication rights licensed to ACM. Reprinted from the author's own published work. The clearance license is given below:



Marketplace

This is a License Agreement between Asanthika Imbulpitaya ("User") and Copyright Clearance Center, Inc. ("CCC") on behalf of the Rightsholder identified in the order details below. The license consists of the order details, the Marketplace Permissions General Terms and Conditions below, and any Rightsholder Terms and Conditions which are included below. All payments must be made in full to CCC in accordance with the Marketplace Permissions General Terms and Conditions below.

Order Date	25-May-2023	Type of Use	Republish in a thesis/dissertation
Order License ID	1358346-1	Publisher	Association for Computing Machinery, Inc
ISSN	2153-2184	Portion	Chart/graph/table/figure

LICENSED CONTENT

Publication Title	ACM Inroads	Country	United States of America
Author/Editor	Association for Computing Machinery.	Rightsholder	ACM (Association for Computing Machinery)
Date	01/01/2010	Publication Type	Journal
Language	English		

REQUEST DETAILS

Portion Type	Chart/graph/table/figure	Distribution	Worldwide
Number of Charts / Graphs / Tables / Figures Requested	2	Translation	Original language of publication
Format (select all that apply)	Electronic	Copies for the Disabled?	No
Who Will Republish the Content?	Academic institution	Minor Editing Privileges?	No
Duration of Use	Current edition and up to 5 years	Incidental Promotional Use?	No
Lifetime Unit Quantity	Up to 499	Currency	AUD
Rights Requested	Main product		

NEW WORK DETAILS

Title	Learn to Model Relational Databases: Understanding the Novice Database Modeller	Institution Name	Auckland University of Technology
Instructor Name	Prof. Jacqueline Whalley	Expected Presentation Date	2023-08-31

Appendix G

Learning Outcomes of the INFS601 and COMP508

The learning outcomes the students should be able to achieve with the completion of the are the INFS601 course are as follows¹⁰:

1. Explain the importance of data, and the difference between file management and database management systems.
2. Construct conceptual data models by applying conceptual design methodologies, using Entity Relationship Modelling.
3. Apply the relational model and mappings from conceptual designs, in particular,normalisation.
4. Implement a relational database.
5. Demonstrate knowledge of SQL by designing, implementing and executing complex queries for realistic business cases.
6. Understand the basic concepts of transactions management and concurrency control.

The learning outcomes the students should be able to achieve with the completion of the are the COMP508 paper are as follows¹¹:

1. Understand the fundamental concepts of relational databases.
2. Understand the conceptual, logical, and physical design processes of relational databases
3. Design and implement a relational database.
4. Define, manage and manipulate data from a relational database (SQL).
5. Understand the concepts of transaction management

¹⁰ Taken from the INFS601 course descriptor.

¹¹ Taken from the COMP508 course descriptor.

Appendix H

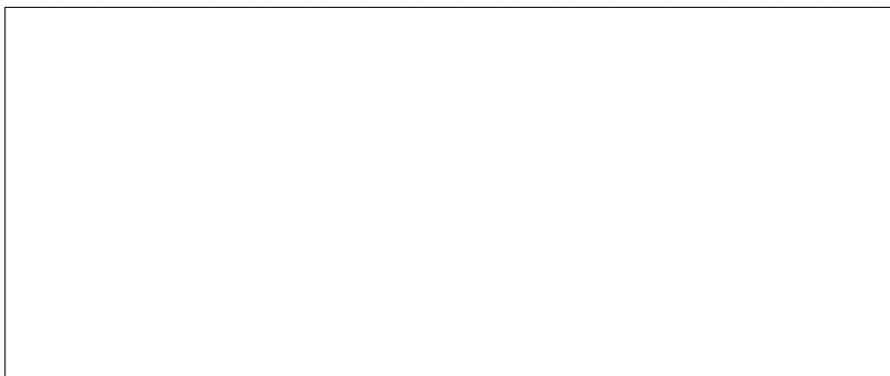
Research Instrument – Case Study

Case Study

The careerAcademy is a major institute that runs training courses for professionals. They wish to develop a database to store all the details of their institute including the trainees, instructors, courses and the classes they offer. After an interview with the institute's manager, the following requirements were identified and the data to be stored are as follows:

Q1. Consider the given below section from the case study and identify all the likely entities, attributes, and relationships.

Trainees: It is necessary to store details about all the trainees who participate in the courses. Each course participant is identified by their trainee ID and this is stored along with their name (composed of first name and last name), date of birth, gender, address (composed of street number, suburb, city, postcode), contact number. Along with these details the previously acquired qualifications of the trainees should be stored in the database. Since most of the time these trainees are referred to the course by their employers it is also necessary to store the details of the employers including the name of the company, address, and contact number. Each employer can be uniquely identified using the employerID. An employer can refer many employees to a training course and a trainee can be only referred to a course via one employer. However, there can be few trainees who are not referred by an employer.



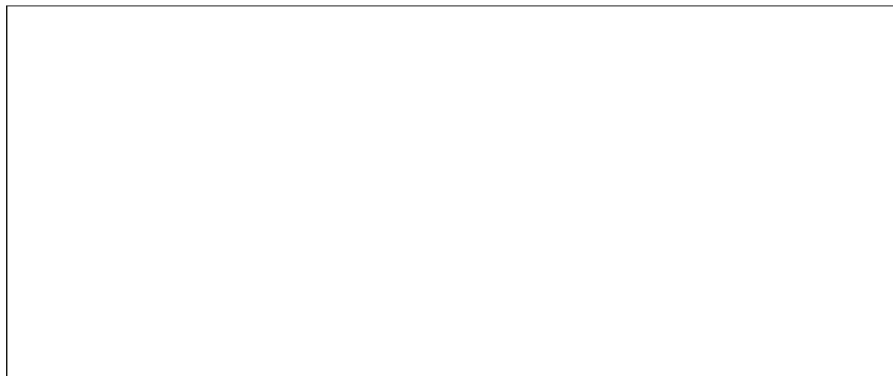
Follow the same process for section 'Courses' and 'Instructors'.

Courses: Each course offered by the institute is stored with a course ID along with the name of the course and the level of the course (eg – An accounting course can have different levels like introduction level, certificate level, diploma level etc). Every time a course is running it may generate several programs (eg – programs can be different sessions the same course is offered depending on the demand) and for each program, the database should store the program ID, start date, and end date. A program can have many trainees registered while a trainee can be registered to only one program at a given time duration. Also, for each program currently in progress, the database should hold

the class id, date of each class, allocated classroom, and the time of the class. A program can generate many classes and each class should belong to one program only.

A large, empty rectangular box with a thin black border, likely intended for a diagram or additional text.

Instructors: Every instructor is given an instructor ID when they joined the institute and along with the ID their name, address, contact number, qualifications, and status (freelance or permanent) need to be stored. An instructor may be assigned to teach any number of programs, while some programs can have more than one instructor assigned to them based on the number of trainees allocated for the program. Some instructors, especially freelance instructors, may not be assigned to teach any programs at a given time.

A large, empty rectangular box with a thin black border, likely intended for a diagram or additional text.

Extra:

For a trainee who is a self-employed professional, the database needs to hold the area of expertise and, if appropriate, the professional title. For a trainee who is an employee, the level and position need to be stored.

Appendix I

Literature Review – Author Collaboration

Author network analysis

Author network analysis was designed to better understand researcher collaborations within the database education research community and to get insight into what type of work they've collaborated on during the past three decades. One popular way to research how people are connected is by conducting a social networking analysis involving developing a structure with actors (people, groups, organisations) and ties (relationships) between two actors. These structures are visualised using network graphs presenting actors as *nodes* and the connected relationships between two nodes as an *edge* indicating the collaboration between the nodes. A Scientific Collaboration Network (SCN) is a type of network used to elaborate the authorships and scientific collaborations among researchers (Newman, 2001). Several researchers in computing education have utilised SCNs to discuss the authorship collaborations among the CSR community (Simon, 2020a; Zhang et al., 2021).

The first step in building the SCN was to define the nodes and edges to feed into the tool used for conducting the network analysis. In this analysis, the authors were represented as nodes and the author name, affiliations and countries were collected to conduct the analysis. Edges represented the collaboration among the authors, and based on their collaboration frequency, a weight was provided (e.g., if two authors have co-authored in only one paper, the weight was 1). The data retrieval and pre-processing were done manually, considering the number of papers. One of the main challenges during the pre-processing of data was matching author names written differently. For example, the names “Storey, V” and “Storey, V.C” were matched together, and so as “Wang, Jeff” and “Wang, J”. There was a rare occurrence of two authors sharing the same surname and initial, “Wang, J,” and based on the time of publication and affiliation countries, they were entered as two different authors. Once the data was processed, Gephi, an open-source software tool (Bastian et al., 2009), was used to conduct the network analysis.

Database education research collaborations

A co-authorship network was developed to understand the research collaborations among the database education researchers. Figure 1 illustrates the co-authorship network for the identified authors from the period 1990 to 2022. It used a weighted undirected network model where the edges are created when two authors have published together at least once, and the weight of an edge is proportional to the frequency of their publications. Further, the size of the nodes (authors) varies according to their degree count, which is the count of authors connected with a specific author, and all similar degree nodes are

indicated in a specific colour. For example, if the degree of a node is higher, the node's size is bigger, and the node's colour is darker and vice-versa. This co-authorship network is composed of 355 nodes (authors) and 425 edges. The average degree was 2.388, meaning that within the observed period of literature, a given author had approximately collaborated with atleast two authors in a published paper. From the data set, it appeared that author “Mitrovic, A” has the highest degree means they published the highest number of collaborations with other authors in database education among the existing authors. Approximately 11% of authors had a degree of 0, where they had papers with 0 collaborations.

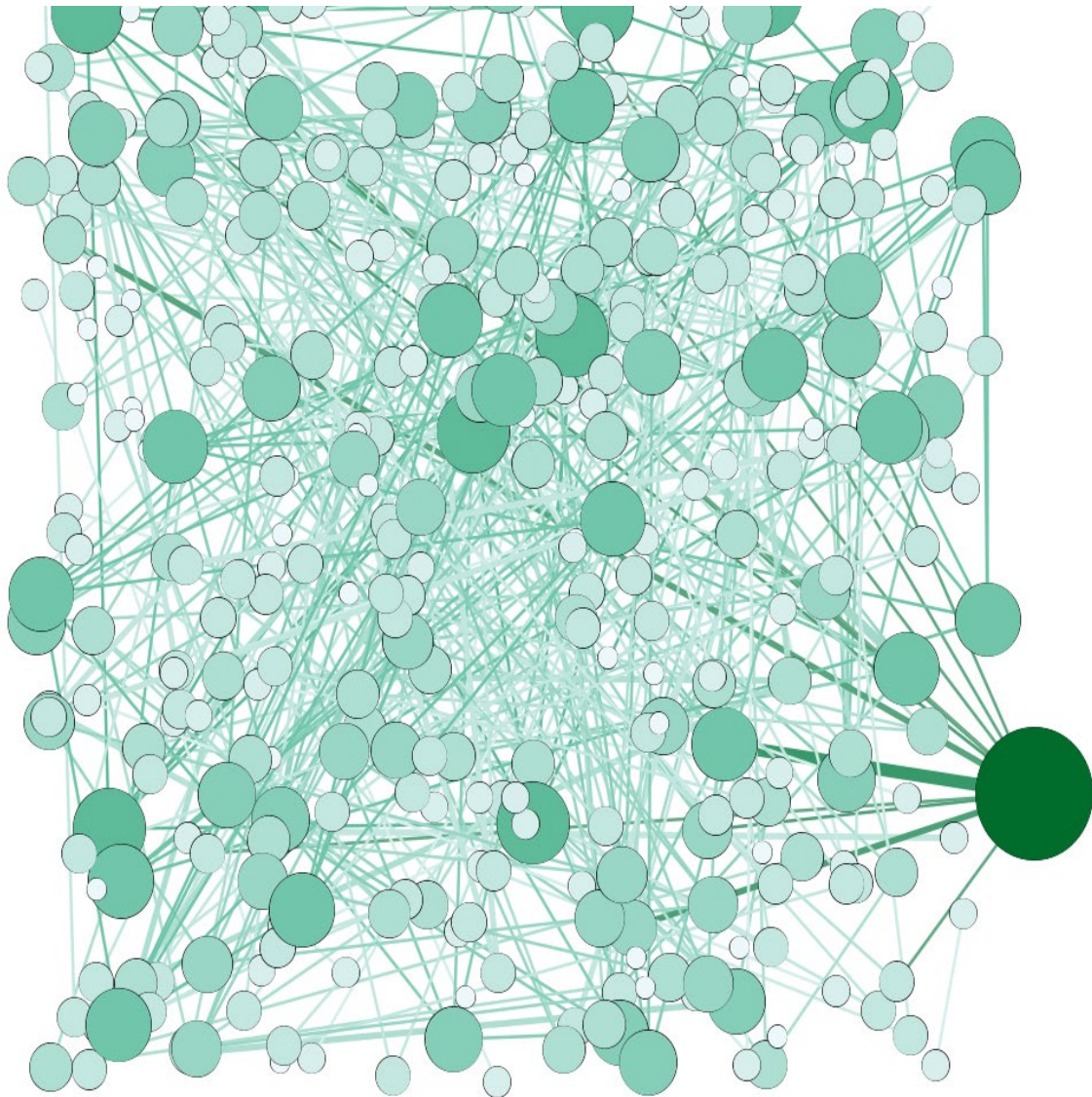


Figure 1 Co-authorship network.

