Combining Challenge-Based Learning and Scrum Framework for Mobile Application Development

Alan R. Santos
PUCRS
Porto Alegre, RS, Brazil
alan.santos@pucrs.br

Afonso Sales
PUCRS
Porto Alegre, RS, Brazil
afonso.sales@pucrs.br

Paulo Fernandes
PUCRS
Porto Alegre, RS, Brazil
paulo.fernandes@pucrs.br

Mark Nichols
The Challenge Institute
Chandler, AZ, USA
marknichols@mac.com

ABSTRACT

The market for mobile applications has been growing dramatically, as has the complexity of the applications and the speed of the development process. These changes require a rethinking of the development process and of how developers are trained. In order to better prepare faculty and students for the emerging mobile application market, this study presents a new learning and software development framework that combines Agile methodologies with the Challenge-Based Learning (CBL) framework. CBL provides a student-centered learning framework that mirrors the modern workplace. Agile methodologies address the changing landscape of mobile development environments. A combination of the CBL learning framework and Agile methodologies can better prepare students for the development market. This paper presents an empirical study applying CBL and Scrum in a mobile application development course evaluated through a series of post surveys. The results indicate that a teaching and learning environment based on practical experience combining the CBL framework with the Scrum process is an effective model to promptly teach undergraduates how to be successful mobile application developers.

Categories and Subject Descriptors

K.3.2 [**Computing Milieux**]: Computers and Education—*Computer and Information Science Education*

Keywords

CBL, Scrum, Mobile Application Development

1. INTRODUCTION

In recent years, the growth in both the development and use of mobile applications has presented new challenges to the software engineering field. The adoption of features such as cameras, sensors, touch and GPS in mobile platforms has rapidly expanded the possibilities for mobile applications. These applications have become more complex and mission critical [10] due to the sudden wave of mobile device use. Simultaneously, speed of deployment has become a key factor due to developers' possibility of submitting apps directly to the market. These changes make it is necessary to identify new ways to prepare developers, and new ways to develop as well. The popularity of mobile application development and the easy access to the market has resulted in more people dedicating themselves to the field of computing who have not been trained as either computer scientists or software engineers. This points to another reason to explore new ways to train future mobile application developers.

Challenge-Based Learning (CBL) [5] is an active, student-directed instructional strategy in which skills are gained through working on real-world problems. At the center of CBL is a call to action that requires students to develop solutions and implement them in authentic environments [7].

The contribution of this paper is an empirical study that combines CBL and Scrum, enabling structured reasoning and decision-making on teaching mobile application development. Although both methodologies/frameworks are commonly used, to our knowledge there is no existing resource that illustrates how the two methods may be combined in a mobile application course.

2. BACKGROUND

Mobile Application Development is a process in which applications are developed for small handheld devices, being either pre-installed on devices during manufacture or downloaded from application stores or other software distribution platforms [3]. In a report from the 2013 ICSE 1st International Workshop on Engineering Mobile, Lewis *et al.* [10] argued that mobile applications are becoming an increasingly important part of enterprise and mission critical systems. According to Wasserman [16],*"using a mobile device is different from working with a desktop or laptop computer. While gestures, sensors, and location data may be used in game consoles and traditional computers, they play a dominant role in many mobile applications"*. The challenge is to understand how to best prepare students to operate in this emerging market [11]. The combination of the Scrum and

ITiCSE'15, July 6–8, 2015, Vilnius, Lithuania.
Copyright © 2015 ACM 978-1-4503-3440-2/15/07 ...$15.00.
http://dx.doi.org/10.1145/2729094.2742602.

CBL frameworks may be an option for a mobile application development environment.

2.1 Scrum Framework

Scrum is an iterative and incremental agile software development approach, presented by Ken Schwaber on a paper describing the Scrum Methodology in 1995 [15]. Hasnain [4] performed a literature review of agile methods and demonstrated that the number of Scrum studies has been increasing each year. Tore and Torgeir [2] performed a literature review of empirical studies of agile software development and found that agile methods like Scrum deserve further attention.

The Scrum workflow is a sequence of iterations, named *Sprints*, which have a duration between one and four weeks. The team is guided by the work foundation as part of a product backlog which is a list of requirements and priorities. Each Sprint has daily meetings where each team member presents what has been done on the previous day, what is going to be done until the next day, and whether there is any roadblock to move forward on development activities. At the end of each Sprint there is a product demo or *Sprint Review*, and following each Sprint Review there is a lessons learned session or *Sprint Retrospective* [14].

