Do We Teach the Right Thing? A Comparison of Global Software Engineering Education and Practice

Sarah Beecham
Lero, the Irish Software Research Centre
University of Limerick, Ireland
Sarah.Beecham@lero.ie

Tony Clear
Auckland University of Technology,
New Zealand
tony.clear@aut.ac.nz

John Noll
University of East London, UK
j.noll@uel.ac.uk

***Abstract*—Global Software Engineering (GSE) is a reality for even the smallest companies, so software engineering students need to learn how to work in a globally distributed development context. Many approaches to teaching GSE have been described in the literature. Since the majority of software development is done by engineers working in small or medium sized enterprises (SMEs) we now ask: Are today's students being trained to work effectively in small distributed companies?**

We surveyed three GSE SMEs to identify which of 70 Global Teaming Model (GTM) practices were problematic and important to this sample. We then mapped recommendations for GSE educators to those pinpointed GTM practices. Finally, we analysed the level to which these needed GTM practices were addressed by the GSE-Education (GSE-Ed) literature, and who performed these practices. Nine GTM practices were found important and relevant to all three SMEs. Seven of these were addressed by GSE-Ed recommendations, and two were seen to be lacking.

A rich set of 63 unique GSE-Ed recommendations were found to support the seven GTM practices, but our analysis unearthed a surprising complexity of roles and responsibilities undertaken by the instructor in GSE-Ed courses. As a result student and client involvement in coordination and collaboration activities tended to be weakened or non-existent. In order to ensure graduates are prepared for the reality, practitioners of SMEs need to take on a more active role in the education process. Also, students need to be given more responsibility so they can learn the broader professional and management skills required when developing software in multi-site SME teams.

Keywords*—*Global Software Engineering; Software Engineering Education; Global Software Engineering; GSE; Software Engineer; Small to Middle Sized Enterprises; SMEs; Empirical Software Engineering

I. Introduction

Global Software Engineering (GSE, also sometimes called Global Software Development (GSD)) is a reality for all companies, large or small. In order to grow, even the smallest companies must look to markets beyond their local region, which means they also need both sales and development presence in those markets [1, 2]. As a result, software engineering students not only need to learn how to work in a globally distributed development context, they need the skills required by small and medium sized companies working in such an a context.

Many approaches to teaching have been described in the literature, ranging from simulations of particular GSE situations, to comprehensive software engineering project courses involving multiple teams at different institutions across the globe [3]. However, most GSE courses involving distributed projects seem to be motivated by the issues faced by large multinational corporations. Some of the pioneering GSE project courses were sponsored by large multinationals [4]. Yet, as discussed in Section II, the majority of software development is done by engineers working in small or medium size companies. So in this paper we ask, are today's students being trained to work effectively in these small companies?

The Global Teaming Model (GTM) [5] is an empirically-validated framework of recommended practices for Global Software Development. Comprising two CMMI©-style "Process Areas" and 70 recommendations, the GTM is a detailed and validated inventory of practices required for successful global software development.

We adopted this Global Teaming Model as a framework describing the requirements for effective global software development. Then, analysing results from GTM assessment surveys administered to employees developing software in three SMEs, we identified those GTM practices that are particularly relevant to our sample. (All three SMEs were distributed across two or more countries). We then mapped the practices recommended for educators by Clear & colleagues [3] to the identified Global Teaming Model recommendations for SMEs. This enabled us to assess the gap between practices addressed by the Global Software Engineering Education (GSE-Ed) literature, and the needs of SMEs engaged in GSE.

We found that seven GTM practices identified as relevant to all three of our SMEs were addressed by GSE-Ed recommendations, and two (associated with regulated domains) were seen to be lacking. We also found that the three key players in the education process, the instructors, the student, and the client took on very different responsibilities, that were likely to have an impact on learning. The instructors had a surprisingly complex set of roles and responsibilities to juggle and were largely responsible for instigating any given practice (ranging from global cross-university responsibilities, to single institution responsibilities). Students were found to also take on responsibilities, but progressively once the practice was in place, and were actively learning as a result. However, the client typically took a more passive role in the process.

In order to ensure graduates are prepared for the reality of working within SMEs in global settings, practitioners need

to become more engaged with the education process. This can involve identifying candidate projects and playing the role of customer for student projects, serving on advisory boards, or acting as co-instructor. GSE-Ed is found to teach students technical aspects of development, but is still falling short in teaching them the many project management and leadership skills required to manage the more complex coordination and collaboration processes that are an integral part of GSE. In an SME setting where employees carry out many concurrent roles, the ability to solely specialise in a technical area is reduced, with additional client-facing, teamwork and leadership demands often being imposed on employees, so graduates need a generalist as well as specialist skill set. To increase student learning in these non-technical aspects, there needs to be a clear shift from the instructor taking on and instigating a set of project management and leadership tasks, to the students taking on these responsibilities.

This paper is organized as follows: in Section II we introduce the background to the problem, and define our research questions. Section III describes the method used. In Section IV and Section V we present the results, and discuss their implications and limitations; Section VI presents conclusions and future directions.

II. Background

In a comprehensive review of the literature [3], Clear and colleagues identified a wide range of courses and approaches to teaching GSE. More recently, we have debated the merits of multi-site multi-university courses versus simulations, or open source projects [6]. The motivations for multi-site, multi-university courses lie with a desire to better prepare tomorrow's global software engineers through challenging courses representing authentic learning experiences [3, 7, 8]. But in thinking through how we are preparing our students, perhaps we need to look more deeply into our assumptions. Do our graduating students mostly work within large multi-national corporations (MNCs)? For many economies, the majority of software companies are actually small to medium enterprises (SMEs) [9–11]. More recently the European Union has reported that "SMEs form the backbone of the EU28 economy. In 2015, just under 23 million SMEs generated €3.9 trillion in value added and employed 90 million people. They accounted in 2015 for two thirds of EU28 employment and slightly less than three fifths of EU28 value added in the non-financial business sector. The vast majority of SMEs are micro enterprises with less than 10 employees – such very small firms account for almost 93 per cent of all enterprises in the non–financial business sector" [12].

SMEs face constraints that multi-national companies don't; for example, due to limited resources, personnel in SMEs may have to assume multiple roles [13].