In a social network, a community is a group of people who interact with each other frequently and detecting or identifying these groups is known as community detection. Gephi’s modularity class algorithm was used to identify distinct communities within the data set. Modularity is a measure of the

structure and strength of a graph, measuring the density of connections within a module or a community (Blondel et al., 2008). Before identifying the communities, it's important to position the nodes in the network in relation to each other for better visualisation. Gephi provides several layout algorithms to spread the nodes by calculating the level of attraction and repulsion for each pair of nodes in the network. Force Atlas layout algorithm was used to better visualise the network indicated in Figure 2. Then the the modularity algorithm was applied and it detected 130 communities or clusters. Among them, the densest community, community number 9, contained 3.93% of the nodes in the total network. The other main community identified via the modularity algorithm was community number 43 (3.37% of the total nodes). Despite being ranked 4th (containing only 1.97% of the nodes) based on the modularity algorithm, community number 13 appeared to have multiple collaborations worth looking at. Interestingly, all these identified communities were isolated clusters indicating that the authors collaborated with the same co-authors they have worked with and haven't collaborated beyond their affiliations or countries. And that can explain the larger number of smaller communities identified compared to the larger communities. Figure 2 provides the earlier presented co-authorship network with the identified communities.

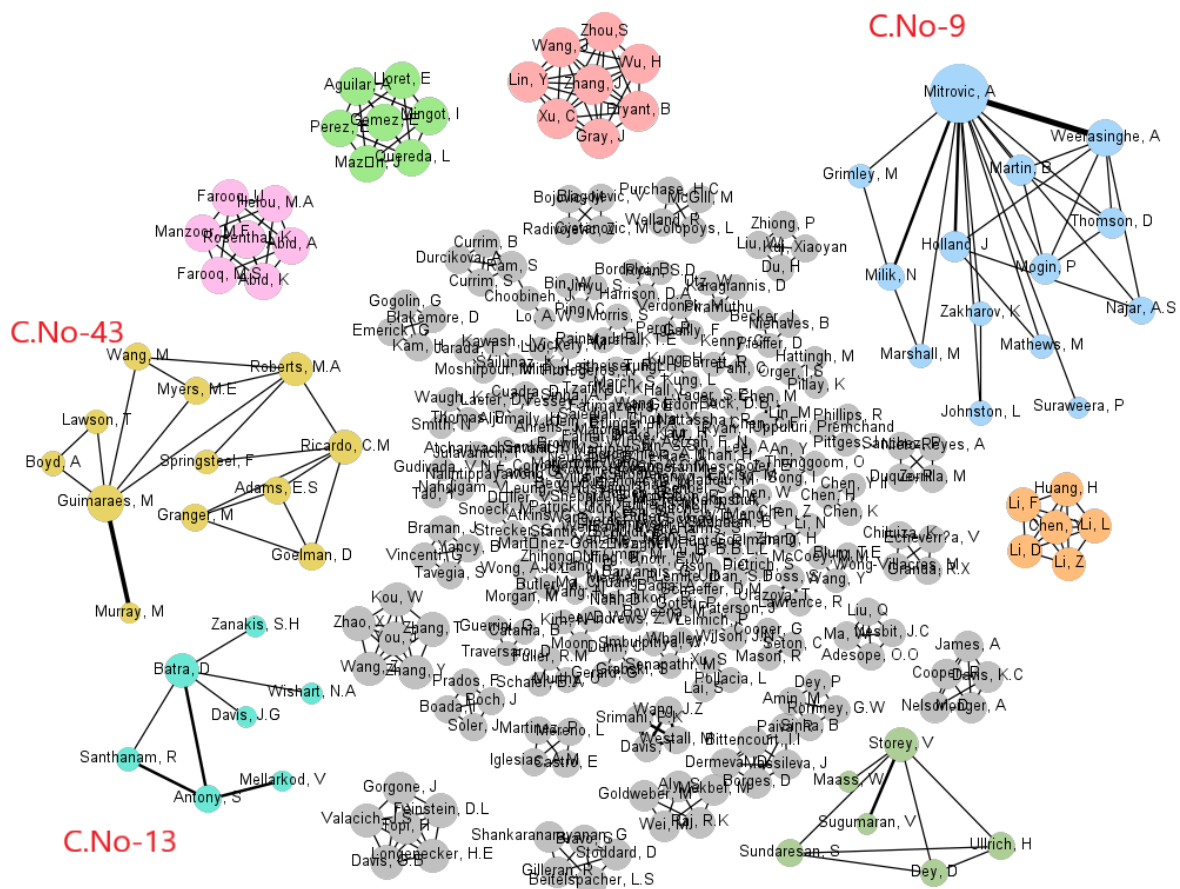


Figure 2 Co-authorship network with detected communities Note. C.No indicates the community number provided by the Gephi tool.

The three clusters identified during the community detection were further analysed based on the published papers' themes (represented as edges) and the authors' affiliated countries (represented as nodes) –please see Figure 3. The largest identified community (community number 9) was a group of authors belonging to the same lab in a New Zealand university however, they had one collaboration with an author affiliated with Australia, but in a closer inspection, it was identified that the author was previously a member of the same research lab. So, all their collaborations were within their own group in the country, and most of their publications were on tools, which is not surprising since their lab is focused on research related to intelligent tutoring systems. There was one publication related to pedagogy. However, it was conducted utilising one of their ITSs. Compared to this cluster, the other two clusters are formed by the authors collaborating in the USA. The authors collaborating in the cluster identified by number 43 have mainly published papers related to course design. However, a few authors have collaborated on publishing articles related to tools and resources within the cluster. In cluster 13, all the authors are from the USA, while there was one collaboration with a co-author from New Zealand. This collaboration was done in a very early publication back in 1992 however it appeared they haven't collaborated after that, at least not within the articles identified in this study. Most of the collaborations within this community were related to tools and resources, and there were few articles related to assessment and pedagogy.

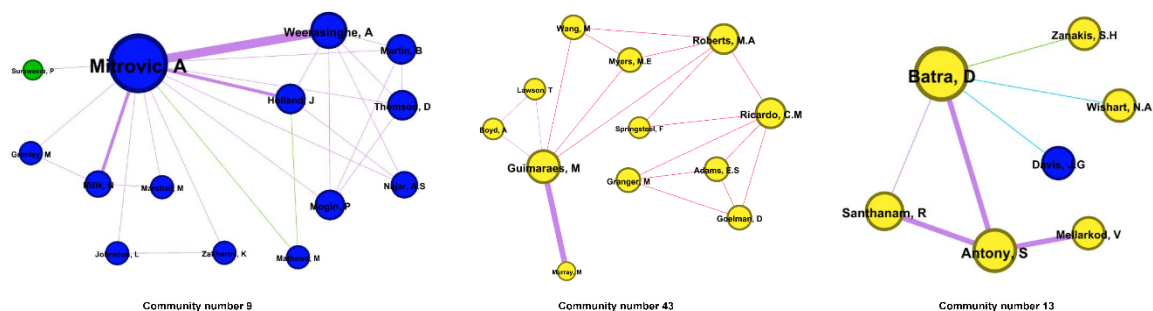


Figure 3 Individual communities identified from the co-authorship network.

Note. The node colour represents the countries, and the edge colour represents the theme of the publication.

Node		Edge	
	New Zealand		Tools and Resources
	Australia		Course design
	USA		Pedagogy
			Assessment

Appendix J

Questions used in the software metrics analysis

Model creation questions

Question 1

Question 1 Develop a logical entity relationship diagram (ERD)

[20 marks]

North Island Scenic Tours (NIST) provides guided tours to groups of visitors to Rotorua City. In recent years, NIST has grown quickly and is having difficulty keeping up with all the various information needs of the company. The company's operations are as follows.

- NIST offers many different tours. Each tour is identified by a tour number, the tour name, approximate length (in hours), and fee charged is needed. Guides are identified by an employee ID, but the system should also record a guide's name (first and last name), home address (street, city, postcode), and date of hire. Guides take a test to be qualified to lead specific tours. It is important to know which guides are qualified to lead which tours and the date that they completed the qualification test for each tour. A guide may be qualified to lead many different tours. A tour can have many different qualified guides. New guides may or may not be qualified to lead any tours, just as a new tour may or may not have any qualified guides. Tours are identified by tour number however the system should also keep the record of tour name, length and fee.
- Every tour must be designed to visit at least three locations. For each location, a name, type, and official description are kept. Some locations (e.g. the Te Puia) are visited by more than one tour, while others (e.g. Rotorua Lake) are visited by a single tour. All locations are visited by at least one tour. Each tour covers multiple locations and the order in which the tour visits each location should be tracked as well.
- When a tour is actually given, that is referred to as an "outing." NIST schedules outings well in advance so they can be advertised and so employees can understand their upcoming work schedules. Outings are identified by outing number. A tour can have many scheduled outings, although newly designed tours may not have any outings scheduled. Each outing is for a single tour and is scheduled for a particular date and time. All outings must be associated with a tour. All tours at NIST are guided tours, so a guide must be assigned to each outing. Each outing has one and only one guide. Guides are occasionally asked to lead an outing of a tour even if they are not officially qualified to lead that tour. Newly hired guides may not have ever been scheduled to lead any outings. Tourists, called "clients" by NIST, pay to join a scheduled outing. For each client, a client identifier, first name, last name, address (street, city, postcode), email address, and telephone number are recorded. Clients may sign up to join many different outings, and each outing can have many clients. Information is kept only on clients who have signed up for at least one outing, although newly scheduled outings may not have any clients signed up yet.

Based on the provided information above, use Visual Paradigm to develop a **logical** ERD that is in 3NF using **Crow's foot notation**. Clearly state any assumptions you make.

- (a) Identify all possible entities and relationships. Name all relationships.
- (b) Resolve any many-to-many relationships.
- (c) Identify (i) strong and weak entities and (ii) identifying and non-identifying relationships.
- (d) Identify the main attributes in each entity including all primary and foreign keys.
- (e) Identify the Cardinality and Participation (optional and mandatory dependencies) for each relationship.

Question 2

Question 4

[20 marks]

Please carefully analyse the transcript of an interview between an external database designer (DD) and the business owner (BO) of company ABC given below.