Harleen and Swati [3] performed a review and analysis of agile methodologies in mobile development contexts and found that the agile approach is effective in mobile application development [4]. Kajeel and Harishankar [8] studied agile and Scrum practices in the context of android software development, confirming that Agile and Scrum processes are effective in project development scenarios with requirements changing frequently and fast.

In the context of studies conducted to evaluate the effectiveness of agile practices in classroom settings, Scharff and Verma worked with Scrum in a classroom setting at Pace University [14]. They report that mobile applications can be developed in a short time, that Scrum was the main reason for the success of the project when time was a constraint, and that throughout the project, mobile application development content was learned just-in-time.

2.2 Challenge-Based Learning

Experiential learning is based on the idea that students learn best when actively involved in open-ended activities rather than as passive participants in staged activities. The foundations of experiential learning can be found within the history of most cultures, but were formally organized and presented by David Kolb [9], drawing heavily on the works of John Dewey [1] and Jean Piaget [13]. The overarching ideas of experiential learning have spawned a wide variety of learning frameworks, including Problem-Based Learning, Project-Based Learning, and Expeditionary Learning, among others. We reviewed these existing learning frameworks and they place the student at the center and focus on students working together to discover or uncover knowledge, rather than acquiring knowledge through traditional direct instruction from a professor. These learning approaches have recently been advocated as a better way to prepare students for a fluid and dynamic modern work environment.

Challenge-Based Learning (CBL)[1] is a learning framework pioneered by educators working with Apple Inc. which has been implemented in a wide variety of educational and corporate settings [5]. CBL has roots within experiential learning, in which students actively acquire knowledge through work on open-ended problems.

CBL has the following characteristics: i) the professor, students and stakeholders work as active collaborators in the learning process; ii) the inclusion of both technical and workplace skills [6]; iii) a focus not only on the final product, but on the process developed through ongoing reflection and publishing of perspectives about what was learned; and iv) more time allocated to incorporating divergent and creative thinking.

The CBL white paper [5] defines the Challenge-Based Learning as a process that begins with a big idea, moves to an actionable challenge, and eventually to the implementation of a carefully considered solution. Details of each phase are presented in Table 1.

Table 1: Challenge-Based Learning stages

Name	Description
Big Idea	A broad concept that can be explored in multiple ways, is engaging, and has importance to students and the broader society.
Essential Question	A process of personalizing and pinpointing the important concepts within the big idea.
Challenge	A call to action designed by professors and students to create a solution that can result in concrete action.
Guiding Questions	A series of questions developed by the learning community, identifying and representing the knowledge and skills needed in order to develop a successful solution.
Guiding Activities and Resources	The activities and resources that learners identify, participate in and utilize to answer the guiding questions.
Analysis	A process for exploring the answers to the guiding questions and identifying overarching themes and concepts. This sets the foundation for solutions.
Solution	A concrete, actionable and clearly articulated idea to solve the challenge. Complicated challenges will often have multiple solutions.
Implementation	This is when the solutions are put into action with an authentic audience.
Evaluation	Learners evaluate their process through the results of the implementation and refine their solution.

Studies of experiential approaches to learning have demonstrated that students acquire more solid workplace skills than when taught with traditional methods such as lecture [6]. Timothy *et al.* [12] presented a comparison of Lecture-Based Learning and Challenge-Based Learning in a workplace setting. The study found that participants in the challenge-based group scored significantly better in post-test items requiring integration [12].

[1] Details and samples about CBL are also available at http://www.challengebasedlearning.org

