2023 International Symposium on Educational Technology (ISET) | 979-8-3503-0491-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISET58841.2023.00054

Learning Gaps in Project-based Requirements Engineering Education - A Case Study of Student Projects

Yishu Li
Dept of Computer Science
City University of Hong Kong
Hong Kong, China
yishuli5-c@my.cityu.edu.hk

Jacky Keung
Dept of Computer Science
City University of Hong Kong
Hong Kong, China
Jacky.Keung@cityu.edu.hk

Xiaoxue Ma
Dept of Computer Science
City University of Hong Kong
Hong Kong, China
xiaoxuema3-c@my.cityu.edu.hk

Jingyu Zhang
Dept of Computer Science
City University of Hong Kong
Hong Kong, China
jzhang2297-c@my.cityu.edu.hk

Zhen Yang
Dept of Computer Science
City University of Hong Kong
Hong Kong, China
zhyang8-c@my.cityu.edu.hk

Shuo Liu
Dept of Computer Science
City University of Hong Kong
Hong Kong, China
sliu273-c@my.cityu.edu.hk

***Abstract*——[Context and motivation] Learning Requirements Engineering is one of the key activities in teaching Software Engineering. Training computer science students to appreciate the importance of requirement engineering in practice is essential but also a difficult task. [Question/problem] In software requirements engineering education courses, project-based practice is commonly adopted as the pedagogical approach. However, feedback shows that projects taken in the class resulted in poorly translated requirements engineering skill sets required in the industry. [Principal ideas/results] This paper is to investigate and illustrate the student learning gaps observed in project-based requirements engineering education. Data containing software requirements were collected from 72 Computer Science student projects. The grounded theory approach had been systematically adopted, and keywords classified by requirements engineering activities were extracted from assessment comments. Based on the analysis of the comment keywords and report details, the issues exposed in student projects have been observed and summarized. Integrating with existing requirements engineering research, we discuss the causes from the teachers' perspective. The leading causes of students' underperformance in requirements engineering education learning outcomes are the lack of the industry environment, the lack of domain knowledge, and the lack of problem-solving skills. [Contribution] This paper contributes to the current educational effort in preparing students for the requirements engineering and software engineering community.**

Keywords—requirements engineering education, learning gaps, student projects, case study

I. Introduction

Requirements engineering (RE) is the foundation of high-quality software [1], which serves as a profession and a vital part of the software engineering (SE) lifecycle [2]. Therefore, teaching RE at the university level is an important endeavor [3]. Requirements engineering education (REE) targets to cover three topics related to RE activities: analysis and modeling skills, interviewing skills, and writing skills [4].

Memon, et al. [5] reported four significant challenges in REE: lack of interest in the RE course, insufficient practice in RE activities, lack of training after the course, and difficulties of RE knowledge application in the real-world context. In a recent systematic literature review of REE [6], the authors observed that contemporary REE literature largely ignored the systematic application of pedagogy. Correctly understanding the problems exposed in REE is essential to apply correct pedagogy in REE.

The overall objective of this paper is to investigate and illustrate the learning gaps in project-based REE. We conducted a case study on student projects in the software design course at City University of Hong Kong. We collected 72 Computer Science student projects and assessment comments. Categorized by RE activities, keywords were extracted from assessment comments. By analyzing the keywords and project report details, we observed the problems exposed in student projects and analyzed the causes from teachers' perspectives. Through learning these problems in depth, educators can make future efforts to reduce the gaps between student performance and industrial expectations.

The rest of the paper is organized as follows. Section II presents the overview of REE and project-based learning. Section III presents the methodology of our case study. Section IV summarizes the problems from our observation, and Section V provides a discussion on the causes from teachers' perspectives. Finally, Section VI concludes our study and discusses future work.

II. Related Work

This section briefly discusses an overview of REE and project-based learning in REE.