GSE is frequently thought of as a large company phenomenon, but in reality software is produced in a far wider range of smaller organizations, many of them working with global collaborators and partners [14]. According to Pino et al, "99.2 percent of software development companies are small and medium (fewer than 250 employees). They develop significant products, for the construction of which the firms need efficient Software Engineering practices that are suitable for their particular size and type of business. [15]"

Yet it is known that SMEs in general, and software SMEs in particular, present opportunities but face particular challenges: "As new technologies and globalisation reduce the importance of economies of scale in many activities, the potential contribution of smaller firms is enhanced" [9]. Yet, "many of the traditional problems facing SMEs ... become more acute in a globalised, technology-driven environment. Small firms need to upgrade their management skills, their capacity to gather information and their technology base" [9]. In a software SME setting specific challenges have been noted such as: the cost of "implementing ... heavy weight software process reference models" and struggling with applying process assessments [16]. Raninen has observed that "Process improvement in software SMEs tends to be moderate in nature" due to SMEs' inability to hire dedicated and suitably skilled personnel, and due to the fact that existing employees are already fully stretched. Therefore Raninen concludes that "SMEs require a lightweight, flexible SPI approach suited to many kinds of processes" [16]. It is likely too that SMEs engaged in GSE require lighter weight and less costly approaches to software development, which has implications for teaching GSE-Ed, outside of the large corporate MNC context. For instance O'Connor [17] has observed that SMEs have limited resources, and are typically practising unique processes in managing their businesses, which impacts on their companies' process infrastructures. As a result most of their management processes are highly informal when making decisions, communicating or problem solving, with a preference for oral discussions over documentation. "This indicates that people oriented and communication factors are very important and significant in very small enterprises" [17]. While GSE-Ed courses certainly exercise students' interpersonal skills, in a distributed setting such informality poses additional challenges.

The pioneering GSE-Ed courses seem to have followed two broad models: either working with large MNCs as clients sponsoring the project, [4, 18] or in a problem based learning mode [19] with assigned project tasks where educators or students perform the roles of client [20–24]. Subsequent initiatives have also included working with smaller companies as clients [25, 26].

In undertaking this study we were further confronted with the issue of the differing stakeholders and roles in the respective GSE-Ed and industry settings (SME or otherwise). Clear and colleagues [3] identified the following groupings for stakeholders and roles in a GSE-Ed course: client, instructor, student, university representative. In addition we had noted the prevalence of role conflict in such courses, and the extent to which students struggled when given leadership responsibilities. Clear and colleagues also remarked on the paucity of studies which addressed a student perspective [3]. The appropriate level of support for students in undertaking broader project management responsibilities remains a vexed issue. For instance in a study of student perceptions in a GSE-Ed context [7], it was clear that a highly techno-centric view predominated among the student cohort, so taking leadership and management responsibilities was likely to prove challenging from the multiple perspectives of students' skills, sense of identity and motivation. One approach outlined in [27], was to use an external mentor as a scaffolding strategy to support the student project leaders with their challenging responsibilities. In [28] "Open Ended Group Projects" were adopted as a de-

liberate pedagogical strategy to develop students' professional competencies, through being exposed to a large, challenging and authentic global collaborative project, in which they had to take responsibility for the direction and conduct of the project and their own learning within it.

Therefore this study investigates whether there is a gap in our findings on the teaching of GSE courses, and how well we are preparing our graduates for working globally within the particular needs of an SME setting. We undertake this investigation by comparing industry needs identified within the *Global Teaming Model* (GTM) [5], with our recommendations for teaching GSE. While the GTM provides the framework for this investigation, we perform our comparison through mapping the needs of GSE-Ed against the findings from a recent industry study in which the needs of software SMEs were mapped against the GTM. The study aims to address the following research questions:

RQ1: What issues do GSE SMEs experience ?

Rationale: 1) The literature shows that SMEs are large employers of Software Engineers. 2) The likelihood is that Software Engineering students will work in an SME. 3) The literature shows SMEs have different structures etc. to MNCs and therefore may experience different issues or have different priorities.

RQ2: What are we teaching our Software Engineering students that relates to GSE SME needs?

Rationale: we need to prepare students for the workplace.

III. Method

The Global Teaming Model contains 70 recommendations for managing Global Software Engineering projects. While the empirical evidence on which the GTM is based suggests that, in general, all of these practices are important, it is not necessarily the case that all apply to any given situation. Consequently, we developed a GTM Assessment instrument [29] to allow companies to assess the importance of each GTM practice to their specific situations, and the degree to which they have implemented those recommendations.

For this study, we used the GTM Assessment instrument [29] to create a list of GTM practices that are important to small and medium size companies (SMEs). Using this assessment, we identified a set of GTM practices that were particularly important to the SMEs in our sample. Then, for each GTM practice, we selected GSE-Ed recommendations from those identified by Clear and colleagues [3], that would give students experience or knowledge about the GTM practice. The resulting mapping shows where Global Software Engineering Education is adequate, and where there are gaps.

The process was performed in three phases, as follows:

A. Phase I – GTM Assessment

First, we had members of the software development staff from an opportunistic sample of three SMEs complete the GTM Assessment from the perspective of their respective companies. The assessment measures the strength of each of the 70 GTM practices using a 10 point scale. Respondents were asked to give each GTM recommendation one of the following ratings:

1) Implemented, works well for me.
2) Implemented, just working this way, not sure of its value though.
3) Implemented, but needs improving as it is not effective in its current form.
4) Implemented, but informally – wouldn't be able to tell you the precise process.
5) Started implementing, then stopped, planning to start when time (sporadic implementation).
6) Planning to implement in near future. – i.e. would like to implement but haven't yet.
7) Would like to implement but can't: Not implemented because my organisation does not have resources, or management won't approve (i.e. in an ideal world would like to, and know about it already, but cannot afford to do this).
8) Not implemented because I didn't know about the practice (i.e. now I know, it seems a good idea and appropriate for my needs; I will think about introducing this practice).
9) Not Needed: Not implemented because it is not necessary for my type of business (e.g., business is too small, don't have any problems that this practice would solve).
10) I don't really understand what the practice is advocating so cannot comment.

Next, we aggregated these responses into four broad categories:

1) **Fully implemented** if the response was 1: "Implemented, works well for me."
2) **Problem Area** if the response was one of 2–8 inclusive, indicating a partial implementation or important but not implemented recommendation. This broad category comprises all responses where the respondent considered the GTM recommendation to be relevant, but not fully implemented. This wide definition of "Problem Area" means our analysis addresses all practices considered relevant but are not already fully addressed.
3) **Not needed** if the response was 9: "Not Needed: Not implemented because it is not necessary for my type of business (e.g., business is too small, don't have any problems that this practice would solve)."
4) **Other** if the response was 10: "I don't really understand what the practice is advocating so cannot comment."

