TONY CLEAR,
AUCKLAND UNIVERSITY OF TECHNOLOGY

Combining Practitioner and Academic Views: Planning a Multi-Vocal Literature Review

ILLUSTRATION ©SHUTTERSTOCK/IGOR LINK

This column provides a brief primer for educators and students on the process of developing a protocol as part of planning a multivocal literature review (MLR) [15]. It arises from reflections on supervising doctoral and postgraduate studies in contemporary computer science (CS) and software engineering (SE) topics. CS and SE research demands that we help students demonstrate a strong grounding in what Brinberg and McGrath have termed the "substantive domain" of large systems and software engineering practice [3]. One strategy for developing this awareness involves conducting a literature review to build a solid awareness of current academic literature on the topic. Various forms of systematic literature review have been advocated in the SE discipline by Kitchenham and colleagues [20] in driving a movement toward rigorous forms of "evidence-based software engineering" [12]. Contrasting with the postgraduate focus of this column, I have previously written on the merits of systematic literature reviews, as one option, for undergraduate computing students undertaking research as part of capstone projects [9]. When adopting systematic strategies for the chosen research inquiry, they typically might include:

- Systematic Mapping Studies [19, 25]
- Systematic Literature Reviews [20][6]
- Meta-Reviews or Tertiary studies [4,30]

Within those strategies, searches may include snowballing approaches [31].

As the title suggests, MLRs add to these strategies for systematic reviews as they seek to listen to and report the voices from a diverse set of actors and sources. These encompass academic and contemporary industry perspectives and are driven by the need to better capture the state of the art in contemporary CS and SE practice. Clearly this has led to this increased interest in the practitioners' voices. In retaining a commitment to a rigorous and evidence-based approach, scholars such as Garousi have developed a set of guidelines to rigorously conduct MLRs [15,16]. An example can be seen in the review of smart contract languages by Varela-Vaca and colleagues [29], and a review of DevSecOps, (with an accompanying protocol), exemplified in the doctoral work of Zhao [32]. MLRs may be preceded by a multivocal literature mapping study (MLM) [15,14], "to classify the body of knowledge in a specific area ... and extended by follow-up studies" to an MLR which may conduct a further in-depth analysis on a given subject. Such MLR and MLM reviews will include evidence drawn from 'white literature' (WL), traditional peer-reviewed academic sources, and 'grey literature' (GL), sources that may not have been peer reviewed but still have some standing or useful insights to add. Garousi and colleagues have observed "Our key findings indicated that (1) GL can give substantial benefits in certain areas

of SE and (2) that the inclusion of GL brings forward certain challenges as evidence in them is often experience and opinion based" [15]. They further advise that omitting the *grey literature* sources may miss critical information that could have a profound impact on steering research directions and limit access to deep expertise held by practitioners (e.g., in areas such as test automation). As a further strategy for identifying relevant sources, Varela-Vaca and colleagues have recommended snowballing [15, 31] as one way to identify such grey literature sources from trawling through the citation lists provided in the 'white literature' sources [29].

Prior to commencing a systematic or multivocal literature review, it is recommended practice, consistent with the tenets of evidence-based software engineering [12], to create an accompanying "protocol" [13]. The protocol aims to guide the process of conducting the review and to transparently evidence the quality of the methodology while revealing the details of the project undertaken. This is helpful for those reviewing the study or seeking subsequently to replicate the findings.

Originating with evidence-based health care researchers, the *preferred reporting items for systematic reviews and meta-analyses* (PRISMA) reporting standard lists information about the systematic review protocol as an "essential element" [24] in Item 24 of the standard. Other guidelines, such as *The Cochrane Handbook*, list registering and completing a protocol as one of the important steps to a successful systematic review [11].

The ability for student and supervisor to dialogue over the protocol, as it iterates through its development, engages students in a more active and transparent form of learning rather than simply asking a chatbot to generate the review

In the CS and SE disciplines, protocols are more variably reported, sometimes made available as technical reports through research repositories in university departments or research centers [2, 6], or as supplementary open science materials for published journal articles [32], and in one case as a preprint in arXiv [1]. Some selected examples of available protocols for reference are given in Table 1.

