2021 ASEE ANNUAL CONFERENCE
Virtual Meeting | July 26–29, 2021 | Pacific Daylight Time
Paper ID #33294

A Study on the Impact of Using Industry Standard Tools and Practices on Software Engineering Courses Projects

Dr. Tajmilur Rahman, Gannon University

Tajmilur Rahman PhD, is an assistant professor in the department of Computer and Information Science at Gannon University in Erie, Pennsylvania. His overarching research interest is to investigate release engineering practices in software systems. His research works are driven by the desire to determine the empirical factors that lead to a successful software development and release. His research interests also include understanding the significance of software architecture for long lasting software systems and providing tool support to the community to nurture software feature architecture. Dr. Rahman is the first author who extracted feature-architecture while understanding the best practices of feature management and its impact on software architecture, which is another major research interest of his. Understanding and visualizing feature architecture is necessary to advance software development and engineering by maintaining a controlled architectural growth of software systems. Dr. Rahman is currently focusing on the following research areas: software feature-architecture, release management in trunk based development, software quality in trunk-based rapid-release cycles, developing knowledge-base to predict potential architectural drift using artificial intelligence. Tajmilur Rahman was a doctoral student at Concordia University. As a doctoral student he worked on empirical understanding on software projects with various lengths of release cycle, he extracted feature-architecture of Google Chrome web browser, he updated the 12 years old browser reference architecture and also he was working on software quality due to post-release bugs in a trunk based development environment. He received his PhD In 2018.

Dr. Stephen T. Frezza, Gannon University

Deacon Steve Frezza, PSEM is a professor of Software Engineering and chair of the Computer and Information Science department at Gannon University in Erie, PA. His research interests include Global Software Engineering, Affective Domain Learning, Engineering Education Research, as well as Philosophy of Engineering and Engineering Education. He is regularly involved in supporting the regional entrepreneurial ecosystem, as well as projects that serve the regional community. He is an active member and volunteer for both the Institute of Electrical and Electronic Engineers (IEEE) Computer Society and the American Society for Engineering Education (ASEE). He has published numerous conference papers and journal articles on innovations in Software Engineering curriculum development and Philosophy of Engineering & Computing.

A Study on the Impact of using Industry Standard Tools and Practices on the Software Engineering Courses Projects

Tajmilur Rahman, Stephen Frezza
rahman007@gannon.edu, frezza001@gannon.edu
Computer & Information Science
Gannon University

Abstract

Traditionally, software engineering courses, especially project based lab courses, are focused on providing students a practical understanding of the subject material at an implementation level. However, if the idea of a lab course is to build student competence for real world software development and engineering problems, why should we not provide them an environment where they can be familiar with the industrial software development, engineering and project management practices? This paper explores the project based instructional benefits in an industry setup using Industry Standard Tools and Practices (IST&Ps) and investigates the learning effectiveness and engagement. IST&P involves software development, deployment and management tools, and common Agile practices using popular web-based tools widely used in industries. Our approach is to engage students in cross-course collaboration [1] with Agile practices with three groups of 3rd and 4th year undergraduate students among three project-based courses: Software Engineering (SE), Requirements and Project Management (RPM) and Mobile App Development (MAD). The goal of the project setup is to engage students in the course content, and course projects in the context of professional software development practices [2]. Our goal also includes observing how effective the study setup is in learning software engineering methods, practices, tools and techniques. An empirical study will be performed to quantify the instructional goals according to the bloom's taxonomy on the participants.

Introduction

The gap between software engineering students and professionals has significant impact on students [3] [4] [5]. When shifting to professional practice, computer science (CS) graduates often face difficulties adjusting to new environment unless they have a substantial amount of internship experience previously. In the undergraduate or graduate schools, project based courses are intended to teach the students how to develop software projects. Although, there are other Software Engineering (SE) courses that help them develop their understanding of how software development and engineering processes work, applying the techniques in a professional setup and the familiarity with the required tools in the current market is always ignored.