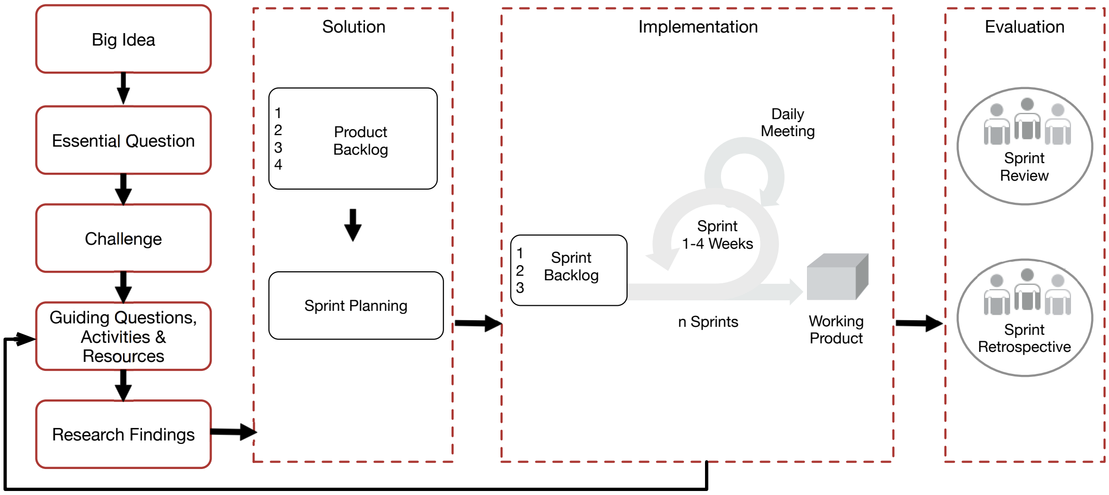

Figure 1: Combining CBL and Scrum

3. THE COURSE

The course focus of this research was a six month iOS development course with 94 undergraduate students. The format of the course involved learning mobile applications programming concepts through completing challenge-based learning assignments. The course was taught in a unique learning space configured to provide a variety of working environments. Each student had his/her own equipment to use as part of class meetings and projects. The course curriculum included the following: Object-Oriented Programming, UI components, Model View Controller, Datasources, Navigation, Animations and Frameworks.

Instead of following a linear content coverage model using direct instruction, the CBL framework and process guided the content, timing and delivery of the curriculum. The faculty and students worked together to identify and acquire the knowledge and skills necessary to solve a variety of real world development challenges. Through this process they covered the specific learning objectives in the course syllabus.

Most of the participants in the course are from an IT related field: 34% from Computer Science, 41% from Information Systems, 8% from Computer Engineering and the remaining students were from other types of undergraduate courses. In this context, 21% of the students were in the 3rd semester, 31% were in the 4th semester, 11% in the 5th semester, 15% in the 6th semester, and the others in different semesters. Another profile information from students is that 35% of the students had already had previous software development courses using Java and C#. In this context, 68% had up to 3 years of experience in development, 18% had between 3 and 5 years of experience, and 14% had more than 5 years software development experience. Most of this previous experience is from other courses, as well as from the industry.

The course was facilitated by 6 instructors. They had knowledge of iOS development, academic and project management background, and four of these instructors had more than five years of experience as software developers.

4. COMBINING CBL AND SCRUM

During course planning, CBL and Scrum were considered as elements separate from the class. CBL was the teaching methodology, while Scrum was to be a development methodology taught to the students. It immediately became apparent that the two approaches overlapped and complemented each other.

An important step in combining CBL and Scrum was setting a timeline with due dates for the Big Idea, Essential Question, Challenge, Guiding Questions, Resources and Activities (GRA), prototype, alpha/beta versions and the final application. In the course, each Challenge began with the students identifying a big idea that was of interest to them. Working together with the faculty, they contextualized this idea through a questioning process that led to a challenge involving the development of a mobile application. The students then used the Guiding Questions, Resources, and Activities to organize and to document their work. This is a critical step in the learning and development process, as it provides time to deeply and widely think about the challenge before developing a solution and starting to code. This step is critical and must be performed before starting the Scrum integration stage, as it ensures that a solution has been thoroughly conceptualized. The combination of CBL and Scrum is presented in Figure 1.

Students may not move to the solution stage until they have answered their core guiding questions, analyzed the results and can support their solution. The professors must also work with students in order to recognize that the iterative process of CBL leads to ongoing guiding questions and the continuing need for activities to answer them. The transition from the challenge to the solution is a critical element in the CBL process, and is often difficult for students who may want to move directly to solutions [5].

After finishing GRA, students present their research findings and analysis. The research findings from GRA are used to build the product backlog sorted by priorities.

The CBL solution stage is when the Scrum product backlog and sprint planning definitions occur, and it repeats as needed through sprints in order to address the scope of work. The mobile development and testing activities are executed as part of each Sprint during the implementation stage. The evaluation stage is when the Sprint Review assesses the incremental product developed, and also when the Sprint Retrospective is performed to verify what the team should start/continue doing and what the team should stop doing in order to promote continuous improvement and learning. At the end of each sprint, the stage of evaluation repeats as long as there are product reviews and lessons learned sessions to be conducted.

In this context, the integration between CBL and Scrum starts at the Solution gate and ends at the Evaluation gate. In order to allow for continued improvement, the stages of Solution, Implementation and Evaluation repeat according to the number of sprints required to address the scope of work. Throughout this process, guiding questions emerge and are answered, deepening the student's understanding of the solution. We have performed an empirical study in order to assess the combination proposed on this research, as follows.