For students and supervisors embarking on some form of systematic review, the value of a protocol cannot be understated. It not only serves as a guide to the process but enables progress to be assessed, constructive dialogues between students and supervisors to take place during regular supervision meetings, and a transparent audit trail to result in evidencing the research undertaken. In an era when 'AI slop' predominates and repositories such as arXiv has ceased to accept review articles [5], a concrete mechanism to support "ethical learning" [8] and avoiding "cognitive offloading" [8, 22,23], thereby diminishing learning by the student, is crucial. The ability for student and supervisor to dialogue over the protocol, as it iterates through its development, engages students in a more active and transparent form of learning rather than simply asking a chatbot to generate the review. Chatbot-generated research reports are mysterious for both the student and the supervisor and are not conducive to "ethical learning" [8].

Table 2 offers an overview of the steps to be completed in producing a solid protocol and subsequent

Table 1. Selected examples of Systematic Review Protocols.

Title	Author(s)	Year	Type of Review	Source
Protocol for Challenges and Recommendations for the Design and Conduct of Global Software Engineering Courses: A Systematic Review [(ITiCSE Working Group One: Technical Report No. Lero_TR_2015_01)].	Beecham, S., Clear, T., Barr, J., & Noll, J.	2015	SLR	Lero Repository https://lero.ie/sites/default/files/Protocol-Teaching_GSDSLR.pdf
Identifying the Primary Dimensions of DevSecOps: A Multi-vocal Literature Review - Protocol	Zhao, X., Clear, T., & Lal, R.	2024	MLR	Zenodo Dataset link: https://doi.org/10.5281/zenodo.7959584
Protocol for a systematic literature review of motivation in software engineering.	Beecham, S., Baddoo, N., Hall, T., Robinson, H., & Sharp, H.	2006	SLR	University of Hertfordshire https://uhra.herts.ac.uk/id/eprint/12247/1/S73.pdf
Requirements engineering in global scaled agile software development environment: a multi-vocal literature review protocol.	Antil, P.	2020	MLR	*arXiv preprint arXiv:2004.12647*

conduct and reporting of the systematic review. More in-depth analysis of what each step of the process involves can be found in other sources [13,15, 32].

Table 2 has suggested a straightforward and sequential set of steps. Yet Felizardo unpacks the iterative nature of protocol development, demonstrating despite apparently linear travel in a figure that has analyzed the degree of iteration that may evolve as each step of a systematic review is planned and the protocol extends [13]. An adapted version of that figure, showing the degree of evolution as the protocol develops, is presented here as Figure 1. The iterations can be further facilitated by a "pilot test" of the prepared draft protocol using a small sample of studies, in effect conducting a 'mini SLR' from which refinements to the protocol may be added. In a sense, the pilot study/test could be considered a critical further step in the protocol development—depicted as step P8 in Table 2 and as a column in Figure 1.

In reflecting on the process of developing a protocol for a systematic review, da Silva Júnior [10] and Felizardo [13] have distinguished the steps in the process as being essentially *creative* or *mechanical*. It is self-evident that these creative steps should *not* be fully automated if learning ethically (with or without AI) is the aim [8] and for educators to behave professionally consistent with the ACM codes of ethics [17,18]. "Ethical learning" [8] with AI has arguably been overlooked in Sami [26] and Scherbakov [27], who focus more on how AI might be used to *automate the process* and in generating the *product* of the review. From both student and supervisor points of view, the need to engage in creative thinking when developing the protocol provides the best opportunity for dialogue over the purpose and focus of the systematic review. This dialogue serves the purpose

Table 2. Systematic Review Steps and Activities

Protocol Creation Steps	Activity	Review Conduct Steps	Activity	Review Reporting	Activity
P1	Background and Motivation	C1	Search and Upload	R1	Collate the final research report
P2	Research Questions	C2	Deduplication		
P3	Search Strategy	C3	Title/Abstract Screening		
P4	Search Strings	C4	Full-text Screening		
P5	Inclusion/Exclusion Criteria	C5	Quality Assessment		
P6	Quality Assessment	C6	Data Extraction		
P7	Data Extraction Form	C7	Synthesis		
P8	Pilot Study				