A. Requirements Engineering Education

In the past few years, there has been growing attention on incorporating RE into software engineering courses for undergraduate students [7]. Hence, several REE papers were published in the SE and RE literature. Davis, et al. [8] proposed a framework to integrate RE research and education to improve REE performance. The integration of RE research and REE increases the connection between RE practices and REE. Therefore, the best RE practices can be identified and fed into project practice. As revealed in the systematic mapping study of [9] conducted in 2012, it is relatively widespread for educating SE curricula at the undergraduate level, and with REE gains increased focus.

979-8-3503-0491-6/23/$31.00 ©2023 IEEE
DOI 10.1109/ISET58841.2023.00054

In the latest systematic literature review of REE [6] conducted in 2022, the authors reported the typical trends in RE instruction of the past decade, which contains involving realistic stakeholders in RE courses, taking elicitation as an RE activity teaching predominantly, and focusing more on student motivation or communication skills in teaching.

B. *Project-based learning in Requirements Engineering Education*

Project-based learning (PBL) is an active learning pedagogical methodology wherein students acquire knowledge and develop skills through participating in projects with real-world context [10]. Usually, the projects are conducted in groups. Gonzalez-Rubio, et al. [11] reported that PBL could help students to gain different competencies (e.g., problem-solving and innovative design). Svensson and Regnell [4] investigated the effect of adopting role-playing as a pedagogical tool in REE. Four cases were studied to investigate the correlation in grades between role-playing projects and written exams. The results showed that role-playing results in better performance in student projects but make no significant difference in the theory examination part.

Nwokeji, et al. [12] observed that PBL helps improve students' REE performance. They conducted a study to investigate how PBL was applied to a RE course by analyzing aggregate student performance affected by the implementation of PBL. The results revealed that PBL benefits RE courses, such as enhancing students' performance. However, few publications have investigated the learning gaps exposed in project-based REE.

III. Methodology

This research aims to evaluate the learning gaps of project-based REE. To achieve this, we use a case study research mode to explore student performance on REE. The case study comprises 72 student projects in a software design course at City University of Hong Kong.

A. *Context and Data Collection*

Data were collected from 72 student projects in Course Software Design in 2020/2021 and 2021/2022 Semester B. TABLE I. shows the distribution of the student projects. Fig. 1. presents the topic distribution of collected projects.

TABLE I. Distribution of Collected Student Projects

Academic Year	No. of Student Projects
2020/2021 Semester B	37
2021/2022 Semester B	35

The course setup is equivalent for all students. REE is incorporated into the software design course. The skills and tools utilized in RE activities are guided in the course. Students are required to conduct a group project to design a software product from Week 1 in the course. Students can form a group with sizes of 4 to 6. Each group can decide and select a project topic to complete the software design task from a requirement perspective. In Week 13, the students should present and submit a project report with requirement-related documents (i.e., software requirement specification, software prototype, use case diagram, and project management activity logs). Our observation and analysis are based on the collected project reports and assessment comments.

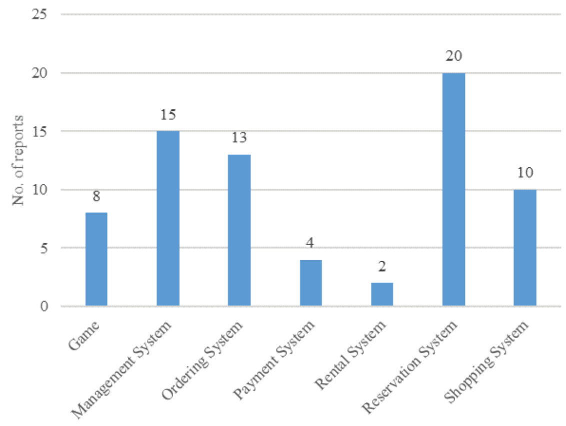

Fig. 1. Categories of student project topics

B. *Data Analysis*

We extracted the requirement-related assessment comments and adopted the grounded theory approach [13] to analyze the issues exposed in the student projects. The problem-related keywords were extracted from the assessment comments. The keywords were categorized according to the requirements activities which control the software quality [14]. The keywords were further reviewed and analyzed. The properties and dimensions of the exposed core issues were sorted out. Finally, the interrelationships of the problems associated with keywords in the student projects were identified.