Commonly in most of the schools during the final year of computer science undergraduate program students are given two options of doing a thesis or a final year project such as capstone project [6]. The idea behind doing a thesis is to teach students how to do research and how to write a scientific paper. However, the idea behind the final project is always to give the students an experience of developing a full software application by collaborating with stakeholders and following appropriate methods for requirements analysis, design, implementation, quality assurance and project planning. In the course projects or project based courses where students are given a project and asked to implement a complete software system by analyzing the requirements and designing the system using UML and object oriented methods. Students get the opportunity to learn how to write documentations, test cases, and to validate their work with stakeholders. Such senior level course projects provide students with the experience of team-work, testing, quality assurance, development life-cycle, documentation, writing technical reports and possibly other formalities from business and SE perspective. Many institutions even leverage common industry-standard tools [7] [8] and practices, for the purpose of connecting students to Industry Standard Tools and Practices (IST&P). Grabowski et. al. [7] conducted a similar study to emulate the corporate professional standard software development among six software engineering courses' student groups. Unlike Grabowski, we are more interested in assessing the learning effectiveness and engagement in development practices and using IST&P instead of business experience.

These attempts at IST&P integrations are laudable, however there remains the question as to how these efforts at integration support student learning. Efforts in competency assessment [8] have been conducted; yet how the application of IST&P remains largely unassessed. Our intention is to see if the addition of IST&P to project-based learning courses would influence the effectiveness and the engagements of students in learning the core software engineering tools, techniques, agile process, practices and terms. A brief overview of the tools and practices employed follows.

Tools

We use the tools commonly used in most of the software development industries and widely popular among the practitioners. For project management and issue tracking we use **Atlassian Jira**, **Bitbucket** for source code maintenance and version control, **Jenkins** for building the project code and integrate unit tests in a continuous integration mechanism.

Atlassian Jira: A wide variety of project management tools have evolved to this modern day [9] facilitating the control over Agile development process. Atlassian Jira [10] is one of the leading software project management tools for decades. Jira is able to manage small as well as extremely large software projects [11] with its useful features to drive an Agile scrum-based software development process.

Planning Poker: To estimate the user stories in Jira many software development teams prefer to use web based tools that help estimation by playing games such as planning-poker [12], pointing poker [13] etc. We are using planning-poker because of its simplicity and popularity which allows estimation using Fibonacci series, custom-fibonacci series, T-Shirt size, etc.

Jenkins: While development is in progress, a working copy of the software has to be ready at all times as part of the Agile methodologies [14]. For that we used Jenkins to continuously build the

project. Jenkins is vastly popular [15] and stable over the past few decades.

Practices

Agile development practice is gaining popularity rigorously over more than a decade [16]. It is the most popular approach of developing client focused software applications. Hence, we are following the Agile development method and its best practices to run the projects.

Bi-Weekly Sprint: Bi-weekly sprint is a common practice in Agile development practice which implies the development iterations [17]

Daily Scrum: It is a daily session with all the developers and team lead that reviews what has been accomplished, as of now and what are the goals next, and any impediments. It is one of the keystone practices in the Agile development process [18][19]. It drives the sprint, keeps the developers engaged and up to date with the sprint progress.

Sprint Retrospective: In a sprint based development where scrum focuses on the progress, retrospective session is a vital session for the sprint where the criticism of the past sprint addresses the points of improvement which is one of the Agile principles [20].

Methods

The three target courses require project-based collaboration, where one set of students primarily focus on the requirements and project management aspects of the IST&Ps, where the other sets of students are more focused on applying the selected IST&Ps to their code development.