Because there were different numbers of respondents from each company (the smallest had only three members in its development organization), we then computed the *mode* of the frequency that these aggregated categories appeared in each company's responses. This was then assigned as that company's 'vote' on the GTM practice.

Finally, we selected the subset of GTM practices for which *every* company's 'vote' was "Problem Area." The resulting list of practices formed the input for the mapping phase, described next.

B. Phase II – GSE-Ed Mapping

Having identified a set of "Problem Area" GTM practices that applied to all three of our SMEs, we then looked to

the Global Software Engineering Education literature to see whether 1) the GTM practice was recognized by educators in some way, and 2) the practice was addressed in some form by existing GSE-Ed curricula.

To do this, we used a review of 82 Global Software Engineering Education publications conducted by Clear and colleagues [3]. This review synthesized a detailed set of identified GSE-Ed challenges, which were accompanied by an actionable set of recommendations to address them. Stakeholders involved in GSE-Ed were also identified, in which instructors, students, and clients were central to the process.

For each GTM practice in our "Problem Area" set, two of us identified matching recommendations from Clear and colleagues' review, with the third author acting as a moderator. The results are shown in Table IV.

C. Phase III – Identify GSE-Ed Stakeholder Roles

Finally, we took each practice that is being taught or recognised in the GSE-Ed literature as listed in Table IV, and considered who this recommendation is aimed at. For example, the practice: "harmonize processes across institutions and establish a clear line of responsibility" is likely to be implemented by the instructors, and the students may not be aware of this need if not directly involved. Since we are asking "Are we teaching the right things?", we need to clarify exactly what is being taught, and what is being learnt. The responsibility mapping in Table V was achieved by two of us going through each practice listed in Table IV and independently mapping the practice to every role according to how engaged they were with the practice. There are five possible levels of involvement from highly involved to not at all: 1) Committed Instigator; 2) Committed team-member; 3) Involved team-member–active; 4) Involved team-member–passive; 5) Not involved.

During our mapping exercise we identified three key roles: Instructor, Student and Client. Noting that the instructor can have three levels of involvement: *Lead instructor* – who instigates a practice across several sites; *Instructors* (as a group) – whereby several instructors instigate the practice as a group; and *Sole Instructor* – who instigates a practice in one site or university only. While the two researchers who conducted the mapping had a good agreement, the third researcher acted as an arbitrator on any disagreements, and we finally gained 100% agreement across all three researchers as to the level of stakeholder involvement (see Table V).

IV. Results

Table I summarizes the characteristics of the companies that participated in the GTM assessment. They range from very small (three employees *total*) to the upper end of the SME scale (350 employees). They represent a range of software product domains and development processes; the actual respondents have roles from across the software development spectrum, including both technical roles such as developers and testers, and managers. Also, individual respondents came from various levels in their respective organizations, up to the most senior management, with a corresponding range in experience ranging from a few years to more than 20 years.

TABLE I. Participant SME Characteritics

	Company 1	Company 2	Company 3
Product domain	construction	communications	medical informatics
Company size	3	350	200
No. Software Devs	1	50	25
Dev. process	ad hoc	waterfall	agile
No. respondents	3	17	16
Respondent roles	Managing Director, Project Manager, Technical Lead	CIO, CTO, (Sr.) Developer, Program Director, Project Mgr.	(Sr.) Developer, Dev. Director, Product Owner, QA/Test lead, Scrum Master
Respondent nationalities	Australian, Irish	Argentinian, English, French, German, Indian, Spanish	American, English, Filipino, Indian, Iranian, Irish, Latvian, Scottish, South African, Sri Lankan, Welsh

TABLE II. Global Teaming Model practices identified as "problem areas" by SMEs

ID	Description
C2	Provide training to ensure that global team has required understanding of the customer base and the business functions to take full advantage of the proximity of the team to the customer base.
F3	If working in regulatory domain, provide training on regulatory requirements and procedures.
G5	Identify issues from lessons learned that require a wider initiative such as a change in organizational culture and report to a global change management agent.
L1	All potential risks should be identified and addressed to include: risks in misunderstanding cultural differences, misunderstanding requirements, feature volatility, schedules, budgets, personnel.
M2	When defining the global strategy for dealing with conflict, different types of conflict have to be taken into account, for example conflict due to fear as well as cultural differences.
O2	Ensure that relevant team members are made aware of how and when they will receive inputs to products they are working on, and when they need to distribute outputs from these products and when complete work products are required.
Q5	Acknowledging team success may require tailoring rewards to the needs of different cultures.
T4	Establish procedures to coordinate implementation of contingencies when and if required.
U	Establish a risk management strategy for regulation.

Table II lists the practices our participating SMEs identified as being a "Problem Area" by the GTM Assessment in Phase I. These practices form the basis for our analysis of how relevant Global Software Engineering Education recommendations are to SMEs.

Table IV shows the results of the next phase of our analysis, namely mapping of Global Software Engineering Education recommendations to GTM practices identified as being a Problem Area. Not surprisingly, the bulk of the relevant GSE-Ed recommendations fall under the "Curriculum/Pedagogy" theme; Fig. 2 summarizes the distribution of relevant GSE-Ed recommendations across different GSE-Ed themes.

The final phase was to identify stakeholders and their roles in implementing each practice (Table V); these are summarized in Fig. 1. Of note is the fact that the *client* – the stakeholder category that SMEs would fall under – is *never* an instigator, is sometimes a committed, or at least involved, participant, but is most often a passive participant, or not involved at all.

TABLE III. STAKEHOLDER ROLES IN GLOBAL SOFTWARE ENGINEERING EDUCATION.

	instigator	committed	involved–active	involved–passive	not involved
Instructor	56 (89%)	3 (5%)	1 (2%)	3 (5%)	0 (0%)
student	7 (11%)	34 (54%)	8 (13%)	13 (21%)	1 (2%)
client	0 (0%)	7 (11%)	10 (16%)	16 (25%)	30 (48%)

V. DISCUSSION

This section discusses the extent to which graduate courses in GSE Education (GSE-Ed) are meeting the needs of the SMEs in our sample.

A. The Gap

1) Missing support for Regulated Software Development: Out of the nine practices we examined as important and not implemented in organisations, we could not find any GSE courses that dealt with issues concerning developing software in a regulated domain. This omission might be due to three things: That the type of collaborations that student teams engage in do not involve building software that is regulated (e.g. Medical Devices), or that the empirical studies examined in the GSD-Ed literature did not go into that level of detail, or finally that the authors of the GSE-Ed review, did not pick up on this point in their data extractions and synthesis. One possible constraint to student projects in these domains may be due to the regulations themselves, with the US Federal Drug Administration for instance prohibiting certain classes of non-employees from developing software for regulated companies.