5. EMPIRICAL STUDY

In order to validate the proposed approach, we performed an empirical study between November 2013 and May 2014, organized in three challenges.

5.1 Implementation of the first challenge

Students had two months of mobile software development technical content, CBL methodology introduction and Scrum framework introduction. They were distributed in 25 Scrum teams of two to five (part-time) students each. The first challenge duration was three weeks. One week was dedicated to work on Big Idea, Essential Question, Guiding questions resources and activities. Time was also allocated to build the product backlog. The next two weeks served as sprints to work on the challenge solution, implementation and evaluation. Daily scrum meetings were held every day in order to track what had been done on the previous day, what was going to be done that day and whether there were any impediments or roadblocks. At the end of the solution gate and the implementation gate, sprint reviews and sprint retrospectives were promoted as part of the CBL evaluation process. As a result, 24 teams were able to finish their own mobile applications, while one group was unable to finish their first mobile application version. The scope of the apps delivered as part of this first challenge covers the following areas: social charity, urban mobility, law, productivity, HR, gastronomy, health care, politics, finance and tourism.

Based on the daily meeting status tracking, it was possible to detect the reasons why one group did not achieve the end result: i) the group did not organize a product backlog; ii) the group did not perform a sprint planning; iii) lack of communication among team members and lack of engagement from part of the team.

5.2 Implementation of the second challenge

After the implementation of the first challenge, students had one more month of mobile software development technical content and a review on CBL methodology and Scrum framework. They were distributed in 25 Scrum teams of two to five (part-time) students each. The second challenge duration and Scrum practices were the same as the first challenge. As a result, 24 teams were able to finish a mobile application version and one group was unable to finish their mobile application project. The scope of the apps delivered as part of this second challenge covers the entertainment and games area. Based also on the daily meeting status tracking, it was possible to detect the reasons why one group did not achieve the end result: i) lack of configuration management which caused several issues on code merge; ii) design solution issues; iii) lack of commitment from part of the team.

5.3 Implementation of the third challenge

After the implementation of the second challenge, students had one more month of mobile software development technical content. They were distributed in 29 Scrum teams of two to four (part-time) students each. The third challenge duration and Scrum practices were also the same as those of the first and second challenges. As a result, all teams were able to finish a mobile application version, with seven teams doing incremental work from previous challenges and 22 teams building brand new applications. The scope of the apps delivered as part of this third challenge covers the following areas: nutrition, sales, sustainability, accessibility, entertainment, psychology, education, beauty services, social networks, urban mobility, security, productivity, services, health care and pets.

5.4 Surveys

Surveys were conducted after the end of each challenge project in order to measure variables related to Mobile Application Development, Challenge-Based Learning and Scrum.

Survey Protocol:

The goal of the surveys conducted in this study was to identify initial student perceptions of the combination method and its impact on mobile application development. The sample population was composed of undergraduate students in a mobile application development program, who were chosen using the convenience criteria due to the fact that participants were selected for their availability. The sample population size was defined using the higher number of available people to participate (94 students).

The questionnaires used the following structure: demographic questions; questions about CBL in order to evaluate aspects such as learning improvement, work control, performance, flexibility and ease to use; questions about Scrum to evaluate aspects such as productivity, effectiveness and utility.

5.5 Results

5.5.1 First challenge results

After ending the first challenge project, the first survey was applied. The survey received 78 responses out of 94 participants. In this sense, 86% of students stated that CBL is very helpful to build better requirements.

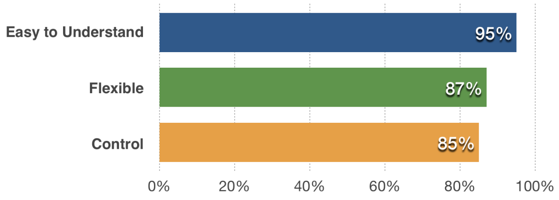

Figure 2: CBL First Challenge results

Moreover, 95% of the students stated that CBL is easy to understand; 87% confirm that CBL is flexible due to the fact that various contents can be practiced when using it; and 85% confirm that by using CBL they have more control over educational activities related to the course.

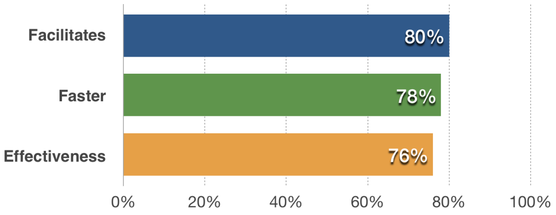

Figure 3: Scrum First Challenge results

In the context of the use of Scrum in the first challenge, 76% confirm that by using Scrum they have improved effectiveness in conducting challenge activities; 78% confirm that Scrum led to faster performance of activities related to the challenges; and 80% confirm that Scrum facilitates mobile applications development organization.