C. *Validity Threats*

According to ACM SIGSOFT empirical standards [15], external validity in a case study refers to the ability to generalize the findings. One threat to external validity in this study is that only one course was evaluated. Thus, we described the course setup in detail, which can help to judge the transferability of the setup to other cases and evaluate cases with similar contexts.

IV. Evaluations

In the student projects conducted in this course, three requirement activities (i.e., requirements elicitation, requirements modeling, and requirements management) were mainly applied for software design. When assessing the project reports, the teachers evaluated the quality of the requirements from these three activities. The comments contain some keywords that reflect the student's learning performance. We extracted the keywords from the assessment comments and categorized them by requirement activities.

In this section, based on analyzing the student project reports and assessment comments, we summarized four typical problems exposed in student projects. TABLE II presents keywords extracted from assessment comments and the number of reports exposed with keyword-related problems. TABLE III lists the detailed percentages of underperformance projects for different project topics.

TABLE II. KEYWORDS EXTRACTED FROM ASSESSMENT COMMENTS, CATEGORIZED BY REQUIREMENT ACTIVITIES

Requirements Activities	Keyword	No. of report commented as underperformance	Percentage
Requirements Elicitation	comprehensive	53	73.61%
	security	55	76.39%
	non-functional	49	68.06%
Requirements Modeling	clear	34	47.22%
Requirements Management	risk	32	44.44%
	conflict	67	93.06%

Problem 1: Missing level of details. Comprehensibility is one of the RE quality attributes, referring to the specified requirements so all stakeholders can understand [14]. Reviewing the keywords related to the activity of requirement elicitation, we found only 19 reports got the comments of *comprehensive requirements* in the assessment comments. That is, 53 reports (73.61%) were exposed to the issue of lacking detailed requirements. From our observation of the report details, we found that the requirement specifications in student projects focus more on high-level functional descriptions but neglect the details. Similarly, these project groups did not provide hierarchical use case diagram models with details in their submissions.

Reviewing the results by different project topics, we found the topic with management systems performed best in describing the comprehensiveness of the requirements. Only 53.33% of project teams were classified as underperformance in this category, accounting for the lowest percentage among all project topics.

TABLE III. PERCENTAGES OF PROJECTS COMMENTED AS UNDERPERFORMANCE, CATERGARIZED BY DIFFERENT PROJECT TOPICS

Project Topics	No.	Percentage of Underperformance					
		comprehensive	*security*	*non-functional*	*clear*	*risk*	*conflict*
Game	8	75.00%	75.00%	75.00%	37.50%	62.50%	87.50%
Management System	15	**53.33%**	**66.67%**	**53.33%**	40.00%	40.00%	93.33%
Ordering System	13	84.62%	92.31%	69.23%	61.54%	**30.77%**	92.31%
Payment System	4	75.00%	75.00%	75.00%	100.00%	0.00%	100.00%
Rental System	2	100.00%	100.00%	100.00%	100.00%	50.00%	**50.00%**
Reservation System	20	75.00%	75.00%	75.00%	**30.00%**	55.00%	100.00%
Shopping System	10	80.00%	70.00%	60.00%	50.00%	50.00%	90.00%

Problem 2: Neglecting non-functional requirements. Performance, reliability, maintainability, usability, security, privacy, and customer satisfaction are well-studied non-functional requirements (NFRs). In the systematic review of REE [6], the authors highlighted safety and security requirements did not receive sufficient attention in REE studies. Similar results are exposed in our study. All selected topics shown in Fig. 1 rely on security and performance because of the system's nature. However, Table II shows 68.06% of the projects neglect the NFRs, and 76.39% of them miss consideration on security requirements. We reviewed the rest 23 reports which covered discussion on NFRs. The results reveal only 16 of them integrated NFRs into design properly, and the remaining 7 projects classified NFRs incorrectly.

Regarding the differences among different project topics, the projects of management systems covered more discussions on security and non-functional requirements, with percentages of underperformance of 66.67% and 53.33%, respectively. By contrast, none of the project teams working on the rental systems took NFRs into account in their RE design.