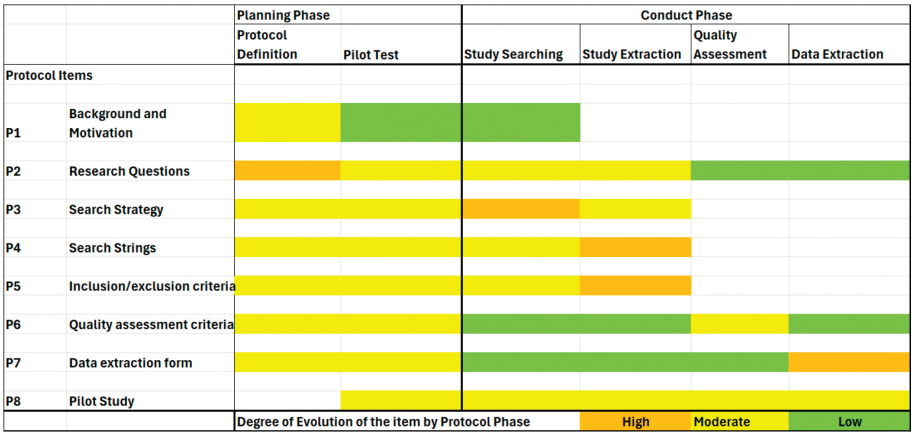


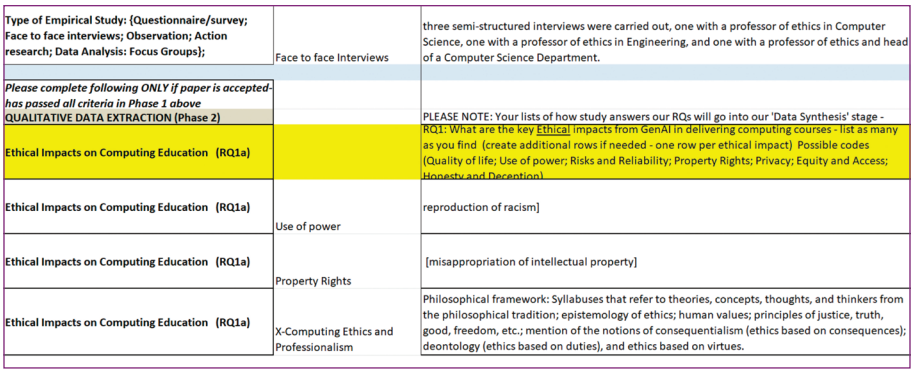
Figure 1. Protocol Degree of Evolution (adapted from Felizardo, K.R. et al [13]).

(for students) of coming to grips with the research process involved in planning the conduct of a systematic review and for evidencing their thought process. This in turn creates opportunities for the supervisor to critique drafts and help improve the focus and design of the study, thus building confidence in the student's grasp of the goals, the artifacts to be created as a systematic review unfolds, and the stages in the planned research process.

Having jointly worked through the process of developing a protocol for the research project, confidence in the student's understanding of what is involved then grows. (An example of a data-extraction form is provided in Figure 2 to illustrate just one element of the protocol produced). The student should now be ready to engage in the review proper with a clear sense of direction and through that growing confidence be better motivated to undertake the work independently. The goal is to prevent students (out of confusion) needing to grasp for the nearest AI tool to make deceptively plausible sense of the process and desired outcome.