DD: *Could you please briefly introduce the business of your company?*

BO: *Our main business covers wallpaper and accessories. We have a few branches and each customer places orders through a branch, and each branch may have many customers. A customer may place many orders, but he or she does not always have to order through the same branch all the time.*

DD: *What customer information would you like to track?*

BO: *We need to record customers' full name, including title, first name and last name, as well as their contact details, e.g., landline or mobile number. Our customers may have one or more accounts, although they may also have no accounts. For each customer's account, we need to record Balance, Last payment date, Last payment amount and Type.*

DD: *Can you explain more about the orders placed by your customers? And what do you need to track in the orders?*

BO: *Sure. Each order is composed of one or more items, and an item may be involved in more than one order. We need to know Order Date and Credit Authorization Status. Each item has Description, Colour, Size, Pattern, and Type.*

DD: *Do you want to track the internal employees' information?*

BO: *Definitely. We have 66 employees and their information including full name (first name and last name), mobile number, date of hire, salary and DOB, is required to be recorded. Each employee works in one and only one branch. An employee may or may not have a direct manager. I have given you pretty much information. Do you have any other questions?*

DD: *I think that's good enough... Oh, one more question. What information about a branch do you need to record?*

BO: *Branch Number and the address, including Street, City, State and Post Code. It just reminds me that each branch has a manager responsible for day-to-day operations.*

DD: *Got it. Thank you!*

Based on the information in the interview transcript above, use Visual Paradigm to create a **logical** Entity-Relationship (ER) Diagram using the **Crow's foot model symbols** and include all attributes. Your diagram must:¹

¹ Please note: You don't need to answer these questions separately. You identify them in the diagram.

- (a) Identify (i) strong and weak entities and (ii) identifying and non-identifying relationships. (8 marks)
- (b) Identify the main attributes in each entity including all primary and foreign keys (5 marks)
- (c) Identify the cardinality and participation (mandatory/optional dependencies) for all the relationships (4 marks)
- (d) Resolve all M:N relationships. (3 marks)

Question 3

Case study



Case Study: APU Photography Database for Online Booking and Ordering System

Database Project Assignment

You have been invited to join the project team that will be involved in developing a computerized booking and ordering system for APU Photography Limited. Your job will be to model the data to be stored in the database that will be created for the new system.

APU invited the project team to a meeting to discuss their requirements. Saba Orange, the founder of APU, and outstanding photographer, started the company in 2015 and has slowly achieved the reputation of being Auckland's leading portrait photographer. Her profile is such that appointments for her photography sessions must be made at least two months in advance. Saba now employs three office staff, an accountant, and three assistants. The increase in business has forced Saba and her staff to consider implementing an online system to manage the photography session bookings, the ordering of photographs by customers, and the dissemination of digital proofs to the customers.

It is planned to develop an APU web site where bookings for photo sessions and orders for photo prints can be entered. Both bookings and orders will have to be paid at the time of their entry on the webpage in order to be accepted. Delivery of the ordered prints is promised within two weeks of ordering. The method of collection of the ordered photographs, whether picked up from the studio, or couriered, will be chosen by the people who place the orders on the website.

Through a series of questions from the team, the following information was obtained about the data that should be stored in the system:

To book a photography session on the web page, the customer will select an available date and session time from charts displayed. Session times currently last 75 minutes, with 15 minutes break between sessions. For each weekday the session times are as follows: 8.30 am – 9.45 am; 10.00 am – 11.15 pm; 11.30 am – 12.45 pm; 1.30 – 2.45 pm; 5.30 pm – 6.45 pm. However, session times may change in the future. Photography sessions are carried out in the APU studio, or on a location of the customer's choice, where location can be at a local beach, park, café, home etc.,. Currently a customer is charged \$250.00 for a daytime session and \$275.00 for an early evening session in the studio, or \$400.00 on a location. It is possible that this charge could also change in the future, with individual sessions all having individual charges. Assume that session times will always be constant for all days in the week.

The web page must prompt the customer as to the reason for the photographic session: graduation, wedding, baby etc. The customer may enter a different reason if their purpose is not included in the webpage list. If customers cannot attend their booking, they must cancel on the web page at least three days in advance in order to receive a refund.

Whenever a customer makes a booking on the web page, they must enter their customer details. A new customer will automatically be assigned an account number, or a returning customer can register using their original account number and password. It is not necessary for you to model the password. The customer must supply details of their name (first name, last name for a person, or organisation name if not an individual person). If the customer is a person, a contact phone number and email address must be supplied. For an organisation their contact person's details must be stored (first name, last name, phone number, email address). A contact person for an organisation may change over time. For each new customer the billing address must be supplied (street address, suburb, city, postcode, country).

Once the photography session is completed, the digitized photographs will be stored in a database table, identified by the session date and start time, and each photo's unique number for that session. Once stored, the customer will be emailed with details of how to access the photos online.

Saba Orange sometimes extracts sections of her photographs and assigns them individual numbers and stores them in their own right. In the database it will be necessary to be able to relate the extracted photographs to the original photograph from which they were sourced.

Customers will order their printed photographs on the webpage. Each order will automatically receive a unique order number, and the date/time of the order acceptance. Customers will be able to view their photos at the time of ordering and select each required photo by clicking on its thumbnail photo. For each selected photo the customer must choose a photo colour (e.g. black and white, sepia, colour etc), size, whether the photos are to have a glossy or matte finish, and specify the number of copies required. It is possible that the same photos will be selected in differing combinations of colour, size, and finish, and that different quantities may be ordered for each combination. Each photo combination of colour, size and finish may have its own price, which may change over time.

The orderer must also indicate whether the finished photographs will be collected from APU's studio or be couriered. If the order is to be couriered, a destination postal address must be entered if it differs from the billing address. The destination for the order may be a person or an organisation with details as described previously for customers. For each destination country, APU charges a fixed postage amount. The postal charge set for each country must be stored in the database. The postal charge may change over time.

The total cost of the order, including the cost of the prints and postal charge, if any, will be calculated by the system, displayed on the webpage, and the customer must pay before the order is accepted. Thus, any booking or order will be paid at the time of recording on the webpage. The handling of the payment will be carried out by a third party so payment details will not be stored in the database. However, a booking or order will only be validated if the third party approves the payment. Thus, only paid bookings and orders will be stored in the database.

Question 2 Construct a logical Entity Relationship Diagram (ERD)**[20 marks]**

Based on the business rules developed in Task 1, construct a **logical** ERD for the case study using the Visual Paradigm tool.

- Identify all attributes in each entity including all primary and foreign keys (any composite and multivalued attributes must be transformed according to the rules of the relational model).
- All relationships must be named (i.e. Verb phrases must be shown for all relationships)
- If super/subtypes are used, indicate whether the super-subtype relationships are disjoint or overlapping, and fully or partially specialised (complete or incomplete).
- Identify the Cardinality and Participation for each relationship.
- Resolve all M:N relationships.
- Describe any assumptions you have made in a text note in the diagram.

Question 4

Question 1 Develop a logical entity relationship diagram (ERD)

[20 marks]

North Island Scenic Tours (NIST) provides guided tours to groups of visitors to Rotorua City. In recent years, NIST has grown quickly and is having difficulty keeping up with all of the various information needs of the company. The company's operations are as follows.

NIST offers many different tours. For each tour, the tour name, approximate length (in hours), and fee charged is needed. Guides are identified by an employee ID, but the system should also record a guide's name (first and last name), home address (street, city, postcode), and date of hire. Guides take a test to be qualified to lead specific tours. It is important to know which guides are qualified to lead which tours and the date that they completed the qualification test for each tour. A guide may be qualified to lead many different tours. A tour can have many different qualified guides. New guides may or may not be qualified to lead any tours, just as a new tour may or may not have any qualified guides. Tours are identified by tour number however the system should also keep the record of tour name, length and fee.

Every tour must be designed to visit at least three locations. For each location, a name, type, and official description are kept. Some locations (e.g. the Te Puia) are visited by more than one tour, while others (e.g. Rotorua Lake) are visited by a single tour. All locations are visited by at least one tour. Each tour covers multiple locations and the order in which the tour visits each location should be tracked as well.

When a tour is actually given, that is referred to as an "outing." NIST schedules outings well in advance so they can be advertised and so employees can understand their upcoming work schedules. Outings are identified by outing number. A tour can have many scheduled outings, although newly designed tours may not have any outings scheduled. Each outing is for a single tour and is scheduled for a particular date and time. All outings must be associated with a tour. All tours at NIST are guided tours, so a guide must be assigned to each outing. Each outing has one and only one guide. Guides are occasionally asked to lead an outing of a tour even if they are not officially qualified to lead that tour. Newly hired guides may not have ever been scheduled to lead any outings. Tourists, called "clients" by NIST, pay to join a scheduled outing. For each client, the first name, last name, address (street, city, postcode), and telephone number are recorded. Clients may sign up to join many different outings, and each outing can have many clients. Information is kept only on clients who have signed up for at least one outing, although newly scheduled outings may not have any clients signed up yet.

Based on the provided information, use Visual Paradigm for UML to develop a logical ERD that is in 3NF. Use the Crow's foot notation. Clearly state any assumptions you make.

- a) Identify all possible entities and relationships. Name all relationships in both directions, with the exception that the relationship name may be omitted on a relationship travelling from an associative or bridging entity [8 entities = 2 marks]
- b) Resolve any many-to-many relationships. [2 marks]
- c) Identify (i) 5 strong (2.5) and 3 weak entities (1.5) and (ii) 5 identifying (3) and 2 non-identifying relationships [5 marks]
- d) Identify the main attributes in each entity including all primary and foreign keys [attributes – 3 marks, primary and foreign keys 4 marks] [7 marks]
- e) Identify the connectivity (Cardinality and Participation) for each relationship. [4 marks]

Question 5

Question 4

[20 marks]

KiwiVision is a nonprofit organisation that provides aid to people after natural disasters.

- Individuals volunteer their time to carry out the tasks of the organisation. For each volunteer, their name, address, and telephone number are stored. Each volunteer may be assigned to several tasks during the time that they are doing volunteer work, and some tasks require many volunteers. A volunteer can exist without being assigned to any task. It is possible to have a task to which no one has been assigned as yet! When a volunteer is assigned to a task, the start time and end time of that assignment must be recorded.
- For each task, there is a task code, task description, task type, and task status. For example, there may be a task with a code of “101,” a description of “prepare 500 packages of basic medical supplies,” a type of “packing,” and a status of “open.”
- For all tasks of type “packing,” there is a packing list that specifies the contents of the packages. There are many different packing lists to produce different packages, such as basic medical packages, child care packages, food packages, etc. Each packing list has a packing list ID number, packing list name, and packing list description, which describes the items that ideally go into making that type of package. Every packing task is associated with only one packing list. A packing list may not be associated with any tasks, or may be associated with many tasks. Tasks that are not packing tasks are not associated with any packing list.
- Packing tasks result in the creation of packages. Package details such as the identification number for each package, the date the package was created, and total weight of the package are recorded. A given package is associated with only one task. Some tasks will not have produced any packages, while other tasks (e.g., “prepare 500 packages of basic medical supplies”) will be associated with many packages.
- The packing list describes the ideal contents of each package, but it is not always possible to include the ideal number of each item. Therefore, the actual items included in each package should be tracked. A package can contain many different items, and a given item can be used in many different packages.
- Each item that the organisation provides contains item ID number, item description, item value, and item quantity on hand. Along with tracking the actual items placed in each package, the quantity of each item placed in the package must also be recorded. For example, a packing list may state that basic medical packages should include 100 bandages, 4 bottles of iodine, and 4 bottles of hydrogen peroxide. However, because of the limited supply of items, a given package may include only 10 bandages, 1 bottle of iodine, and no hydrogen peroxide. The fact that this package includes bandages and iodine needs to be recorded along with the quantity of each included. The organisation can have items donated that have not been included in any package yet, but every package will contain at least one item.

Based on the information above, create a logical Entity-Relationship (ER) Diagram using the Crow’s foot model symbols and include all attributes.