A notable exception which we identified subsequently, was the 'IT in Society' Course [28], which has a standing collaboration with a health sector client, and in the instance cited in [28], resulted in a White Paper to the European Commission on issues relating to the online availability of patient records under a new Swedish Law. What does remain uncertain, however, is the extent to which this course addresses the need for a risk management strategy for regulation, or whether training is given on regulatory requirements and procedures.

Clearly, given that this is a need for practitioners (those constrained by regulations in health and finance domains for example), more research is needed. We need to take a closer look at what constraints may be in operation and what the literature is stating about university engagement with regulated software development in GSE-Ed, or, new studies need to be undertaken in which universities collaborate with software organisations willing to provide project experiences for students within a regulated domain. Unless addressed in the context of a client's project and domain needs, it is likely that a more general GSE-Ed course addressing issues to do with regulated domains, (unless highly specialised), would either touch on these particular needs through a survey style lecture or two, or through a research seminar based approach. Alternatively it might delve deeper into one or more specific aspects, perhaps through an assignment into a selected topic area. Striking a balance between the specific and the generic is likely to be challenging, and may be better covered by other courses in the curriculum, for instance a formal methods course.

2) Missed opportunities for Student Learning: Of the seven GTM practices that we found had recommendations associated with them, nearly all the associated 63 GSE-Ed practices required the instructor to instigate the practice, suggesting that rarely were students allowed (or able) to be proactive, or lead. However when looking at whether the student takes over responsibility once the practice has been started or introduced, there is a healthier profile, with over half the practices being driven by students taking over responsibility (34 out of 63). One of the purposes of building software in multi-site teams is to try to emulate a real-world scenario. If this is to reflect the real world, then students need also to be involved in creating the right environment to allow this collaboration to flourish. If the instructors do most of the setting up and running of the projects, we question whether the student is truly learning about how to collaborate and coordinate and control GSE projects. We recommend that students engage early – before developing software, so that they can take responsibility for setting up the development environment; a suggestion for instructors would be to go through all the tasks required to run a GSE-Ed project and for them to give students responsibility for those they feel they have the capacity to deal with. In that way the student will also learn about project management

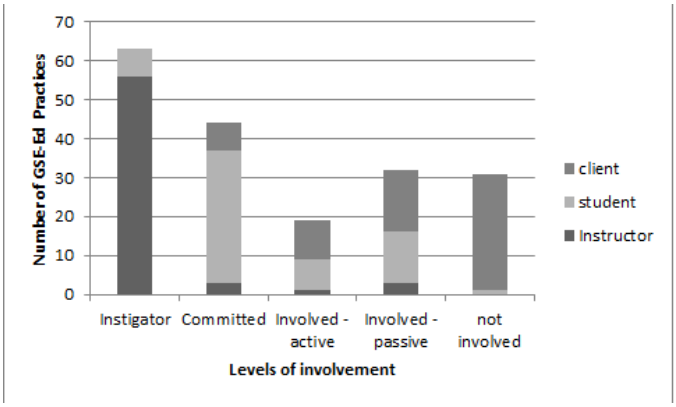

Fig. 1. Summary of stakeholder participation (roles) in Global Software Engineering Education.

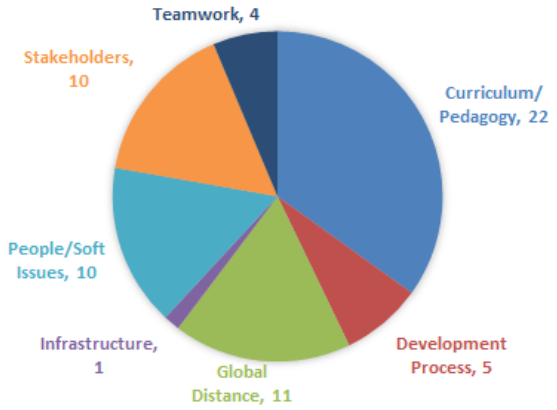

Fig. 2. Summary of Global Software Engineering Education themes that address Global Teaming Model practices.

practices.

Another approach that is not explored well in the literature involves students participating in open source software projects. Open source projects have many aspects of global software development projects, including global distribution, multiple cultures, and lack of face-to-face communication [6].

B. Roles and Responsibilities

As we worked through our GSE-Ed mapping to the GTM practices we were surprised by the complexities and the highly nuanced dimensions which emerged of the roles and responsibilities involved in a GSE-Ed course. In prior work it had been noted that there were a plethora of roles in global software courses [30], but in the synthesis of 82 papers, Clear et al [3] addressed only the obvious positional roles such as 'Instructor', 'student' and 'client'. In conducting the mapping exercise in this paper it became clear that these roles need to be complemented by a further responsibility dimension that addressed the degree of engagement in a practice (e.g. committed instigator, involved team member – passive team member, etc.). On top of this responsibility dimension a level of global involvement in executing the practice by the instructor also emerged as important in GSE-Ed (i.e. 'global leader', 'member of collaborative global group' or 'local site leader').

It is our conjecture that this additional complexity of roles and responsibilities is a phenomenon specific to GSE-Ed, as opposed to typical industrial practice within an SME setting for GSD. The precise reasons for this difference need further study, but two possible reasons may lie in: 1) the relatively loose and informal nature of such global collaborations between educational institutions (and their clients - if present); 2) the learning context of the setting where students may only gradually adopt and master a new set of practices and therefore require support from the instructor to scaffold their learning and in designing and managing the learning context. Thus roles will tend to be fluid and evolving in response to situational dynamics. However, we do also note the need for roles to be flexible in an industry SME setting. since as noted in Raninen et al, there may not be enough personnel to allow for individuals to specialise and remain in one role [16].

These dynamics could for instance explain the three differing levels of involvement by instructors that we see in framing GSE-Ed courses, as: Lead instigator – global, Lead instigators – collaborative or Instructor – local. Such courses typically will have no formal institutional contracts (and probably should not have), since as noted in the prior review [3] they are frequently dependent upon champions, often conducted in the 'skunk works', even in face of opposition from internal forces e.g. student and colleagues' discomfort with courses that diverge from the institutional or programme norm [31]. It is also common for different courses and schedules to be run across sites, institutional policies may differ, and the characteristics and skill sets of student cohorts across sites can be widely divergent.

As noted in the GSE-Ed review [3], gaining full client involvement in GSE-Ed courses appears to be challenging, with our mapping in Table V reflecting this. In very few cases did the client take a fully committed role, with passive rather than active roles being the norm. The client role itself may have some inherent ambiguities where the client is a sponsor providing funding or resources for key aspects of the project (e.g. for students to travel to meet with remote colleagues or setting up competitions, or acting as client for instructor provided projects).