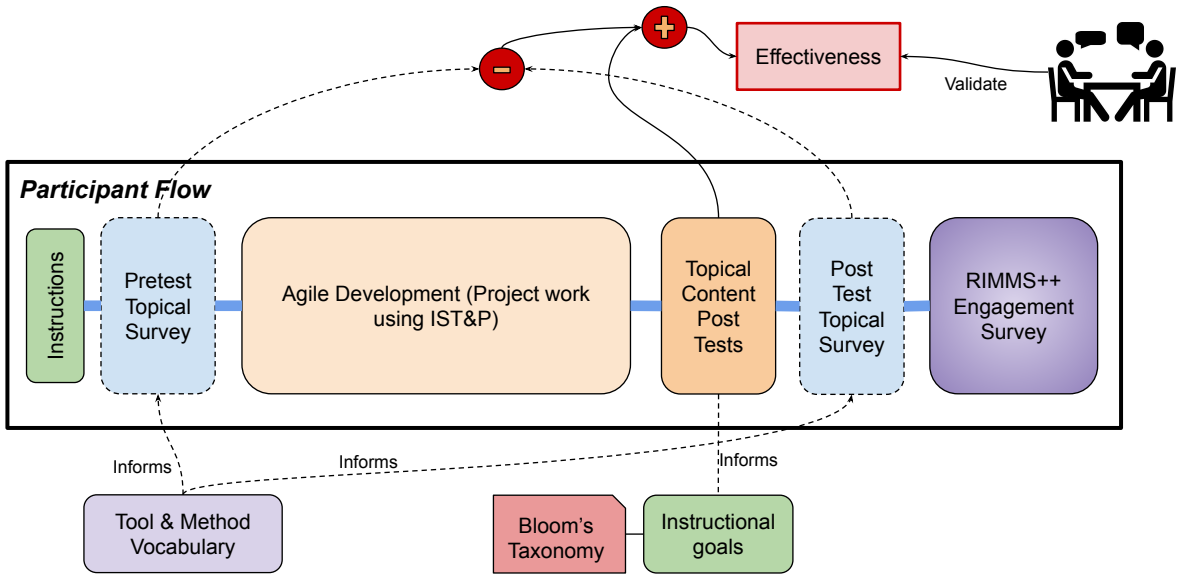

Figure 1: Methodology mapped to participant flow

Our approach involves an empirical investigation on students' experiences and outcomes. As depicted in Figure 1, learning assessment will begin with an IST&P topical survey focusing on

students' perception of their familiarity and ability to use the selected IST&Ps. Another data source will be a comparison of student performance on shared IST&P-related (Blooms level 2 [21]) assessment questions. At the end of the projects we will conduct a variant of the RIMMS [22] survey (RIMMS++) to analyze student engagement and a series of interviews to add qualitative connections to the quantitative findings. Figure 1 represents our methodology at a glance.

RIMMS++:

This work focuses on the assessment of effectiveness and engagement of IST&P-enhanced project-based learning. As indicated in Figure 1, effectiveness will be observed through the change in IST&P topical survey and in performance on shared IST&P-related assessment questions. Engagement will be observed through the RIMMS++ instrument and the interview findings. At the center of this approach is the application of the RIMMS approach in assessing student engagement. While the RIMMS work developed a series of validated questions useful for assessing Attention, Relevance, Confidence and Satisfaction (ARCS) [23]. The RIMMS instrument was built around 12 questions, with three questions in each of the ARCS categories.

For our use, these questions required modification. RIMMS centred on ARCS assessment of student engagement specific to instructional message design [22], our interest is in assessing engagement with a set of IST&Ps. Our approach was to parametrize the RIMMS questions, for the purpose of shifting the target from "user instructions" to student engagement with a particular IST&P, while at the same time shifting the context from "using a telephone" to the team project. The R07 record in Table 1 depicts one such transformation, illustrating the original RIMMS question and the resulting RIMMS++ question. In Table 1 "R" with the question numbers indicates that it's a RIMMS question, and "rewrite" at the end indicates we propose to re-write that RIMMS question.

In this conversion process, we observed that the original RIMMS included two very similar "satisfaction" questions: R11 and R12 (See Table 1). For our experiment, we included these in the RIMMS++ instrument, but also added two differentiating satisfaction questions (e.g., R11 rewrite) following the ARCS model. In all, this approach generated a set of 14 questions that can be easily answered for each of the six IST&Ps integrated into the course.