Therefore, what is advocated here is a dialogic process of learning through a co-design approach between student and supervisor in developing a protocol, conducting

and collating the results to report the review. Protocol development can be seen essentially as a creative process of progressively revealing the methodology design for a systematic review—both the *how* and the *what* of the research. Therefore, any limited role for AI in assisting in the more *mechanical* steps in the process needs to be both agreed upon and made transparent [8]. This co-design strategy presents a relational model of conscious research ownership and supervision with the steps traversed and the results reported made transparent to all. Developing a protocol for systematic reviews in this way is a design that promises an active form of *ethical learning* between student and supervisor. ◆



Type of Empirical Study: (Questionnaire/survey; Face to face interviews; Observation; Action research; Data Analysis: Focus Groups);	Face to face Interviews	three semi-structured interviews were carried out, one with a professor of ethics in Computer Science, one with a professor of ethics in Engineering, and one with a professor of ethics and head of a Computer Science Department.
Please complete following ONLY if paper is accepted- has passed all criteria in Phase 1 above		
QUALITATIVE DATA EXTRACTION (Phase 2)		PLEASE NOTE: Your lists of how study answers our RQs will go into our 'Data Synthesis' stage -
Ethical Impacts on Computing Education (RQ1a)		RQ1: What are the key Ethical impacts from GenAI in delivering computing courses - list as many as you find (create additional rows if needed - one row per ethical impact) Possible codes (Quality of life; Use of power; Risks and Reliability; Property Rights; Privacy; Equity and Access; Honesty and Deception)
Ethical Impacts on Computing Education (RQ1a)	Use of power	reproduction of racism]
Ethical Impacts on Computing Education (RQ1a)	Property Rights	[misappropriation of intellectual property]
Ethical Impacts on Computing Education (RQ1a)	X-Computing Ethics and Professionalism	Philosophical framework: Syllabuses that refer to theories, concepts, thoughts, and thinkers from the philosophical tradition; epistemology of ethics; human values; principles of justice, truth, good, freedom, etc.; mention of the notions of consequentialism (ethics based on consequences); deontology (ethics based on duties), and ethics based on virtues.