- (a) Identify all possible entities and relationships. (4.5 marks)
- (b) Identify (i) strong and weak entities and (ii) identifying and non-identifying relationships. (6.5 marks)
- (c) Identify the main attributes in each entity including all primary and foreign keys (5 marks)
- (d) Identify the mandatory/optional dependencies for all the relationships (4 marks)

Question 6

Case study

INFS601 Case Study for Database Project Assignment

A major faculty in the AsiaPacific University (APU) has approached you for the design and implementation of a relational database system that will provide information on the courses (programmes) it offers, the academic departments that run the courses, the academic staff and the enrolled students. The system will be used mainly by the students and the academic staff.

The requirement collection and analysis phase of the database design process provided the following data requirements for the *Faculty Database* system.

Each department runs a number of courses. The faculty provides a set of papers (e.g. Programming I, Logical Database Design) used in different courses (e.g. Bachelor of Science, Bachelor of Information Technology). Each course uses a number of papers but not every paper is used. A course is assigned a unique course code and a paper is identified by a unique paper code. A paper can be used in more than one course, and it can be studied by many students. In addition to the paper code, each paper has a unique title, start date, end date, paper resources (e.g. textbooks), and assessment scheme (i.e. assessment and exam marks percentages) are also stored.

Each course is managed by a member of academic staff, and each paper is coordinated by a member of academic staff also. The database should also store each course unique title, and duration (in years).

A student can enrol in one course at a time. Once enrolled a student is assigned a unique student identification number. To complete a course, each student must undertake and pass all the required papers in the course that the student is enrolled in. This requires that the database store the performance (pass or fail) of each student in every paper.

Additional data stored on each student includes student name (first and last), address (town, street, and post code), date of birth, sex, and student financial loan (Yes or No). For emergency purposes the database stores the last name, first name, address (not composite), phone, and relationship of each student next-of-kin. None of the next-of-kin's attributes is unique. Assume that every next-of-kin is a next-of-kin of one student only.

Each department is led by a member of academic staff (e.g. Head of the Department). The database should record the date the staff member started the role as Head of the department. Each department has a name, phone number, fax number, and location (e.g. Z Block). Each department employs many members of academic staff.

A member of academic staff can be the coordinator of at most one course but can be the coordinator of more than one paper. A member of academic staff may not be assigned any of the above-mentioned roles (coordinator, course leader, Head of the Department manager). All members of academic staff teach papers. Every member of academic staff may teach zero, one or more papers, and a paper may be taught by more than one member of academic staff. The database should record the number of hours per week a member of academic staff spends teaching each paper. Each member of academic staff is identified by a unique staff number. Additional data stored on each member of academic

staff includes name (first and last), phone extension number, office number, sex, salary, position (lecturer, or senior lecturer, or Professor, etc.), qualifications, and address (not composite). A member of academic staff works for one department only.

Question 2

Construct a logical Entity Relationship Diagram (ERD) [20 marks]

Based on the business rules developed in Task 1, construct a logical ERD for the case study using the Visual Paradigm tool.

- Identify all attributes in each entity including all primary and foreign keys (any composite and multivalued attributes must be transformed according to the rules of the relational model).
- All relationships must be named (i.e. Verb phrases must be shown for all relationships)
- If super/subtypes are used, indicate whether the super-subtype relationships are disjoint or overlapping, and fully or partially specialised (complete or incomplete).
- Identify the Cardinality and Participation for each relationship.
- Resolve all M:N relationships.
- Describe any assumptions you have made in a text note in the diagram.

Comprehension questions

Question 7 / Question 8

Midterm assessment – individual

Question 1

[15 marks]

A rail transport company wishes to hold details of train services, stations and operators. Details include the following:

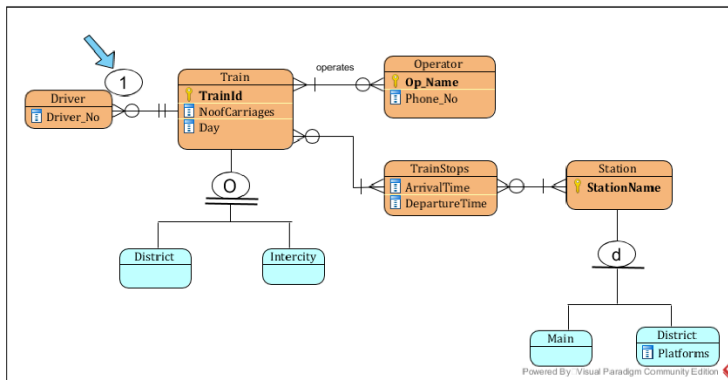
Trains: A train has a unique id code, a collection of days it runs on (Monday, Sunday etc.), and the number of carriages it has. Each train must have at least one driver but can have many drivers that can drive it, and a driver may drive many trains. Each driver is identified by an employee number and a name. A train is designated as either an InterCity or District train though other types of train are possible. Each train must have an operator and not more than one operator. Each operator has a name and a phone number. The operator can operate many trains (or none at all!)

Stations: A station has a name and the number of platforms it has. All stations are classified as either Main

Timetable: A train has a number of stops at stations during its journey. It never visits the same station twice on the same journey. At each stop, the arrival and departure times are recorded. A station has many trains passing through it, often arriving at the same time, and a train visits many stations.

A new database designer has made a first attempt at an EER schema, shown in Figure 1, but there are *at least* 10 mistakes. Examples of mistakes include incorrect cardinality and participation in relations, missing or incorrect attributes, and incorrect constraints (e.g., disjoint, partial).

Identify 7 mistakes by giving a number (e.g., 1, 2). Clearly explain each mistake and what should be the correct answer in the table provided on next page. An example is given for the mistake numbered 1 (pointed by the arrow) and its explanation is in the table next page.



Requirements translation questions

Question 9 / Question 10 (related to the case studies provided in question 3 and question 6)

Question 1

Identify Entities & Develop Business Rules [10 marks]

Identify all entities and relationships in the case study and develop a set of **business rules**. You should follow the syntax given below. Each relationship must be described by two rules, one in each direction.

Each/A/An ENTITY_1 *May/Must* Relationship_Verb_Phrase *number* ENTITY_2

Model transformation questions

Question 11

Question 2 Normalisation

[25 marks]

Examine the relation given below which contains data about students enrolled in various courses during a specific year. Carefully analyse the relation and answer the following questions:

Stud_ID	LName	FName	Crse_ID	Crse_Name	Grade	Lect_ID	Lect_Lname	Lect_Fname	Lect_Office	Semester
001	Rose	Johnson	ENG01	ENGLISH	A	L001	Ruger	John	102	Sem1
001	Rose	Johnson	GER01	GERMAN	B	L002	Findling	Holger	101	Sem2
002	Anthony	Joshua	ENG01	ENGLISH	A	L001	Ruger	John	102	Sem1
002	Anthony	Joshua	GER01	GERMAN	B	L002	Findling	Holger	101	Sem2
003	May	Parker	GER01	GERMAN	A	L002	Findling	Holger	101	Sem2
004	Susanna	Grant	ITA01	ITALIAN	B	L003	Fresco	Luise	103	Sem2

- In **what level of normalisation** is the above relation (Table 1)? Identify the **primary key attributes**. Briefly justify your answer. (3 marks)
- Identify all the dependencies using **a dependency diagram**. (7 marks)
- Using the data shown in Table 1 derive a set of relations in third normal form (3NF). Clearly explain each step of normalisation – 1NF, 2NF, and 3NF. Show all primary keys in each relation with a solid underline and foreign keys with a dashed underline. (10 marks)
- Draw an Entity Relationship Diagram (ERD) from the 3NF relations and include all attributes, primary keys, foreign keys, cardinality, and participation. (5 marks)

Question 12

Question 3

[15 marks]

Table 1 shows data about properties rented to clients in Auckland. It also shows details of the owners who own the property.

Table 1

clientNo	propertyNo	cName	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	PG4	John Key	6 Lawrence St, Auckland	1-Jul-13	31-Aug-14	350	CO40	Tina Murphy
CR76	PG16	John Key	5 Novar Dr, Auckland	1-Sep-14	1-Sep-15	450	CO93	Tony Brunt
CR56	PG4	Alice Stewart	6 Lawrence St, Auckland	1-Sep-12	10-Jun-13	350	CO40	Tina Murphy
CR56	PG36	Alice Stewart	2 Manor Rd, Auckland	10-Oct-13	1-Dec-14	375	CO93	Tony Brunt
CR56	PG16	Alice Stewart	5 Novar Dr, Auckland	1-Nov-15	10-Aug-16	450	CO93	Tony Brunt

- Using the data shown in Table 1 derive a set of relations in **third normal form (3NF)**. Clearly show each step of normalisation – 1NF, 2NF, and 3NF. Show all primary keys in each relation with a solid underline and foreign keys with a dashed underline. (10 marks)
- Draw an **Entity Relationship Diagram (ERD)** from the 3NF relations and include all attributes, primary keys, foreign keys, cardinality, and participation. (5 marks)

Question 13

Question 2 Normalisation

[20 marks]

The Shipping manifest table below shows shipping information of a large manufacturing company. Carefully analyse the table and answer the following questions.

- Which normal form does the report conform to? Briefly explain your answer. [2 Marks]
- Identify all the dependencies using a Functional Dependency Diagram. [8 Marks]
- Using the dependencies identified in (ii) above, convert the report to 3NF relations. Use Visual Paradigm to create an ERD based on the 3NF relations. Include all attributes, primary keys, foreign keys, cardinality and participation. State clearly any assumptions you make. [10 Marks]

Shipment ID:	00-0001	Shipment Date:	01/10/2018
Origin:	Boston	Expected Arrival:	01/14/2018
Destination:	Brazil		
Ship Number:	39	Captain:	002-15 Henry Moore

Item Number	Type	Description	Weight	Quantity	TOTALWEIGHT
3223	BM	Concrete Form	500	100	50,000
3297	BM	Steel Beam	87	2,000	174,000
Shipment Total:					224,000

Question 14

Question 3

[15 marks]

Examine the relation given below (Table 1) which contains data about properties rented to clients in Auckland. It also shows details of the owners who own the property. Carefully analyse the relation and answer the following questions:

Table 1

clientNo	propertyNo	cName	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	PG4	John Key	6 Lawrence St, Auckland	1-Jul-13	31-Aug-14	350	CO40	Tina Murphy
CR76	PG16	John Key	5 Novar Dr, Auckland	1-Sep-14	1-Sep-15	450	CO93	Tony Brunt
CR56	PG4	Alice Stewart	6 Lawrence St, Auckland	1-Sep-12	10-Jun-13	350	CO40	Tina Murphy
CR56	PG36	Alice Stewart	2 Manor Rd, Auckland	10-Oct-13	1-Dec-14	375	CO93	Tony Brunt
CR56	PG16	Alice Stewart	5 Novar Dr, Auckland	1-Nov-15	10-Aug-16	450	CO93	Tony Brunt

- In what level of normalisation is the above relation? Identify the primary key attributes. Briefly justify your answer. (2 marks)
- Identify all the dependencies using a dependency diagram. (4 marks)
- Using the dependencies identified in (b), decompose the relation into a set of 3NF relations. State clearly any assumptions you make. (4 marks)
- Draw an Entity Relationship Diagram (ERD) from the 3NF relations and include all attributes, primary keys, foreign keys, cardinality, and participation. (5 marks)

Question 15

- To further demonstrate your knowledge of normalisation, assume that a proposed (badly structured) relation for the Faculty Database has the structure shown below. Identify the functional dependencies represented in this relation and demonstrate the process of normalising this relation into 3NF relations. [10 marks]

StudentID	Student lname	Student fname	sex	PaperTitle	Paper startDate	performance	flatNo	address
005021	McLeod	Andrew	M	DB Design	13/07/2020	Pass	F001	7 Symonds Street, Auckland
004647	Smith	Anna	F	Program Design	15/07/2020	Pass	F001	7 Symonds Street, Auckland
014670	Owen	Michael	M	Networking	17/07/2020	Fail	F002	24 Vulcane Lane, Auckland
016765	Smith	Anna	F	DB Design	13/07/2020	Pass	F003	24 Vulcane Lane, Auckland
005021	McLeod	Andrew	M	Networking	17/07/2020	Pass	F001	7 Symonds Street, Auckland
004647	Smith	Anna	F	Networking	17/07/2020	Fail	F001	7 Symonds Street, Auckland

Question 16

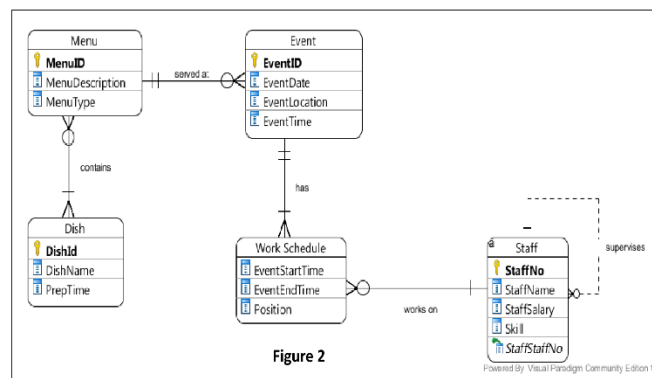
Question 2

[10 marks]

- a) Figure 2 shows a **Conceptual Entity Relationship diagram** for a university dining service. This organisation provides dining services to a major university. Each staff can have many skills: so, the attribute Skill in the Staff relation is a multi-valued attribute.