Analysis

This study expects to generate quantitative and qualitative data that can be assessed for validity. To quantitatively test the learning effectiveness, we will perform a two-way ANOVA between the pre-test and the post-test survey results groups using the following null hypotheses:
H1: Means of observations grouped by pre-test are the same
H2: Means of observations grouped by post-test are the same
H3: There is no interaction between pre-test and post-test
The results of the Bloom's Taxonomy [24] based outcome metric from the "Topical Content Post-Tests" will then be used to test the learning effectiveness by performing another two-way ANOVA between the post-test survey results and the topical post-test results. The learning engagement will be measured based

Table 1: RIMMS 2015 Questions Plus Two Rewrites (RIMMS++)

RIMMS Q	**ARCS Principles**	**Original RIMMS Question [22]**	**RIMMS++ Question**
R7	Confidence - Learning requirements	As I worked with these ⟨user instructions⟩, I was confident that I could learn how to work well with the ⟨telephone⟩	As I worked with this **industry standard tool or practice**, I was confident that I could learn how to succeed in the **course project**
R11	Satisfaction - Intrinsic Reinforcement	I really enjoyed working with these ⟨user instructions⟩	I really enjoyed working with this **industry standard tool or practice**
R12	Satisfaction	It was a pleasure to work with such well-designed ⟨user instructions⟩	It was a pleasure to experience working with this well-designed **industry standard tool or practice**
R11 rewrite	Satisfaction - Extrinsic	⟨Completing the exercises⟩ gave me a satisfying feeling of accomplishment	**Working with this industry standard tool or practice** gave me a satisfying feeling of accomplishment
R12 rewrite	Satisfaction - Equity	The design of the ⟨user instructions⟩ made for a satisfying learning experience. (36S06)	The integration of **working with this industry standard tool or practice** into my course project made for a satisfying learning experience

on the RIMMS++ total score by performing a one-way ANOVA.

Qualitative Survey / Interviews

Once the project development is completed we will conduct a series of interviews with individual students with a set of questionnaires to understand their interest, confidence, and satisfaction within the process using IST&P. The interview will be conducted separately by both the instructors engaged in this research, and the responses will be collected separately. Each student will be interviewed for at least 15 - 20 minutes depending on our final set of questionnaires. The questionnaires will be very similar to the post test topical survey. However, the answers from the face-to-face interview will provide more insights that the predefined answers set in the survey questionnaires might not capture. This will help us validate the empirical analysis from the topical content test, pre and post-test survey, and RIMMS++ survey data.

Conclusion

This work is still in-progress. Our study has been set up with three undergraduate courses involving two faculties. In this emulation students get the opportunity to perform through all standard software development phases in Agile method including requirements analysis, user-story backlog creation. Once the backlog is ready, developers plan for sprints and drive each sprint equipped with daily-scrum, retrospective and planning the next sprint. The entire process is driving through use of IST&P. Once this study is done, our collected data will give us an insight about how this protocol impacts the learning effectiveness and how it engages the students. We believe that the empirical data will give us a positive result reflecting the engagement and learning effectiveness, and we will be able to validate it through the qualitative interviews. This study will also evaluate the two questions, R11-rewrite and R12-rewrite that we brought back in from the original IMMS table and added into RIMMS and will update RIMMS as

RIMMS++ if they show any significance.

References

[1] S. Frezza, M. Daniels, A. Pears, Å. Cajander, V. Kann, A. Kapoor, R. McDermott, A.-K. Peters, M. Sabin, and C. Wallace, "Modelling competencies for computing education beyond 2020: a research based approach to defining competencies in the computing disciplines," in *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, 2018, pp. 148–174.

[2] B. Rolandsson, M. Bergquist, and J. Ljungberg, "Open source in the firm: Opening up professional practices of software development," *Research Policy*, vol. 40, no. 4, pp. 576–587, 2011.

[3] J. Liebenberg, M. Huisman, and E. Mentz, "The relevance of software development education for students," *IEEE Transactions on Education*, vol. 58, no. 4, pp. 242–248, 2015.