Problem 3: Vague requirements. The IEEE requirements quality standard [16] defines unambiguity as only having one possible interpretation in requirements descriptions [14]. Reviewing the learning outcome of 72 project reports, we found 47.22% of them were commented as *not clear*. For example, based on their requirement specification, different use case diagrams can be generated under a specific curriculum. This reveals the project requirements are not sharing the same understanding among different parties.

From the perspective of a clear understanding of requirements modeling, the reservation system projects performed best. However, the payment and rental system projects performed worst. None of these projects were commented as *clear*.

Problem 4: Poor risk management. As reported in [6], students tend to forget risk management in most project-based approaches. For academic projects, some students are easily ignored risk control [6]. Similarly, 44.44% of projects in our study never discuss risk management for the RE process. And 93.06% of them never discuss the requirement conflict in the reports. Reviewing the activity logs in student submissions, we found only a few projects reported how to manage the requirements changes and how to manage the conflicts.

The payment and rental systems project teams were concerned more with risk management. All payment system project teams had covered the discussion of requirement risk management, while half of the rental system project teams covered the discussion of solving requirement conflicts in their design.

V. DISCUSSION

Looking at the problems the student projects exposed, gaps exist in student performance and industry expectations.

To sustain and improve student performance on project-based REE, the causes need to be correctly understood. Using our collected data and preliminary results on student performance, we summarize and analyze the causes in this section.

Lack of industry environment. Regev, et al. [3] pointed out that the "*sterile*" nature of academic classroom projects makes them translate poorly to the industry, which inadequately reflects industrial practices. The lack of industrial practices in RE activities makes the students more easily ignore exercise and understanding in acquiring RE skills.

The student projects were conducted in one semester, which limits the product size. Such time-limited project experience makes it insufficient for students to apply the learned skill sets. The reported Problem 2 shows over half of the student projects neglected the NFRs, regardless of the project topics. Such issues are not easily exposed to a small-size product. Meanwhile, students tended to discuss only the high level of requirements instead of studying a product in detail, as reported in Problem 1. This is also affected by time-limited practices.

Lack of domain knowledge. Another limitation of academic project-based learning is the domain knowledge of students. The students were trained while implementing the projects. They did not have prior knowledge, and it was hard for them to build a knowledge system in RE before starting the project, which made them miss the consideration of details and results in Problem 1.

Reviewing the reports, some project teams adopted roles playing techniques to conduct the requirements elicitation process. The students played as stakeholders during requirements negotiation. However, the lack of domain knowledge in business flow leads to vague requirements issues (as reported in Problem 3). The students did not have a business background in requirements elicitation, which makes such gaps hard to be identified. This problem is evident for project teams working on payment and rental systems, none of which were classified as *clear* (with 100% underperformance, as presented in Table III). These two topics require more understanding of financial business flow, which makes the gaps more obvious.

Lack of problem-solving skills. Problem-solving is the core skill of RE. Ouhbi, et al. [17] reported the importance of solving skills and pointed out such skills become influential for requirement engineers. In the study of [6], the observation results reveal students can safely fail in practice. That is, the students explore solution alternatives without a grade penalty for being wrong or without threatening project success [6]. The low-stakes environment makes students provide poor problem-solving skills in REE projects.

The activity logs show that most students did not provide a buffer for risk management. Unlike the requirement engineers well trained in the industry, most undergraduate students lack a sense of risk control in our study, which results in Problem 4 as reported.

VI. CONCLUSION

REE attracts increasing emphasis on educating high-quality RE engineers. This study collected 72 student projects from a software design course and explored the learning gaps in the project-based REE. Through evaluating the keywords extracted from assessment comments, we summarized four problems exposed to student projects: missing level of detail, neglecting of non-functional requirements, vague requirements, and poor risk management. These issues reveal gaps exist between student learning outcomes and industry expectations.

In order to improve student performance on project-based REE and provide higher quality REE, we analyzed and reported the causes from teachers' perspectives. Three aspects mainly cause poor learning outcomes in project-based REE. The lack of an industry environment cannot offer students a more practical experience. And the lack of domain knowledge leads to insufficient well-trained techniques applied in the projects. Moreover, undergraduate students easily ignore the sense of risk control and problem-solving skills in RE practices, which results in poor RE risk management.