From the student perspective GSE-Ed courses pose challenges for student project leaders, who operate in a situation where they have no formal authority, as noted in [27] due to a "lack of power structure within the group", and have few if any sanctions they can apply. Therefore they have to exercise informal forms of leadership based on the respect, and the perception of competence and expertise accorded to them by their team. Compounding these challenges are the variable skill levels in project management and team leadership that the students may possess. Under GTM practice L1 of Table III for instance, a noted issue is that 'students should have the appropriate background before they enter the course'. But what should that be? What level of assumed knowledge and prior preparation might be stipulated and achieved in practice? Some students or cohorts may lack the necessary technical skills, which can cause problems with resultant loss of faith in team members across sites [24]. For instance in a highly technical computer science course programme students may have had no experience with Project Management prior to such courses, so a Project Management course may need to run in parallel with a GSE-Ed course, or be embedded within it in a just-in-time learning model? As Peters has observed [7] students should probably have the appropriate technical background for the course, but that will mean they are probably less prepared for the more professional capabilities of project management, team leadership and client negotiations.

The instructor then must adopt a supportive role, and build motivation in such students to see these facets as an integral aspect of working as a professional in a global software team. As noted in [28] it therefore behoves GSE-Ed educators to adopt approaches such as 'Open Ended Group Projects' as a strategy to develop students' broader professional capabilities, rather than simply give up and withdraw to delivering courses with a narrow technical focus only based on a *students as coders* model! But as observed by Damian based on her experience [32], "Despite this success, setting up, teaching and evaluating student work in such educational environments require more effort, strategy and instructor's resilience than in traditional courses". The hidden complexities of the instructors' roles and responsibilities that we have unearthed through our mapping of a subset of GTM practices in this study, may go some way to explaining this need for additional instructor effort and resilience, and warrant further investigation.

C. Limitations

We only looked at whether GSE-Ed supports the 9 practices that all of our SMEs indicated were causing them issues, and were looking to improve. Clearly students also need to be aware of those practices that companies have implemented (and also need). However, in order to scope and prioritize this research, we first start with those issues that appear the most difficult to apply in practice.

There is a threat to external validity, as the we only sampled three SMEs to see what their issues were (and took only

those issues that were problematic for all as our base set of requirements for GSE-Ed). The three organisations in our sample may not be representative of all SMEs, however, we applied our method as a way to scope the study, since we could not map GSE-Ed recommendations to all 70 GTM practices.

With respect to internal validity, we believe the degree of independent mapping and subsequent cross checking and discussion to reach a consensus on our mappings has helped us calibrate and consolidate our assumptions related to each practice and role or responsibility assessment, which has built rigour into the research design.

We have one doubt relating to construct validity and our interpretation of GTM practice U - establishing a risk management strategy for regulation. While we have evaluated that practice from the perspective of developing software for regulated domains, it could alternatively have been interpreted as developing a risk management strategy for rules and regulations across collaborating institutions. The GSE-Ed review [3] did note several high level challenges relating to organisational distance e.g. "dealing with continuous changes of rules at each site" and "differing university regulations". Related recommendations suggested "harmonizing processes across institutions and ...establishing responsibilities and power" and that "a key success factor is flexibility in accepting different rules and habits".

VI. Conclusions

In this study we have adopted the Global Teaming Model as a framework describing the requirements for effective global software development. Drawing from a set of assessment surveys of three SMEs, we identified those GTM practices that are particularly relevant and challenging for SMEs. Building on this industry perspective, we then assessed the gap between practices addressed by the GSE-Ed literature, and the needs of SMEs engaged in GSD. We found that seven GTM practices identified as relevant to SMEs were addressed by GSE-Ed recommendations, and two (associated with working in a regulated domain) were seen to be lacking.

We also found that the three key players in the education process, the instructors, the student, and the client took on very different responsibilities, with consequential implications for learning. The instructors had a surprisingly complex set of roles and responsibilities to juggle and were largely responsible for instigating any given practice (ranging from Global-cross university responsibilities to single institution responsibilities). Students were found to also take on responsibilities but progressively once the practice was in place, and were actively learning as a result. However, the client typically took a more passive role in the process.

While we have no survey data to support our conjecture that these complexities and demands inhibit the wider provision of GSE-Ed courses, we are aware from our own experiences that GSE-Ed courses are not in the mainstream for teaching SE. Matthes has observed that "When considering the personal requirement today's software engineers are facing in their daily work life, it is surprising to see that teaching GSE at universities is still in its infancy" [33].

We argue therefore that SME practitioners need to become more engaged with the education process in order to ensure that graduates are prepared for the reality of working within SMEs in global settings. This might involve identifying candidate projects and playing the role of customer for student projects, serving on advisory boards, or acting as co-instructor. Providing opportunities to undertake projects in regulated domains appears to be an area of need. Such partnerships will need to be designed so that there is clear benefit for the busy SME employees involved.

GSE-Ed is found to teach students technical aspects of development, but is still falling short in teaching them the many project management, interpersonal and leadership skills required to manage the more complex coordination and collaboration processes that are an integral part of GSE. In an SME setting where employees carry many concurrent roles, the ability to solely specialise in a technical area is reduced, with additional client-facing, teamwork and leadership demands often being imposed on employees, so graduates need a generalist as well as specialist skill set.

To increase student learning in these non-technical aspects, there needs to be a clear shift from the instructor taking on and instigating a set of project management and leadership tasks, to the students taking on these responsibilities.

A. Future Directions

It is our conjecture that the additional complexity of participant roles and responsibilities highlighted in this study is a phenomenon specific to GSE-Ed, as opposed to typical industrial practice within an SME setting for GSE. The reasons for this and the variations in an industrial setting warrant further study.

While we have focused on a subset of GTM recommendations specific to SMEs, there are many other GTM practices that our companies have implemented that have yet to be investigated. Determining whether those are taught in GSE-Ed classes could be a focus for further work. Also, in order to focus the analysis, our method focused on those practices which respondents from all three companies agreed were relevant. It would be useful to consider other practices that only some SMEs identified important.

Consideration of strategies for managing organisational risk (variability and volatility of rules and regulations between institutions) is a topic we could fruitfully address in later work.

Strategies for involving SMEs in GSE-Ed courses need further work. Perhaps options at an industry rather than individual firm level, such as establishing research groupings within existing industry bodies (e.g. professional societies, user groups and forums) may be an option.