[4] T. Clear, "Thinking issues: Meeting employers expectations of devops roles: can dispositions be taught?" *ACM Inroads*, vol. 8, no. 2, p. 19–21, 2017.

[5] J. C. Nwokeji, R. Stachel, T. Holmes, and R. O. Orji, "Competencies required for developing computer and information systems curriculum," in *2019 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2019, pp. 1–9.

[6] D. R. Schachter and D. Schwartz, "The value of capstone projects to participating client agencies," *Journal of Public Affairs Education*, vol. 15, no. 4, pp. 445–462, 2009.

[7] L. M. Grabowski, C. F. Reilly, and W. A. Lawrence-Fowler, "Emulating a corporate software development environment through collaboration between student projects in six courses," in *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*. IEEE, 2014, pp. 1–7.

[8] M. Kropp, A. Meier, and R. Biddle, "Teaching agile collaboration skills in the classroom," in *2016 IEEE 29th International Conference on Software Engineering Education and Training (CSEET)*. IEEE, 2016, pp. 118–127.

[9] A. Mishra and D. Mishra, "Software project management tools: a brief comparative view," *ACM SIGSOFT Software Engineering Notes*, vol. 38, no. 3, pp. 1–4, 2013.

[10] Atlassian, *Atlassian - Jira Software*, 2021 (accessed March 5, 2021), https://www.atlassian.com/software/jira.

[11] J. Fisher, D. Koning, A. Ludwigsen *et al.*, "Utilizing atlassian jira for large-scale software development management," in *14th International Conference on Accelerator & Large Experimental Physics Control Systems (ICALEPCS)*, 2013.

[12] N. C. Haugen, "An empirical study of using planning poker for user story estimation," in *AGILE 2006 (AGILE'06)*. IEEE, 2006, pp. 9–pp.

[13] P. Poker, *Pointing Poker*, 2021 (accessed March 5, 2021), https://www.pointingpoker.com.

[14] G. Kumar and P. K. Bhatia, "Impact of agile methodology on software development process," *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, vol. 2, no. 4, pp. 46–50, 2012.

[15] M. Lenz, "Continuous integration with jenkins," in *Python Continuous Integration and Delivery*. Springer, 2019, pp. 39–52.

[16] M. Al-Zewairi, M. Biltawi, W. Etaiwi, A. Shaout *et al.*, "Agile software development methodologies: survey of surveys," *Journal of Computer and Communications*, vol. 5, no. 05, p. 74, 2017.

[17] T. Haugen, S. Seiler, Ø. Sandbakk, and E. Tønnessen, "The training and development of elite sprint performance: an integration of scientific and best practice literature," *Sports medicine-open*, vol. 5, no. 1, pp. 1–16, 2019.

[18] K. Schwaber and M. Beedle, *Agile software development with Scrum*. Prentice Hall Upper Saddle River, 2002, vol. 1.

[19] M. Paasivaara, S. Durasiewicz, and C. Lassenius, "Distributed agile development: Using scrum in a large project," in *2008 IEEE International Conference on Global Software Engineering*. IEEE, 2008, pp. 87–95.

[20] R. C. Martin, *Agile software development: principles, patterns, and practices*. Prentice Hall, 2002.

[21] R. Britto and M. Usman, "Bloom's taxonomy in software engineering education: A systematic mapping study," in *2015 IEEE Frontiers in Education Conference (FIE)*, 2015, pp. 1–8.

[22] J. M. Keller, "First principles of motivation to learn and e3-learning," *Distance Education*, vol. 29, no. 2, pp. 175–185, 2008.

[23] J. Keller and E. Burkman, "Motivation principles," *Instructional message design: Principles from the behavioral and cognitive sciences*, vol. 2, pp. 3–49, 1993.

[24] D. R. Krathwohl, "A revision of bloom's taxonomy: An overview," *Theory into practice*, vol. 41, no. 4, pp. 212–218, 2002.