Figure 2. Sample Data Extraction Form

REFERENCES

1. Antil, P. Requirements engineering in global scaled agile software development environment: A multi-vocal literature review protocol. *arXiv preprint* arXiv:2004.12647, (2020).
2. Beecham, S. et al. Protocol for a systematic literature review of motivation in software engineering. University of Hertfordshire, Hertfordshire, U.K. (2006); https://uhra.herts.ac.uk/id/eprint/12247/1/S73.pdf
3. Brinberg, D. and McGrath, J.E. A validity network schema. In *Proceedings of the 67th Annual Meeting of the American Educational Research Assoc.* (1983); https://files.eric.ed.gov/fulltext/ED230607.pdf
4. Budgen, D. et al. Reporting systematic reviews: Some lessons from a tertiary study. *Information and Software Technology* 95 (2018), 62-74.
5. Castelvecchi, D. Preprint site arXiv is banning computer-science reviews: Here's why. *Nature* (Nov. 7, 2025); DOI: https://doi.org/10.1038/d41586-025-03664-7
6. Clear, T. et al. Challenges and recommendations for the design and conduct of global software engineering courses: A systematic review. N. Ragonis and P. Kinnunen, (ed). In *Proceedings of the Working Group Reports of the 2015 on Innovation & Technology in Computer Science Education Conf.* ACM (2015), 1-39; DOI: http://dx.doi.org/10.1145/2858796.2858797
7. Clear, T. et al. AI integration in the IT professional workplace: A scoping review and interview study with implications for education and professional competencies. In *Proceedings of the 2024 ACM Conf. on Innovation and Technology in Computer Science Education*, ACM (2024); DOI: 10.1145/3760545.3783975
8. Clear, T., Dhanjal, M., and Clear, A. Ethical learning in higher computing with agentic-AI. A 'Human-in-the-Loop' protocol [under review]. In *Proceedings of the 2026 ACM Conf. on Innovation and Technology in Computer Science Education*, ACM (2026).
9. Clear, T. Systematic literature reviews and undergraduate research. *ACM Inroads 3, 4* (2012), 10-11.
10. da Silva Júnior, E.M. and Dutra, M.L. A roadmap toward the automatic composition of systematic literature reviews. I*beroamerican J. of Science Measurement and Communication* 1, 2 (2021), 1-22; DOI: 10.47909/ijsmc.52
11. Duke Medical Center Library & Archives Systematic reviews. Duke University Library, Durham, NC, (2026); https://guides.mclibrary.duke.edu/sysreview/protocol
12. Dyba, T., Kitchenham, B.A., and Jorgensen, M. Evidence-based software engineering for practitioners. *IEEE Software* 22, 1 (2005), 58-65.
13. Felizardo, K.R. et al. Defining protocols of systematic literature reviews in software engineering: A survey. In *Proceedings of the 43rd Euromicro Conf. on Software Engineering and Advanced Applications* (2017), 202-209; DOI: 10.1109/SEAA.2017.17
14. Garousi, V. et al., Software test maturity assessment and test process improvement: A multivocal literature review. *Information and Software Technology* 85 (2017), 16-42.
15. Garousi, V., Felderer, M., and Mäntylä, M.V. Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. *Information and Software Technology 106*, (2019), 101-121.
16. Garousi, V. et al. Introduction to the special issue on: Grey literature and multivocal literature reviews (MLRs) in software engineering. *Information and Software Technology* 141 (2022), 106697.
17. Gotterbarn, D. et al. ACM code of ethics: A guide for positive action. *Commun. ACM* 61 (2017), 121-128.
18. Gotterbarn, D. Miller, K., and Rogerson, S. Software engineering code of ethics. *Commun. of the ACM* 40, 11, (1997), 110-118.
19. Jalali, S. and Wohlin, C. Agile practices in global software engineering: A systematic map. In *Proceedings of the 5th IEEE Intern. Conf. on Global Software Engineering*, IEEE (2010), 45-54.
20. Kitchenham, B. and Charters, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering*, Technical report, EBSE-2007-03, Keele University, Staffordshire, U.K. (2007); https://legacyfileshare.elsevier.com/promis_misc/525444systematicreviewsguide.pdf
21. Kitchenham, B. et al. Systematic literature reviews in software engineering: A systematic literature review. *Information and Software Technology* 51, 1 (2018), 7-15.
22. Mak J. et al. A plan for an ACM task force working group into the ethical and societal impacts of generative AI in higher computing education. In *Proceedings of the 2025 ACM Conf. on Innovation and Technology in Computer Science Education*. ACM (2025), 2; DOI: 10.1145/3724389.3731282
23. Mak, J. et al. Navigating the ethical and societal impacts of generative AI in higher computing education. *arXiv preprint arXiv:2511.15768* (2025).
24. Page, M.J. et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ 372*, 71 (2020); https://www.bmj.com/content/bmj/372/bmj.n71.full.pdf
25. Petersen, K. et al. Systematic mapping studies in software engineering, In *Proceedings of the 12th Intern. Conf. on Evaluation and Assessment in Software Engineering* (EASE) 12, (2008), 1-10.
26. ami, A.M. et al. System for systematic literature review using multiple AI agents: Concept and an empirical evaluation. *arXiv preprint arXiv:2403.08399*, (2024).
27. Scherbakov, D. et al. The emergence of large language models (LLMs) as a tool in literature reviews: An LLM automated systematic review; *arXiv preprint arXiv:2409.04600*, (2024).
28. Vardi, M.Y. I was wrong about the ethics crisis. *Commun. ACM* 68, 1 (2024), 5.
29. Varela-Vaca, Á.J. and Quintero, A.M.R. Smart contract languages: A multivocal mapping study. *ACM Computer Survey* 54 1, (2021), Article 3.
30. Verner, J.M. et al. Risks and risk mitigation in global software development: A tertiary study. *Information and Software Technology 56* 1, (2014), 54-78.
31. Wohlin, C. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th International Conf. on Evaluation and Assessment in Software Engineering*, ACM, New York (2014), 38.
32. Zhao, X., Clear, T., and Lal, R. Identifying the primary dimensions of DevSecOps: A multivocal literature review. *J. of Systems and Software*. 214, (2024), 112063, DOI: https://doi.org/10.1016/j.jss.2024.112063

TONY CLEAR
School of Computing and Mathematical Sciences
Auckland University of Technology
Auckland, 1142 New Zealand
Tony.Clear@aut.ac.nz

DOI: 10.1145/3811411