Please **transform the ER model (in Figure 2) to a logical relational schema (i.e., logical design):**

- List all the functional dependencies. Show all relations including any new relations that need to be created as part of the logical design process. (5 marks)
- Clearly indicate **ALL** primary and foreign keys for each relation. Show primary keys in each relation with a solid underline and foreign keys with a dashed underline. (5 marks)

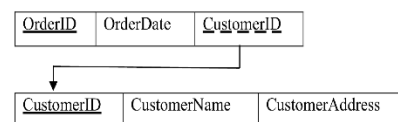


Examples of how to list functional dependencies and how to show relations with primary and foreign keys are given below:

Functional dependencies:

- $\text{OrderID} \rightarrow \text{OrderDate}, \text{CustomerID}, \text{CustomerName}, \text{CustomerAddress}$
- $\text{CustomerID} \rightarrow \text{CustomerName}, \text{CustomerAddress}$

Relations:



Question 17

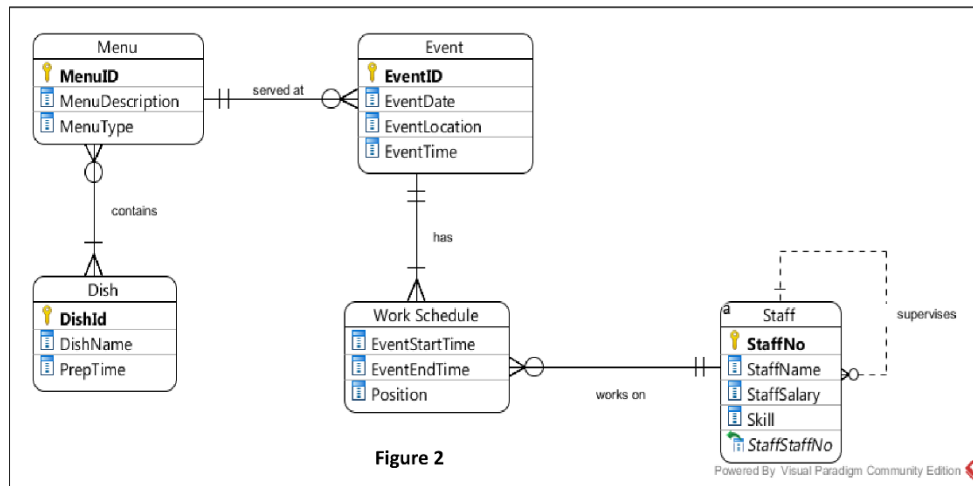
Question 2

[10 marks]

- a) Figure 2 shows a Conceptual Entity Relationship diagram for a university dining service. This organisation provides dining services to a major university. Each staff can have many skills: so, the attribute Skill in the Staff relation is a multi-valued attribute.

Please transform the ER model (in Figure 2) to a relational schema (i.e., logical design):

- Diagram all the functional dependencies. Show all relations including any new relations that need to be created as part of the logical design process. (5 marks)
- Clearly indicate ALL primary and foreign keys for each relation. (5 marks)



Appendix K

Hierarchy among database modelling skills

Based on our preliminary exploration of the literature, no research has looked at the nature of data modelling questions or the possibility of a hierarchy of skills in modelling. Earlier work has established that there is a hierarchy of skills in learning to program (Lopez et al., 2008; Philpott et al., 2007b; J. L. Whalley & Lister, 2009), and intuitively, it makes sense that there is some dependency on building models that describe the technical requirements of software including databases.

This section focused on investigating whether or not such a hierarchy of skills also exists for novices learning database modelling. The conjecture of a task hierarchy was explored using a statistical modelling approach, stepwise regression. Students' performance on modelling tasks was utilised as a variable to establish a pathway model for novice data modelling tasks. Similar methods have been used to develop a hierarchy of skills in novice programming, which investigated the correlative and causative relationships between code reading, writing and tracing tasks (Lopez et al., 2008). This understanding will lead to empirically based design regarding the structure and nature of assessments and learning tasks in the subject area.

The main objective of this analysis was to investigate the relationship between different types of questions and establish the notion of a hierarchy (if exists) among the skills required to complete the questions. The main questions addressed in this analysis are as follows,

1. What are the skills associated with the ability to create a database model?
2. Does the overall student performance account for a notion of hierarchy among the database modelling skills?

As explained in section 5.7, naturally occurring data from the INFS601 course collected in the 1st semester of 2021 (data from 78 participants) was selected to establish a hierarchy of skills among novice database modellers. Based on the existing studies, it was decided to use the final exam questions which were done in a controlled environment, to conduct this analysis. However, due to the change of assessments during the COVID-19 pandemic, instead of the final exam questions, the assignment tasks were provided for the students since no exams were conducted during that time.

The instrument

Here we explored all the relevant assignment tasks related to database modelling. Descriptive statistics were used to understand the distribution of the marks for each question, as indicated in Table 1. Before the analysis, all the assignment tasks were categorised into the pre-determined set of task categories as

Requirements translation, Comprehension, Model transformation – Type 1, Model transformation – Type 2, and Model creation. Further, Table 2 provides simple statistics based on the type of questions.

Table 1: Descriptive statistic for each assessment question.

	N	Assessment	Question Type	Possible Mark	Avg Mark	Avg Mark (%)	1st Quartile	Median	3 rd Quartile
Q1	75	N&LDDA	Model creation	20	16.18	80.9	14.5	18	20
Q2	77	MA	Model creation	20	14.29	71.45	12	14.5	17.5
Q3	74	DBP	Model creation	20	15.77	78.85	13.38	16	18.13
Q4	77	MA	Comprehension	15	14.12	94.13	13	15	15
Q5	73	DBP	Requirement translation	10	8.49	84.9	7.5	8	10
Q6	75	N&LDDA	Model transformation - Type 1	25	20.07	80.28	18	22	25
Q7	73	MA	Model transformation - Type 1	15	9.48	63.2	7	10	12
Q8	70	MA	Model transformation - Type 2	10	8.04	80.4	7	8	10

Note. The acronyms in column 2 are as follows: N&LDDA –Normalisation and logical database design assessment, MA –Midterm assessment, DBP –Database project.

Table 2: Descriptive statistics for type of questions.

	Ranking (average mark)	Question Type	Possible Mark	Average Mark	Average Mark (%)	First Quartile	Median	Third Quartile
	4	Model creation	60	46.24	77.06	39.88	48.5	55.63
1 - Easiest		Comprehension	15	14.12	94.13	13	15	15
2		Requirement translation	10	8.49	84.9	7.5	8	10
5 - Hardest		Model transformation - Type 1	40	29.55	73.88	25	32	37
3		Model transformation – Type 2	10	8.04	80.4	7	8	10

As the first step to understanding the dependency and hierarchy among the novice database design tasks, we identified different modelling tasks based on the previous analysis conducted to understand the nature of the database modelling tasks (Imbulpitiya et al., 2021). The detailed definitions of the identified categories are as follows:

1. Describe – The tasks which require the students to describe, at a recall level, the concepts related to database modelling.
2. Requirement translation -The tasks which require the students to translate a provided scenario/case study into appropriate modelling constructs so that the case study can be designed accordingly.
3. Model transformation - The tasks which require the students to transform a database model from one form to another.
4. Model interpretation/comprehension - The tasks which require the students to read, explain or interpret a provided design.
E.g., Identifying errors and/or inconsistencies such as rule violations in a design.
5. Model evaluation – The tasks which require the students to make judgements/critique a given design(s).
6. Model creation - The tasks which require the students to develop/create a database model (e.g., ERD, EERD, relational schema) based on a provided specification and case study.

The categories mentioned above were identified based on an analysis conducted on previous examination questions. However, due to the assessment shift during the pandemic, we only had assessment tasks to be evaluated, where some of the questions didn't fit into any category.

Describe

These questions are intended for the students to recall or retrieve learnt information such as terms, concepts and modelling notation, explicitly taught in the course, from their long-term memory definitions of concepts. Some of the possible questions we identified from the previous analysis that can be categorised under *Describe* are as follows:

Example 1: Define the terms i) Database and ii) Database Management System.

Example 2: List five important properties of a relation.

As mentioned previously, none of the assessment tasks were suitable to categorise under *Describe*.

Requirement translation (Question 5)

This question required the students to translate a provided case study or a scenario into appropriate modelling constructs, which can be considered the initial step in designing the database model. The following question was presented in the final database project with a case study. The students were required to translate the requirements into business rules by identifying the primary constructs like entities and relationships.

Question 5: Identify all entities and relationships in the case study and develop a set of business rules. You should follow the syntax given below. Each relationship must be described by two rules, one in each direction.

Each/A/An ENTITY_1 May/Must Relationship_Verb_Phrase number ENTITY_2

This question was worth ten marks, and the general performance of the students was good, where the average mark was 8.49 out of 10 (84.9%).

Comprehension (Question 4)

Comprehension type questions required the students to read a provided design and interpret it. In this question, the students were presented with an Extended Entity-Relationship (EER) diagram and the scenario used to develop it. The students were required to identify seven design-level errors in the diagram and justify why they considered it a mistake and what should be the correct way to present it. Students were asked to identify seven errors, while eleven errors were in the design.

Students were awarded 15 marks for each correctly identified design error and the justification. Partial marks were given in the scenarios where the explanation and correction were absent despite identifying the error.