Based on the reflection on the problems exposed in the project-based REE, we intend to investigate further how RE research can integrate and help with high-quality REE in future studies. To narrow the gap between REE and industrial RE practices, we plan to invite some industrial guests to give talks or guidance to students. Therefore, the RE techniques embedded in real-world practices can be disseminated in REE, and students can acquire better RE skill sets as practitioners.

ACKNOWLEDGMENT

This work is supported in part by the General Research Fund of the Research Grants Council of Hong Kong and the research funds of the City University of Hong Kong (6000796, 9229109, 9229098, 9220103).

REFERENCES

[1] M. Ochodek and S. Kopczyńska, "Perceived importance of agile requirements engineering practices–a survey," Journal of Systems and Software, vol. 143, pp. 29-43, 2018.

[2] B. Berenbach, "A hole in the curriculum," in International workshop on Requirements Engineering Education & Training, 2005.

[3] G. Regev, D. C. Gause, and A. Wegmann, "Requirements engineering education in the 21st century, an experiential learning approach," in 2008 16th IEEE international requirements engineering conference, 2008: IEEE, pp. 85-94.

[4] R. B. Svensson and B. Regnell, "Is role playing in Requirements Engineering Education increasing learning outcome?," Requirements Engineering, vol. 22, pp. 475-489, 2017.

[5] R. N. Memon, R. Ahmad, and S. S. Salim, "Problems in requirements engineering education: a survey," in Proceedings of the 8th international conference on frontiers of information technology, 2010, pp. 1-6.

[6] M. Daun, A. M. Grubb, V. Stenkova, and B. Tenbergen, "A systematic literature review of requirements engineering education," Requirements Engineering, pp. 1-31, 2022.

[7] N. Yusop, Z. Mehboob, and D. Zowghi, "The role of conducting stakeholder meetings in requirements engineering training," Proc. of REET, vol. 7, pp. 48-55, 2007.

[8] A. M. Davis, A. M. Hickey, and A. Chamillard, "Moving beyond the classroom: Integrating requirements engineering research & education to improve practice," in Proceedings of the 1st international workshop on requirements engineering education and training, REET, 2005, vol. 5: Citeseer, pp. 78-87.

[9] B. Malik and S. Zafar, "A systematic mapping study on software engineering education," International Journal of Educational and Pedagogical Sciences, vol. 6, no. 11, pp. 3343-3353, 2012.

[10] J. C. Nwokeji and P. S. T. Frezza, "Cross-course project-based learning in requirements engineering: An eight-year retrospective,"

in 2017 IEEE Frontiers in Education Conference (FIE), 2017: IEEE, pp. 1-9.

[11] R. Gonzalez-Rubio, A. Khoumsi, M. Dubois, and J. P. Trovao, "Problem-and project-based learning in engineering: A focus on electrical vehicles," in 2016 IEEE Vehicle Power and Propulsion Conference (VPPC), 2016: IEEE, pp. 1-6.

[12] J. C. Nwokeji, F. Aqlan, A. Olagunju, T. Holmes, and N. C. Okolie, "WIP: Implementing project based learning: some challenges from a requirements engineering perspective," in 2018 IEEE Frontiers in Education Conference (FIE), 2018: IEEE, pp. 1-5.

[13] A. Strauss and J. M. Corbin, Grounded theory in practice. Sage, 1997.

[14] C. Denger and T. Olsson, "Quality assurance in requirements engineering," Engineering and managing software requirements, pp. 163-185, 2005.

[15] P. Ralph et al., "ACM SIGSOFT empirical standards," 2020.

[16] I. C. S. S. E. S. Committee and I.-S. S. Board, IEEE recommended practice for software requirements specifications (no. 1998). IEEE, 1998.

[17] S. Ouhbi, A. Idri, J. L. Fernández-Alemán, and A. Toval, "Requirements engineering education: a systematic mapping study," Requirements Engineering, vol. 20, pp. 119-138, 2015.