Finally, given the learning goals of a GSE-Ed course, viable options and strategies need to be developed for sequencing student capability development preparatory to such courses and assessing their readiness for GSE-Ed. Such preliminary steps will be critical to developing approaches to weaning students off reliance on their instructors and enabling them to take more active ownership of the course design and operation as an integral part of their learning.

Acknowledgments We would like to thank the staff of the three SMEs who completed our GTM survey.

This work was supported, in part, by Science Foundation Ireland grants 10/CE/I1855 and 13/RC/2094 to Lero - the Irish Software Research Centre (www.lero.ie), and by Enterprise Ireland and the European Development Fund through grants IR20130022 and CF20144348.

REFERENCES

[1] I. Richardson, G. Avram, S. Deshpande, and V. Casey, "Having a foot on each shore-bridging global software development in the case of SMEs," in *2008 IEEE International Conference on Global Software Engineering*. IEEE, 2008, pp. 13–22.

[2] A. Boden, B. Nett, and V. Wulf, "Coordination practices in distributed software development of small enterprises," in *International Conference on Global Software Engineering (ICGSE 2007)*, Aug 2007, pp. 235–246.

[3] T. Clear, S. Beecham, J. Barr, M. Daniels, R. McDermott, M. Oudshoorn, A. Savickaite, and J. Noll, "Challenges and recommendations for the design and conduct of global software engineering courses: A systematic review," in *Proceedings of the Working Group Reports of the 2015 on Innovation & Technology in Computer Science Education Conference*, N. Ragonis and P. Kinnunen, Eds. New York: ACM, 2015, pp. 1–39.

[4] I. Richardson, S. Moore, D. Paulish, V. Casey, and D. Zage, "Globalizing software development in the local classroom," in *20th Conference on Software Engineering Education & Training, CSEET '07*, 2007, pp. 64–71.

[5] I. Richardson, V. Casey, F. McCaffery, J. Burton, and S. Beecham, "A process framework for global software engineering teams," *Information and Software Technology*, vol. 54, no. 11, pp. 1175–1191, 2012.

[6] S. Beecham, T. Clear, D. Damian, J. Barr, J. Noll, and W. Scacchi, "How best to teach global software engineering? Educators are divided," *IEEE Software*, vol. 34, no. 1, pp. 16–19, Jan 2017.

[7] A. Peters, W. Hussain, A. Cajander, T. Clear, and M. Daniels, "Preparing the global software engineer," in *Proceedings 2015 IEEE 10th International Conference on Global Software Engineering*, M. Nordio and B. Al-Ani, Eds. Los Alamitos, California: IEEE, 2015, pp. 61–70, 26-29 August.

[8] S. Beecham, T. Clear, J. Barr, M. Daniels, M. Oudshoorn, and J. Noll, "Preparing tomorrow's software engineers for work in a global environment," *IEEE Software*, vol. 34, no. 1, pp. 9–12, Jan 2017.

[9] OECD, "Small and medium-sized enterprises: Local strength, global reach," Organisation for Economic Co-operation and Development, Tech. Rep., 2000.

[10] MBIE, "New Zealand sectors report series ICT summary report, 2015," Ministry of Business Innovation and Employment, Tech. Rep., 2015 2015.

[11] A. Raninen, "Practical process improvement: How to initiate software process improvement in a small compnay," Ph.D. dissertation, School of Computing, University of Eastern Finland, 2015.

[12] European Commission, "European commission annual report on European SMEs 2015/2016 - executive summary," European Commission, Tech. Rep., 2016.

[13] J. Bass, S. Beecham, J. Noll, and M. A. Razzak, "All hands to the pumps: The product owner role in small companies," Lero - The Irish Software Research Centre, Report, 2017. [Online]. Available: http://www.lero.ie/sites/default/files/Lero_TR_2017_01_Bass_Beecham_Noll_Razzak_AllHandsToThePump.pdf

[14] W. Hussain and T. Clear, *Spreadsheets as Collaborative Technologies in Global Requirements Change Management*. Los Alamitos, California: IEEE, 2014, pp. 74–83, 26-29 August.

[15] F. J. Pino, F. Garcia, and M. Piattini, "Software process improvement in small and medium software enterprises: a systematic review," *Software Quality Journal*, vol. 16, no. 2, pp. 237–261, 2008.

[16] A. Raninen, H. Merikoski, J. Ahonen, and S. Beecham, "Applying software process modeling to improve customer support processes," *Journal of Software: Evolution and Process*, vol. 27, no. 4, pp. 274–293, 2015, jSME-13-0159.R1. [Online]. Available: http://dx.doi.org/10.1002/smr.1713

[17] R. V. O'Connor, *Quantitative Multidisciplinary Approaches in Human Capital and Asset Management*. Hershey, PA: IGI Global, 2015, ch. Towards an Understanding of Team Dynamics in Very Small Enterprises: An Exploratory Study in Software, p. 42.

[18] B. Bruegge, S. Krusche, and L. Alperowitz, "Software engineering project courses with industrial clients," *Trans. Comput. Educ.*, vol. 15, no. 4, pp. 1–31, 2015.

[19] D. Boud, *Problem Based Learning in Perspective*. Sydney: Higher Education Research Society Of Australia, 1985, pp. 13–18.

[20] D. Damian, A. Hadwin, and B. Al-Ani, *Instructional design and assessment strategies for teaching global software development: a framework*. Shanghai, China: ACM, 2006, pp. 685–690.

[21] M. Nordio, C. Ghezzi, B. Meyer, E. Di Nitto, G. Tamburrelli, J. Tschannen, N. Aguirre, and V. Kulkarni, "Teaching software engineering using globally distributed projects: the dose course," in *Proceedings of the 2011 Community Building Workshop on Collaborative Teaching of Globally Distributed Software Development*. ACM, 2011, pp. 36–40.

[22] M. Daniels, M. Petre, V. Almstrum, L. Asplund, C. Bjorkmann, C. Erickson, B. Klein, M. Last, and A. Berglund, "Runestone, an international student collaboration project," in *IEEE Frontiers in Education Conference*. IEEE, 1998.

[23] T. Clear, "A collaborative learning trial between New Zealand and Sweden - using Lotus Notes Domino in teaching the concepts of human computer interaction," in *The 4th Annual Conference On Innnovation & Technology In Computer Science Education*, T. Wileusz, Ed., vol. 1. ACM, 1999, pp. 111 – 114.

[24] D. Petkovic, R. Todtenhoefer, and G. Thompson, "Teaching practical software engineering and global software engineering: Case study and recommendations," in *36th Annual Frontiers in Education Conference*, 2006, pp. 19–24.