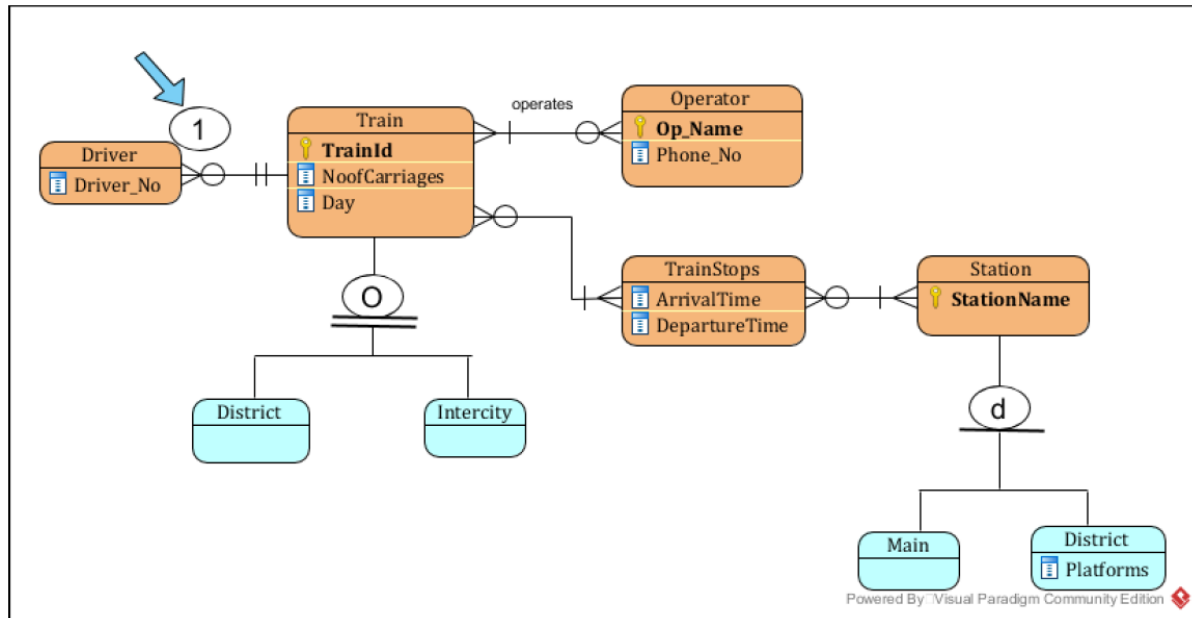
Question 4: A rail transport company wishes to hold details of train services, stations and operators. Details include the following:

Trains: A train has a unique id code, a collection of days it runs on (Monday, Sunday etc.), and the number of carriages it has. Each train must have at least one driver but can have many drivers that can drive it, and a driver may drive many trains. Each driver is identified by an employee number and a name. A train is designated as either an InterCity or District train though other types of train are possible. Each train must have an operator and not more than one operator. Each operator has a name and a phone number. The operator can operate many trains (or none at all!)

Stations: A station has a name and the number of platforms it has. All stations are classified as either Main or District only. There are no other types of stations.

Timetable: A train has a number of stops at stations during its journey. It never visits the same station twice on the same journey. At each stop, the arrival and departure times are recorded. A station has many trains passing through it, often arriving at the same time, and a train visits many stations.

A new database designer has made a first attempt at an EER schema, shown in Figure 1, but there are at least 10 mistakes. Examples of mistakes include incorrect cardinality and participation in relations, missing attributes, and incorrect constraints (e.g. disjoint, partial). Identify 7 mistakes by giving a number (e.g. 1, 2) – an example is given below by naming the first mistake as number 1 (pointed by the arrow). Clearly explain each mistake and what should be the correct answer.



We were surprised by how well students performed in this question, with an average mark of 14.12 out of 15 (94.13%); hence this can be considered the easiest question among all the assessment tasks. Almost all the students demonstrated mastery of the *Comprehension* skills, where 75% of marks are less than or equal to the full mark.

Model transformation (Questions 6,7)

Initially, we defined this category to explain tasks that required the students to transform a database model from one form to another (i.e. conceptual model to a logical model). In these tasks, the students were expected to read and understand the provided model to select and apply the appropriate procedure to transform the model correctly. We identified two main types of transformations 1) relation into an ERD and 2) ERD into a relational schema. Even though all these tasks represented a type of transformation, the applied process was different, leading to a different difficulty level. Hence we divided this category further into two classes as follows.

Model transformation - Type 1

These tasks required the students to transform a provided relation (row set of data) into an ERD, and questions 6 and 7 were categorised under the *model transformation – Type 1* category. These questions expected the students to apply normalisation, which has multiple steps that must be adapted to get to

the outcome. Once the students are provided with the raw data in relation, they are asked to identify the current normal form of the relation and normalise it to the third normal form (3NF) to transform the normalised schema into a logical ERD. This can be considered a step-by-step transformation depending on how the question is presented, but sometimes with the developed skills, students can skip some of the steps to get to the outcome. Considering the nature of the novices students, the questions were asked in different scaffolded levels, as indicated in questions 6 and 7.

Question 6: Examine the relation given below, which contains data about students enrolled in various courses during a specific year. Carefully analyse the relation and answer the following questions:

Stud_ID	LName	FName	Crse_ID	Crse_Name	Grade	Lect_ID	Lect_LName	Lect_FName	Lect_Office	Semester
001	Rose	Johnson	ENG01	ENGLISH	A	L001	Ruger	John	102	Sem1
001	Rose	Johnson	GER01	GERMAN	B	L002	Findling	Holger	101	Sem2
002	Anthony	Joshua	ENG01	ENGLISH	A	L001	Ruger	John	102	Sem1
002	Anthony	Joshua	GER01	GERMAN	B	L002	Findling	Holger	101	Sem2
003	May	Parker	GER01	GERMAN	A	L002	Findling	Holger	101	Sem2
004	Susanna	Grant	ITA01	ITALIAN	B	L003	Fresco	Luise	103	Sem2

a) In what level of normalisation is the above relation (Table 1)? Identify the primary key attributes. Briefly justify your answer.

b) Identify all the dependencies using a dependency diagram.

c) Using the data shown in Table 1 derive a set of relations in third normal form (3NF). Clearly explain each step of normalisation – 1NF, 2NF, and 3NF. Show all primary keys in each relation with a solid underline and foreign keys with a dashed underline.

d) Draw an Entity Relationship Diagram (ERD) from the 3NF relations and include all attributes, primary keys, foreign keys, cardinality, and participation.

The students were awarded 25 marks for this question and the distribution of the marks is 3,7,10,5 marks for subparts a,b,c and d, respectively. The average mark was 20.07 out of 25 (80.28%), where students did exceptionally well considering that this was the first assessment task related to model transformation.

Question 7: Table 1 shows data about properties rented to clients in Auckland. It also shows details of the owners who own the property.

clientNo	propertyNo	cName	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	PG4	John Key	6 Lawrence St, Auckland	1-Jul-13	31-Aug-14	350	CO40	Tina Murphy
CR76	PG16	John Key	5 Novar Dr, Auckland	1-Sep-14	1-Sep-15	450	CO93	Tony Brunt
CR56	PG4	Alice Stewart	6 Lawrence St, Auckland	1-Sep-12	10-Jun-13	350	CO40	Tina Murphy
CR56	PG36	Alice Stewart	2 Manor Rd, Auckland	10-Oct-13	1-Dec-14	375	CO93	Tony Brunt
CR56	PG16	Alice Stewart	5 Novar Dr, Auckland	1-Nov-15	10-Aug-16	450	CO93	Tony Brunt

a) Using the data shown in Table 1 derive a set of relations in third normal form (3NF). Clearly show each step of normalisation – 1NF, 2NF, and 3NF. Show all primary keys in each relation with a solid underline and foreign keys with a dashed underline. (10 marks)

b) Draw an Entity Relationship Diagram (ERD) from the 3NF relations and include all attributes, primary keys, foreign keys, cardinality, and participation. (5 marks)

This question was given to the students in their midterm assessment which was a time-controlled assessment compared to question 6, and it's not as scaffolded even though both the questions are somewhat similar and require the students to apply the same set of steps. In general, the students did poorly in this question compared to the previous one, and the average mark was 9.48 out of 15, with an average percentage of 63.2%. The drop in the average mark despite the similarity of the question might be the fact that it was a time-controlled assessment despite being a take-home assignment.

Model transformation – Type 1 can be considered the hardest type of question for the students, with an average mark of 29.55 out of 40 (73.88%).

Model transformation – Type 2 (Question 8)

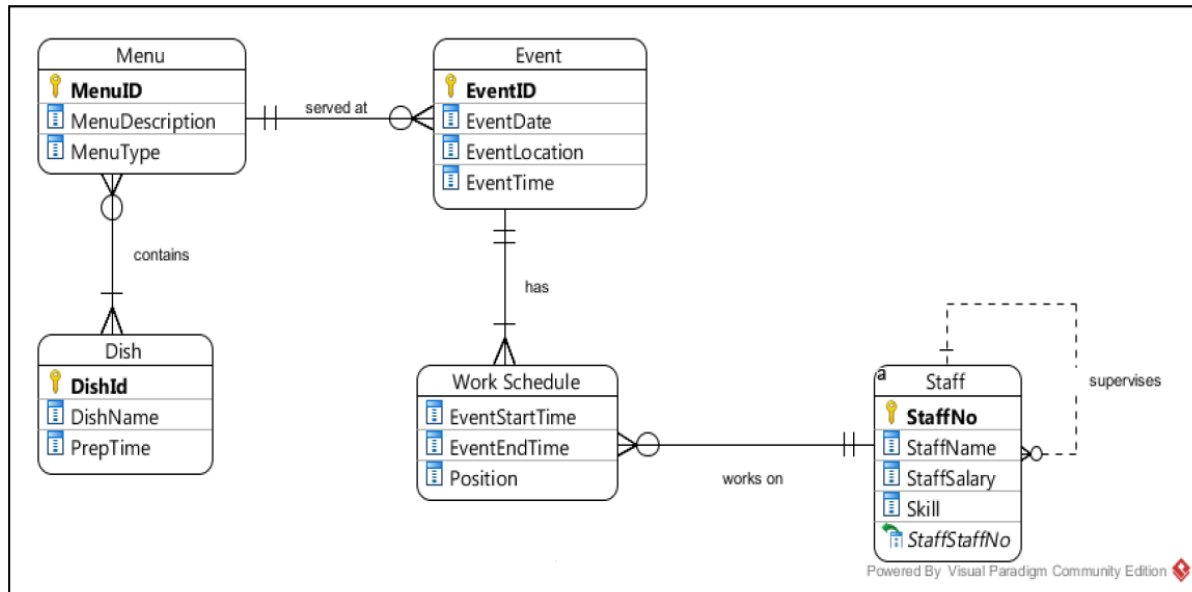
In this question, students were required to transform a provided ERD into a logical relational schema. Interestingly, an example of the transformed model was supplied with the question. Overall, the students performed exceptionally well on this question, where the average mark was 8.04 out of 10 (80.4%). The performance here contrasts with the students' performance of *model transformation – Type 1* questions, despite all these questions, have required the students to transform a model from one form to another. The question under this category is as follows:

Question 8: Below figure shows a **Conceptual Entity-Relationship diagram** for a university dining service. This organisation provides dining services to a major university. Each staff can have many skills: so, the attribute Skill in the Staff relation is a multi-valued attribute.

Please **transform the ER model (in Figure) to a logical relational schema (i.e., logical design):**

(i) List all the functional dependencies. Show all relations including any new relations that need to be created as part of the logical design process. (5 marks)

(ii) Clearly indicate ALL primary and foreign keys for each relation. Show primary keys in each relation with a solid underline and foreign keys with a dashed underline. (5 marks)

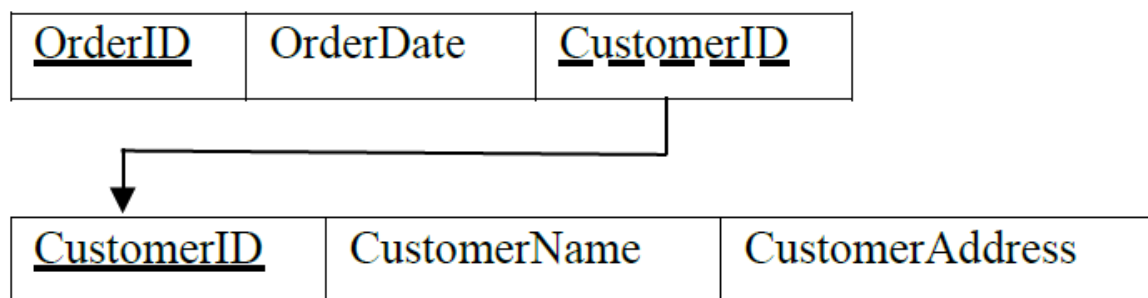


Examples of how to list functional dependencies and how to show relations with primary and foreign keys are given below:

Functional dependencies:

- (i) OrderID → OrderDate, CustomerID, CustomerName, CustomerAddress
- (ii) CustomerID → CustomerName, CustomerAddress

Relations:



Compared to the *model transformation – Type 1* questions, this transformation required the students to follow a different process, which is more like a one-to-one transformation, possibly explaining the difference in the average mark percentages of the students.