[25] I. Bosnić, I. Čavrak, M. Žagar, R. Land, and I. Crnković, "Customers' role in teaching distributed software development," in *23rd IEEE Conference on Software Engineering Education and Training (CSEE&T)*, 2010, pp. 73–80.

[26] M. Paasivaara, K. Blincoe, C. Lassenius, D. Damian, J. Sheoran, F. Harrison, P. Chhabra, A. Yussuf, and V. Isotalo, "Learning global agile software engineering using same-site and cross-site teams," in *Proceedings of the 37th International Conference on Software Engineering-Volume 2*. IEEE Press, 2015, pp. 285–294.

[27] A. Cajander, T. Clear, and M. Daniels, "Introducing an external mentor in an international open ended group project," in *39th ASEE/IEEE Frontiers in Education Conference Proceedings*. San Antonio, Texas: IEEE, 2009, pp. T2D1–T2D6.

[28] M. Daniels, A. Cajander, A. Pears, and T. Clear, "Engineering education research in practice: Evolving use of open ended group projects as a pedagogical strategy for developing skills in global collaboration (special issue on applications of engineering education research)," *International Journal of Engineering Education*, vol. 26, no. 4, pp. 795–806, 2010.

[29] S. Beecham, I. Richardson, and J. Noll, "Assessing the strength of global teaming practices: A pilot study," in *IEEE 10th International Conference on Global Software Engineering (ICGSE)*, Ciudad Real, Spain, July 2015. [Online]. Available: http://dx.doi.org/10.1109%2FICGSE.2014.19

[30] T. Clear, J. Whalley, J. Hill, Y. Liu, A. Pears, and B. Plimmer, *A Global Software Project: Developing a Tablet PC Capture Platform for Explanograms*, Koli, Finland, 2008.

[31] T. Clear, "Diagnosing your teaching style: How interactive are you?" *ACM Inroads*, vol. 1, no. 2, pp. 34–42, 2010.

[32] "Keynotes," in *2016 IEEE 11th International Conference on Global Software Engineering Workshops (ICGSEW)*, Aug 2016, pp. xv–xvi.

[33] F. Matthes, C. Neubert, C. Schulz, C. Lescher, J. Contreras, R. Laurini, B. Rumpler, D. Sol, and K. Warendorf, "Teaching global software engineering and international project management-experiences and lessons learned from four academic projects," in *CSEDU (2)*, 2011, pp. 5–15.

TABLE IV. MAPPING OF GSE-ED RECOMMENDATIONS TO GTM PRACTICES.

GSE-Ed Theme	GSE-Ed Recommendation
GTM Practice C2: Provide training to ensure that global team has required understanding of the customer base and the business functions to take full advantage of the proximity of the team to the customer base.	
Stakeholder Role	Use student contests as an alternative form of external customer. Use simulators in lieu of actual clients. The customer involved should be a company representative who can spend time with the students discussing the project proposal and status. Use a resident coach to actively mentor the local team and make sure that the project lives up to its expected outcomes for the customer.
GTM Practice F3: If working in regulatory domain, provide training on regulatory requirements and procedures.	
(none)	
GTM Practice G5: Identify issues from lessons learned that require a wider initiative such as a change in organizational culture and report to a global change management agent.	
Global Distance	Harmonize processes across institutions and establish a clear line of responsibility. Project leaders should agree to a course specific set of terms. Be flexible in accepting different rules and habits. Students should be selected by the instructor based on a student profile to ensure a balanced team.
Curriculum/Pedagogy	Encourage reflective attitudes in students. Use reflection to develop students insight into their own learning and make the learning from the course more explicit. Inculcate the habits of a reflective practitioner through a final phase of reflection about what the student has experienced.
GTM Practice L1: All potential risks should be identified and addressed to include: risks in misunderstanding cultural differences, misunderstanding requirements, feature volatility, schedules, budgets, personnel.	
Global Distance	Team members should be flexible about meeting times. Schedules should be shared to include working hours of the team. A mandatory project communication plan should be drawn up. Teammates should be told if the student cannot attend a meeting or will be unable to answer email. Use a mix of synchronous and asynchronous communication organised at regular intervals.
People/Soft Issues	Teams should travel to other sites. Use icebreaking sessions, and fun moments during the course, to break the serious course atmosphere. Move beyond formal introductions to deeper informal interaction between the team members. Address cultural differences through assignments comparing cultures.
Stakeholder Role	Students should have the appropriate background before entering the course. The instructor must be intimately involved with the teams and projects, and must provide clear and predictable guidance. Provide the students with guidelines for all tasks and deliverables. Instructors should actively assist students in developing project goals and creating architectural designs. Instructors must be flexible and anticipate change.
Curriculum/Pedagogy	Establish the course schedule incorporating regular deadlines. Conduct pre-semester GSD training sessions and a crisp preparatory GSE overview with a project management focus. At the beginning, present past courses and typical challenges experienced. Explain to the students the rationale behind vague requirements, to minimise student frustration. Sponsors and tutors should consistently state the main objective right at the beginning. Focus on the process before the project topics and tools. Projects must be testable with the time available. Keep project scope to three months with prior defined outcomes. Align project with the sponsor's needs. Deliver a complex software system for a real client. Do not try to run disparate projects with the latest technologies until the underlying process works. Ensure real-life projects are deployable and sustainable.
GTM Practice M2: When defining the global strategy for dealing with conflict, different types of conflict have to be taken into account, for example conflict due to fear as well as cultural differences.	
People/soft issues	Get the local and remote teams to be familiar with each other as soon as possible. Keep the students highly motivated. Remember: we are different. Be enthusiastic.
GTM Practice O2: Ensure that relevant team members are made aware of how and when they will receive inputs to products they are working on, and when they need to distribute outputs from these products and when complete work products are required.	
Teamwork	Create a responsibility chart comprised of the tasks students must complete to fulfill their role in the course. Ensure clear and constant communication to promote visibility, better decision-making, understanding of goals, and team synergy. Start communication early to have a positive effect on synergy. Get the local and remote teams to be familiar with each other as soon as possible.
Curriculum/Pedagogy	Focus on the process before the project topics and tools. Establish the course schedule incorporating regular deadlines.
Development Process	Manage merge conflicts when integrating software. Use design by contract to specify module interfaces. Require mandatory code review of interfaces before proceeding to implementation. Partition designs for independent development. Document design decisions and rationale to facilitate knowledge transfer to other teams.
GTM Practice Q5: Acknowledging team success may require tailoring rewards to the needs of different cultures.	
Global Distance	Identify the cultural and educational differences between the students in different locations, and have students learn from each other with respect to culture.
People/Soft Issues	Use learning agreements to help students to be conscious of the broader learning goals of a GSE course. Teams should travel to other sites. Use competitions to motivate students, especially when student teams work on a product defined by an external customer from a foreign university.
Curriculum/Pedagogy	Design an assessment process tailored to GSE, with rules adapted from GSE practice. Inform learners about assessment objectives. In grading, emphasize the entire software lifecycle. Identify the learners' starting skill set by self-assessment. Define three deliverables for evaluation: initial presentation, final presentation and final report. Assign grades based on a thorough analysis and testing of the final product. Award higher marks to students who invest more effort.
GTM Practice T4: Establish procedures to coordinate implementation of contingencies when and if required.	
Global Distance	Start communication by brute force if necessary.
Stakeholder Role	Require regular and frequent team and student status reporting and monitoring that is more intensive at start.
Infrastructure	Student teams should be required to write a communication plan in order to encourage frequent, effective communication.