Model evaluation

These questions are intended to evaluate a student's ability to judge based on criteria and standards. In this type of question, a student should be able to critique a provided database design, as in the following example:

Example: You have been provided with a problem description and four different ERD diagrams, each presenting a different design solution to the problem. Which ERD diagram gives the best solution to the problem? Explain the reasons that led you to select that particular ERD.

To answer the question, students need to critically assess the diagrams and determine which is the best solution. There may not necessarily be a single correct answer. Instead, such a question intends to evaluate the students' ability to critique, evaluate, and justify their decision. Again this is a type of question we couldn't identify in the assessment tasks.

Model creation (Questions 1-3)

Questions 1 to 3 were all categorised under the *Model creation* category, where the students were required to develop a database model based on the provided scenario. All of the questions classified at this level in the assignments analysed, provided students with a scenario or problem specification (of varying lengths and levels of detail) and asked them to build specified design artefacts. The case studies used for all three questions are given in Appendix _. The questions provided under this category were as follows:

Question 1: Based on the provided information above, use Visual Paradigm to develop a logical ERD that is in 3NF using Crow's foot notation. Clearly state any assumptions you make.

- a) Identify all possible entities and relationships. Name all relationships.
- b) Resolve any many-to-many relationships.
- c) Identify (i) strong and weak entities and (ii) identifying and non-identifying relationships.
- d) Identify the main attributes in each entity including all primary and foreign keys.
- e) Identify the Cardinality and Participation (optional and mandatory dependencies) for each relationship.

Question 2: Based on the information in the interview transcript above, use Visual Paradigm to create a logical Entity-Relationship (ER) Diagram using the Crow's foot model symbols and include all attributes. Your diagram must:

- a) Identify (i) strong and weak entities and (ii) identifying and non-identifying relationships.
- b) Identify the main attributes in each entity including all primary and foreign keys
- c) Identify the cardinality and participation (mandatory/optional dependencies) for all the relationships
- d) Resolve all M:N relationships.

Question 3: Based on the business rules developed, construct a logical ERD for the case study using the Visual Paradigm tool.

- a) Identify all attributes in each entity including all primary and foreign keys (any composite and multivalued attributes must be transformed according to the rules of the relational model).
- b) All relationships must be named (i.e. Verb phrases must be shown for all relationships)
- c) If super/subtypes are used, indicate whether the super-subtype relationships are disjoint or overlapping, and fully or partially specialised (complete or incomplete).
- d) Identify the Cardinality and Participation for each relationship.
- e) Resolve all M:N relationships.
- f) Describe any assumptions you have made in a text note in the diagram.

All these questions require the students to develop a logical ERD despite having some variations in the case study. Question 1 was from the 1st assessment given to the students, and their performance was quite good, with an average mark of 16.18 out of 20 (80.9%). But with questions 2 and 3, the average mark percentage dropped to 71.45% and 78.85%, respectively. The drop in the average mark in question 2 was possible since it was from the time controlled mid assessment and question 3 was from the final database project, where the case study was quite extensive compared to the other two case studies.

The average combined mark on these three questions was 46.24 out of 60 (77.06%), with three-quarters of the students scoring more than 55, which is extremely good considering that the students need to develop the models based on an unfamiliar case study from scratch. However, when looking into the average marks, this can be considered the second most challenging type of question following model transformation – Type 1 questions.

Data analysis and results

This section discusses the statistical analysis conducted to examine the relationship between the selected questions. It was decided to use stepwise multiple regression to identify the relationships among the different types of questions. It has been successfully applied to measure the same in novice programming (Lopez et al., 2008).

In this multiple regression technique, in the initial step *Model creation* was considered the outcome variable, while all the other types of questions were considered predictor variables. Then the multiple regression was conducted, selecting the stepwise regression as the technique which combines forward selection and backward elimination methods. When running the stepwise regression at each step, the predictor variable that contributes most to the prediction model is entered first based on statistical criteria. Parallely after adding each variable, all predictor variables are checked to see if their significance has been reduced. If a non-significant variable is found, that will be removed from the model. This process continues until additional predictor variables contribute significantly to the regression model, and the analysis stops when no additional predictor variables add anything statistically meaningful to the model.

Once the significant predictors for the *Model creation* were identified, the process was then repeated, taking each of the predictor variables as an outcome variable to identify the significant predictors. This process continued for all the identified variables, presumably to build a path diagram that can explain the potential associations between the variables.

Checking assumptions

Prior to the multiple regression analysis, several assumptions need to be checked related to linearity, multicollinearity, heteroscedasticity and normality (Berry & Feldman, 1985). Linearity among the outcome and predictor variables was checked using the scatterplots. The following Figure 1 indicates a scatter plot matrix developed among the variables, and as given, there were linear relationships between each outcome variable and the related predictor variable. Even though some were weak linear relationships, it was decided to conduct the analysis as the assumption was met. The multicollinearity that occurs when predictor variables are highly correlated can be observed using Figure 1. Further, this assumption was tested using the Variance Inflation Factor (VIF), where the values were less than three and assumed no multicollinearity among the predictor variables.

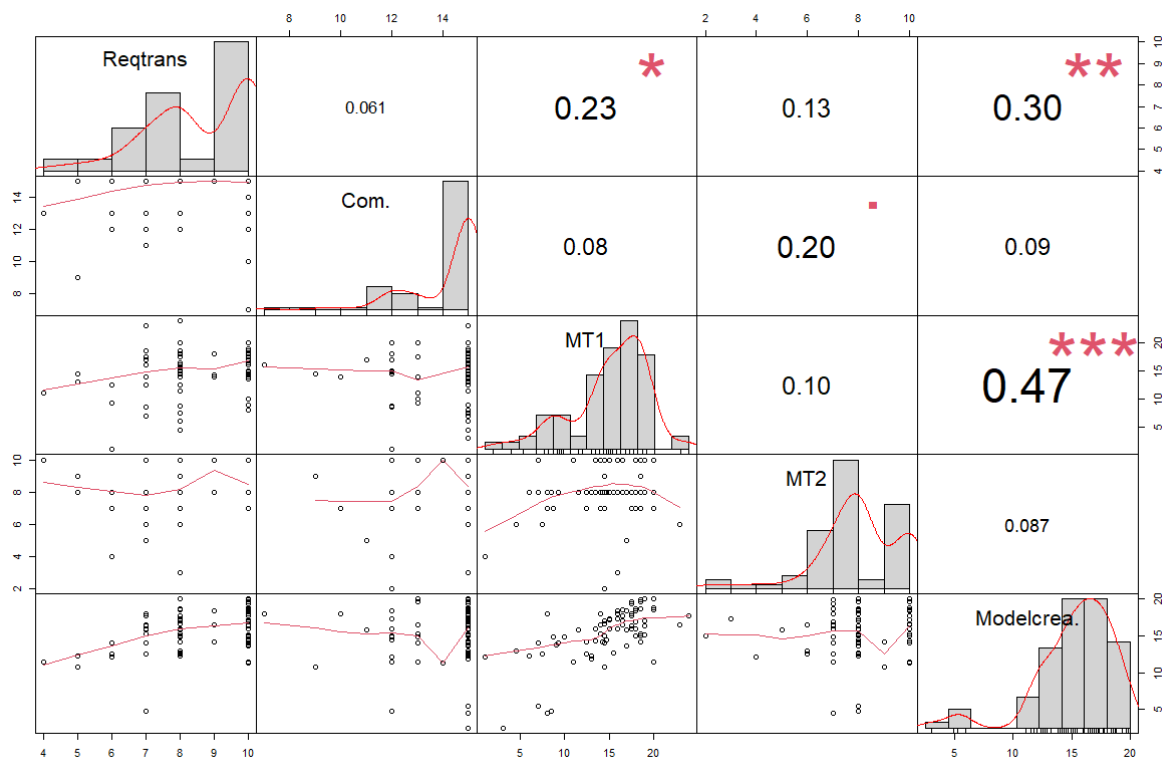


Figure 1: Scatter plot matrix among the variables. In this matrix, the distribution of each variable is shown on the diagonal, while the bottom of the diagonal displays the scatter plots with a fitted line. The values of the correlations (Kendall's tau-b) and the signification levels (indicated using starts) are displayed on top of the diagonal.

Another assumption of multiple linear regression is that the variance of the residuals is constant, which is also called homoscedasticity. This was tested by plotting the standardised residuals against the predicted values on a scatter plot to determine if the points were fairly spread across all the values. And the data indicated no homoscedasticity and hence assumed the assumption was met. The multivariate normality was checked using the normal P-P plots, which show how the residuals' values are distributed. All the P-P plots indicated that the residuals approximately followed a normal distribution. Even though this suggests that the assumption may have been violated, only extreme deviations from normality are likely to impact the findings significantly. Hence we continued the analysis; however, this was an indication that the results needed to be interpreted with caution. Finally, multiple regression observations should be independent of one another (no autocorrelation). This assumption was checked using the Durbin-Watson statistic, and the assumption was met since the obtained Durbin-Watson values were between 1.5 and 2.5, indicating no autocorrelation among the observations.

Evidence for a hierarchy?

No prior literature explored the dependency of modelling tasks and the possibility of a hierarchy of such skills. However, we had our intuition about how the hierarchy of different modelling skills was structured. This was further established with the analysis we conducted to understand the nature of the database modelling tasks (Imbulpitiya et al., 2021). This section discusses the findings of the multiple regression analysis to establish whether the hypothesised hierarchy is consistent with the path diagram that emerged from the statistical analysis.

To develop the path diagram, initially, multiple regression was conducted, introducing *Model creation* as the outcome variable while the other variables were used as predictors. The model summary output in Figure 2 indicates the potential models identified via the stepwise multiple regression technique.

Model Summary				
Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.673 ^a	.453	.444	2.255
2	.701 ^b	.492	.476	2.190

a. Predictors: (Constant), MT1
b. Predictors: (Constant), MT1, ReqTranslation

Figure 2: Model Summary – SPSS output.

The R values (0.673 and 0.701) for models 1 and 2 suggested a strong relationship between the outcome and predictor variables. This indicates that both models can be considered relatively a good predictors of the outcome. Further, Model 1 showed that 45.3% of the variance in the data could be explained by the predictor variable (*Model transformation – Type 1*), while 49.2% of the variance in data can be

explained by model 2 using *Model transformation – Type 1 and Requirement translation* predictors. Comparing the R^2 values of both models, model 2 was deemed the better fit to predict the outcome variable.

Then, we looked into the Analysis of Variance (ANOVA) table output of SPSS to check whether the identified model is a significant predictor of the outcome variable (See Figure 3). The ANOVA results indicated that model 2 was a significant predictor of the *Model creation* variable, $F(2,64)=30.948$, $p<0.001$.

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	273.463	1	273.463	53.800	<.001 ^b
	Residual	330.393	65	5.083		
	Total	603.856	66			
2	Regression	296.882	2	148.441	30.948	<.001 ^c
	Residual	306.974	64	4.796		
	Total	603.856	66			

a. Dependent Variable: ModelCreation
b. Predictors: (Constant), MT1
c. Predictors: (Constant), MT1, ReqTranslation

Figure 3: ANOVA – SPSS output.

Once we established that the overall model is a significant predictor of the outcome variable, the coefficient table was used to interpret the contribution of individual predictor variables to the model. The SPSS output for the coefficient is provided in Figure 4. According to the p values, both *Model transformation – Type 1* ($p<0.001$) and *Requirement translation* ($p=0.031$) contribute significantly to predicting the *Model creation* variable. Further, looking at the standardised coefficients (Beta), *Requirement translation* skill has a positive 0.212 relationship with the outcome *Model creation*, which is significant at a p-value of 0.301. Similarly, *Model transformation – Type 1* skill has a positive 0.595 relationship with the outcome variable, which is significant ($p<0.001$). Finally, the outcomes were used to initiate a path diagram, as indicated in Figure 5.