TABLE V. STAKEHOLDER PARTICIPATION IN GLOBAL SOFTWARE ENGINEERING EDUCATION PRACTICES.

Global Software Engineering Education practice	Instructor as global lead	Instructors as a group	Instructor as individual	Student	Client
GSE-Ed Theme: Curriculum/Pedagogy					
Align project with the sponsor's needs.	instigator			passive	committed
Assign grades based on a thorough analysis and testing of the final product.			instigator	committed	active
At the beginning, present past courses and typical challenges experienced			instigator	passive	not involved
Award higher marks to students who invest more effort.			instigator	passive	not involved
Conduct pre-semester GSE training sessions and a crisp preparatory GSE overview with a project management focus			instigator	active	not involved
Define three deliverables for evaluation: initial presentation; final presentation and final report	instigator				
Deliver a complex software system for a real client.	instigator			active	committed
Design an assessment process tailored to GSE, with rules adapted from GSE practice.			instigator	passive	not involved
Do not try to run disparate projects with the latest technologies until the underlying process works.	instigator			passive	committed
Encourage reflective attitudes in students.			instigator	committed	not involved
Ensure real-life projects are deployable and sustainable.	instigator			passive	committed
Establish the course schedule and incorporate regular deadlines.	instigator			committed	not involved
Explain to the students the rationale behind vague requirements, to minimise student frustration.			instigator	active	passive
Focus on the process before the project topics and tools.			instigator	committed	passive
Identify the learners' starting skill set by self-assessment.			instigator	committed	not involved
In grading, emphasize the entire software lifecycle.			instigator	passive	not involved
Inculcate the habits of a reflective practitioner through a final phase of reflection about what the student has experienced.			instigator	committed	not involved
Inform learners about assessment objectives.			instigator	passive	not involved
Keep project scope to three months with prior defined outcomes.	instigator			active	committed
Projects must be testable with the time available.	instigator			committed	active
Sponsors and tutors should consistently state the main objective right at the beginning.			instigator	passive	committed
Use reflection to develop students insight into their own learning and make the learning from the course more explicit.			instigator	committed	not involved
GSE-Ed Theme: Development Process					
Document design decisions and rationale to facilitate knowledge transfer to other teams.			passive	instigator	not involved
Manage merge conflicts when integrating software			passive	instigator	not involved
Partition designs for independent development			active	instigator	passive
Require mandatory code review of interfaces before proceeding to implementation.			instigator	committed	not involved
Use design by contract to specify module interfaces.			passive	instigator	not involved
GSE-Ed Theme: Global Distance					
A mandatory project communication plan should be drawn up.			instigator	committed	active
Be flexible in accepting different rules and habits.		instigator		committed	active
Harmonize processes across institutions and establish a clear line of responsibility.		instigator		passive	active
Identify the cultural and educational differences between the students in different locations, and have students learn from each other with respect to culture.		instigator		committed	not involved
Project leaders should agree to a course specific set of terms.		instigator		active	passive
Schedules should be shared to include working hours of the team.			committed	instigator	active
Start communication by brute force if necessary.			instigator	committed	passive
Students should be selected by the instructor based on a student profile to ensure a balanced team		instigator		passive	not involved
Team members should be flexible about meeting times.			committed	instigator	active
Teammates should be told if the student cannot attend a meeting or will be unable to answer email.			committed	instigator	active
Use a mix of synchronous and asynchronous communication organised at regular intervals.			instigator	committed	active
GSE-Ed Theme: Infrastructure					
Student teams should be required to write a communication plan in order to encourage frequent, effective communication.			instigator	committed	passive
GSE-Ed Theme: People/Soft Issues					
Address cultural differences through assignments comparing cultures.			instigator	committed	not involved
Be enthusiastic.			instigator		
Get the students to be familiar with each other as soon as possible.		instigator		committed	not involved
Keep the students highly motivated.			instigator	committed	not involved
Move beyond formal introductions to deeper informal interaction between the team members.			instigator	committed	passive
Remember: we are different.			instigator	committed	active
Teams should travel to the other sites.		instigator		committed	not involved
Use competitions to motivate students, especially when student teams work on a product defined by an external customer from a foreign university.	instigator			committed	not involved
Use icebreaking sessions, and fun moments during the course, to break the serious course atmosphere.			instigator	committed	not involved
Use learning agreements to help students to be conscious of the broader learning goals of a GSE course			instigator	committed	not involved
GSE-Ed Theme: Stakeholder Role					
Use student contests as an alternative form of external customer.	instigator			committed	not involved
Instructors must be flexible and anticipate change.			instigator	passive	not involved
Instructors should actively assist students in developing project goals and creating architectural designs.			instigator	committed	passive
Provide the students with guidelines for all tasks and deliverables.		instigator		active	passive
Require regular and frequent team and student status reporting and monitoring that is more intensive at start.			instigator	committed	passive
Students should have the appropriate background before entering the course			instigator	committed	not involved
The customer involved should be a company representative who can spend time with the students discussing the project proposal and status.		instigator		active	committed
The instructor must be intimately involved with the teams and projects, and must provide clear and predictable guidance.			instigator	active	passive
Use a resident coach to actively mentor the local team and make sure that the project lives up to its expected outcomes for the customer.		instigator		committed	not involved
Use simulators in lieu of actual clients.	instigator			committed	not involved
GSE-Ed Theme: Teamwork					
Create a responsibility chart comprised of the tasks students must complete to fulfill their role in the course.	instigator			committed	passive
Ensure clear and constant communication to promote visibility, better decision-making, understanding			instigator	committed	passive