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	9.123	.909		10.035	<.001
	MT1	.437	.060	.673	7.335	<.001
2	(Constant)	6.316	1.547		4.083	<.001
	MT1	.386	.062	.595	6.204	<.001
	ReqTranslation	.416	.188	.212	2.210	.031

a. Dependent Variable: ModelCreation

Figure 4: Coefficient – SPSS output.

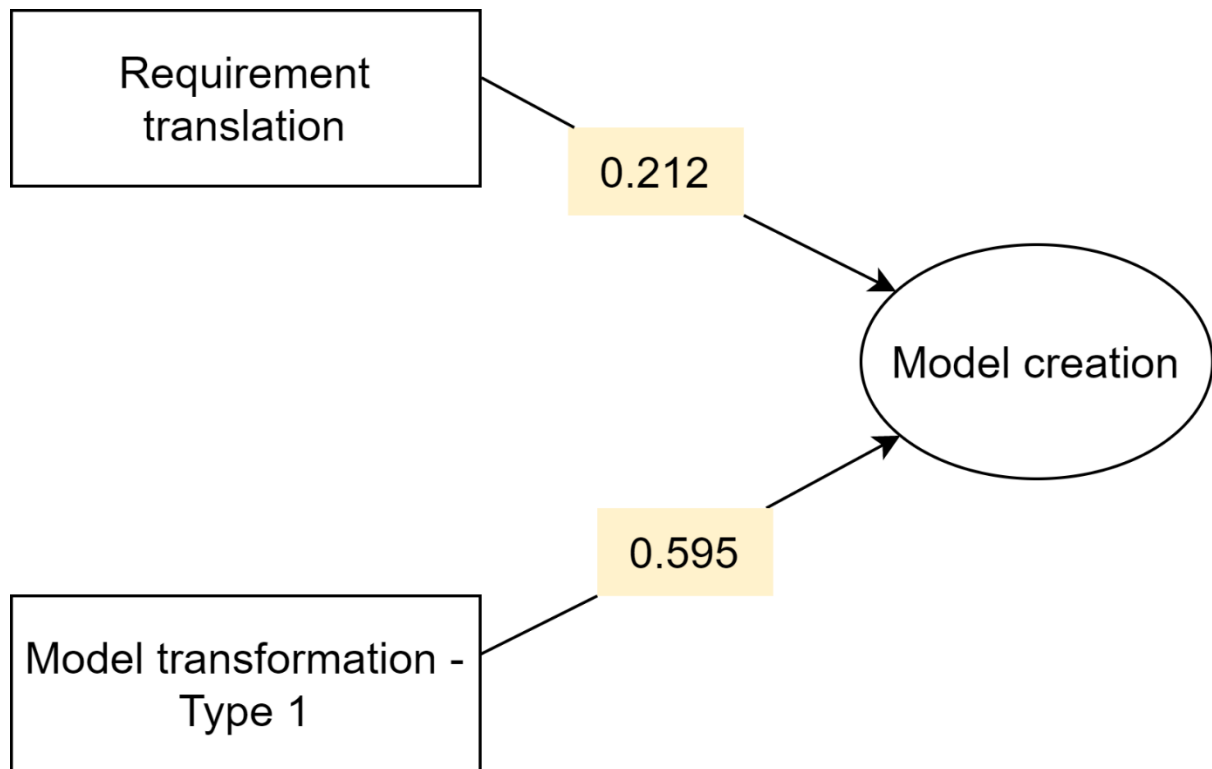


Figure 5: Initial path diagram. The boxes on the paths indicate the beta weights of the paths.

This stepwise multiple regression analysis was continued by taking each variable as an outcome variable while placing other variables as predictors at a time. The initial path diagram was extended along with the emerged statistical outcomes. Figure 6 indicates the completed path diagram that emerged from the statistical findings.

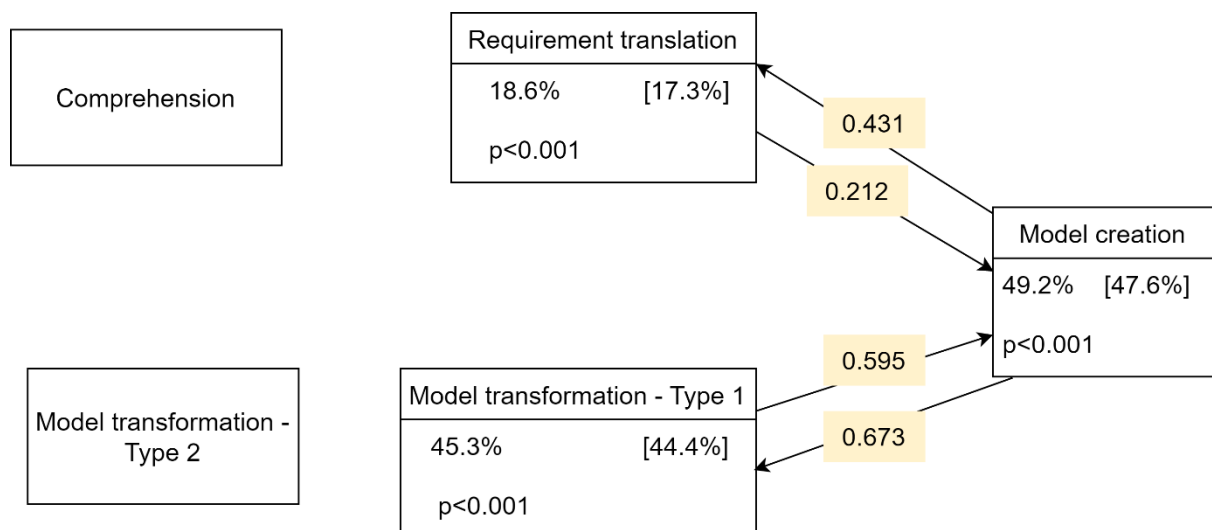


Figure 6: The path diagram. The notations of the path diagram were adapted from (Lopez et al., 2008). In the path, diagram variables are indicated using a titled box which contains the variance explained (R), adjusted R^2 in square brackets and the significance of the overall significance (p-value). The box in the path shows the standardised coefficient (Beta) of the path.

As per the hypothesised model, we assume that *Describe* type tasks are at the bottom of the hierarchy while *Model creation* is on the topmost, followed by *Model evaluation* tasks. It was concluded based on the developed framework for classifying database modelling tasks (Imbulpitiya et al., 2021). However, due to being limited to assignments, no questions were categorised under *Describe* and *Model evaluation* tasks. However, with the emerged findings, *Model creation* appeared at the top of the hierarchy, consistent with the hypothesised model since the students need to master several skills to develop a model from scratch. In combination, *Requirement translation* and *Model transformation – Type 1* account for 49.2% of *Model creation*, consistent with our intuition that *Requirement translation* and *Model transformation – Type 1* are intermediate skills. However, we were surprised that *Comprehension* didn't account for the variance of *Model creation*. In fact, we hypothesised that *Comprehension* would explain the variance of every skill, even in a smaller proportion. However, the *Comprehension* task used in the assignment might have been too easy to determine the association between other tasks.

Prior to the statistical analysis, we hypothesised a statistically significant relationship between *Model transformation – Type 1* and *Model transformation – Type 2*. We were surprised that it didn't emerge from the analysis despite both tasks belonging to a transformation of a model from one type to another. Perhaps the given tasks for the assignment were too simple to draw a conclusive relationship between the tasks. Further, this proves the argument by Lopez et al. (2008) in a similar study conducted in programming, where they explain that characterising the difficulty of a task simply by its nature might be insufficient. Interestingly only the *Model creation* explains the variance of *Requirement translation* with only 18.6% of the variance. However, *Model creation* also solely accounts for 45.3% of *Model transformation – Type 1*. Based on our intuition and the prior work conducted in programming, we didn't expect such reverse associations among the tasks. Nonetheless, this makes a valid argument as an example, if a student can solve a *Model transformation – Type 1*, they are more likely to do *Model creation*. On the other hand, students who can do *Model creation* are capable of completing *Model transformation – Type 1* tasks.

Even though the outcome of the emerged path diagram contradicted the hypothesised model built upon our prior intuition, we can confirm that a hierarchy might exist between different database modelling skills. At least there is statistical evidence to indicate the relationship and a hierarchy among *Model creation*, *Requirement translation*, and *Model transformation -Type 1* tasks. There were also surprising outcomes in the path diagram where *Comprehension* and *Model transformation -Type 2* were unconnected. The main limitation of this analysis might have been the instrument utilised since we used the marks obtained by the students from different assignments. This contradicts the research instruments used in prior research in programming, where the researchers have used exam papers as a research instrument to explore the concept of skill hierarchy. Also, the limited number of questions students

attempted during the assignments (e.g., only one question under Comprehension) raises an argument about whether it's sufficient to understand the relationships between other skills. Nonetheless, as per the best of our knowledge, this is the first attempt to explore a hierarchy between the database modelling skills and warrants future work in the area to address the limitations and difficulties we encountered during the study.

Appendix L

Think Aloud protocol script

Introducing the process

[Present a copy of the consent form to the participant.]

“Hello, thank you for participating in this session. We are interested in how novice database designers solve database modelling tasks, so that’s why we have asked you and some other students to do the logical database design paper to attempt some database modelling tasks for us and let us listen to how you actually do that. The tasks will be similar to what you do in LDD paper, so don’t worry if at points you struggle we are rather interested in how you are thinking about the tasks than the answer.”

“So before we start, have you ever participated in a think-aloud session before or heard about it?”

[Provide the answer based on what students have to say, assuming they are not familiar with the concept,]

“So what happens in a think-aloud session is you have to talk aloud while you are doing the task, that means basically you have to talk about everything that goes through your mind while doing it.”

“Are you clear with that idea? If you need I can explain more or we can go ahead and do a small practice session. What do you like to do?”

Practice Session

[Assuming that they need to do a practice session,]

“Now I’m going to ask you to try this simple task of identifying entities in this scenario in the same way I described earlier. So we are using this smart pen and paper to record your voice and capture what you are writing.

We have found that we can identify a great deal of information if we ask people to think aloud as they work. All you have to do is talk aloud about everything that goes through your mind while doing it. It might be a bit awkward at first, but it’s easy once you get used to it.

I’ll help you out if you are struggling at some point or if you forget to talk aloud I’ll remind you to do it. I might not intervene much in the actual session compared to this practice session.”

“Do you have any questions if not we can go ahead and start now?”

[Ask the students to click on the record button on the dot paper first and start the practice session.]

.

[During the session if a participant becomes quiet for more than 10 seconds prompt them]

1. I'd like to hear what you are thinking.
2. If you could just say what you are thinking

[End of the practice session]

“Ok so do you have any questions, if not we can go ahead and start the actual session.”

Actual Session

“Ok, so our main task is based on this scenario. You have to identify all the likely entities, attributes and relationships. You can write the answer in the area given under each paragraph.”

[Prompt the participants to think aloud if they stay quiet for more than 10 seconds.]

[If they struggle more try to give some hints.]

[Even with the hints they can't move forward redirect them to the less difficult tasks.]

.

Retrospective meeting

[This will be conducted as required at the end of each task or session]

“You did really good, I have few questions for you. I will be recording your answers to these questions.”

1. I noticed you had a long pause during (point to the time frame), what were you thinking during that time?
2. Which of the designs/models did you find it hard to develop and why?
3. Is there anything that you found particularly difficult while creating the models/designs?
4. Did you find it helpful when you were redirected to a different task while struggling to continue with the task you were given?
5. What would you do differently in developing a database model/design next time?

**** Maybe a question around domain knowledge to get the student's perspective.**

1. Are you familiar with the domain that you worked with today? Do you think it will be easy to work with a domain that you are not familiar with?