5.5.2 Second challenge results

After ending the second challenge implementation, another survey was applied to collect students input about the use of CBL integrated to the Scrum framework. The survey received 83 responses out of 94 participants.

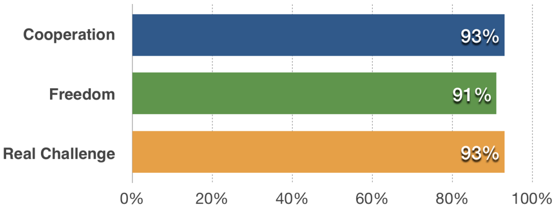

Figure 4: CBL Second Challenge results

In regards to the use of CBL in the second challenge, 93% confirm that with CBL they have real world challenges that demand instructors and students to work on complex solutions; 91% confirm that CBL provides an environment where students have freedom to learn and to teach; and 93% confirm that by using CBL they work in cooperation with instructors and teammates.

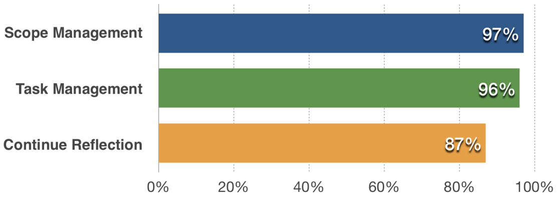

Figure 5: Scrum Second Challenge results

In the context of the use of Scrum in the second challenge, 87% confirm that using daily meetings makes them continually reflect on the CBL solution implementation; 96% confirm that by using Scrum they organize the tasks to be developed during the sprint, which facilitates the development process organization; and 97% confirm that by using Scrum they apply the product backlog artifact, facilitating the CBL solution scope management. In this second challenge, when asked if they would like to share any other opinion about the learning and development process using CBL and Scrum on challenges, it was possible to confirm the effectiveness of CBL and Scrum based on the answers: *"The process is becoming more structured, speeding development"*; *"The purpose of the CBL has become clearer in the second challenge due to the experience gained in the first challenge"*; *"The use of both methods helps to have a greater perception of the size of the project scope, resources and tools. This allows for a better management of the project, since it helps to ensure that tasks are completed"*.

5.5.3 Third challenge results

After ending the third challenge, a final survey was applied in order to collect students' impressions on the use of CBL integrated to the Scrum framework. The survey received 80 responses out of 94 participants.

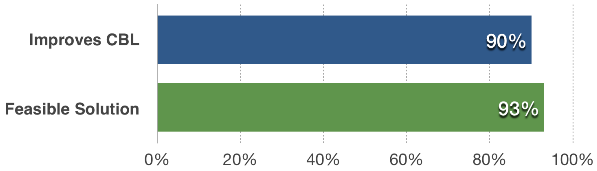

Figure 6: CBL and Scrum Third Challenge results

In regards to CBL and Scrum, 90% confirm that the use of Scrum improves the CBL framework for mobile application development, and 93% confirm that CBL is a feasible solution for a mobile learning development environment.

Based on the open question about how using Scrum improves CBL to develop mobile applications, some interesting results were found: *"Scrum is to assist the development of the app, especially team communication and project documentation. Scrum keeps the team at an appropriate pace with no impact on application development"*; *"By organizing backlog, prioritization and ownership of the activities you can keep development scope under control"*; *"With Scrum, we always know where we are and where we should go in developing the app, avoiding waste of time"*.

When asked about the factors that make the CBL a feasible solution for a mobile development learning environment, answers included: *"The CBL reminds students to look for more knowledge and not to be limited, thus making it a more effective learning"*; *"Possibility of changes over work, division of tasks and clarification of what is required or not for the project"*. When asked about their knowledge on mobile application development before the course using a scale from 0 to 10, the average value was 2.75. When asked about their knowledge on mobile application development after the course using the same scale, the average value was 8.03. This indicates that our course actually represented a change on the knowledge of the majority of the students.

6. DISCUSSION

As far as we know, this is the first documented empirical study examining the use of Challenge-Based Learning and

Scrum with undergraduate students learning mobile application development. This work contributes to two important discussions: how to educate developers of mobile applications and effective models for the development of mobile applications.

None of the participants had contact with Challenge-Based Learning before the course, and for 68% of the participants it was their first contact with Scrum. For 85% of participants it was their first contact with mobile application development. For the majority of the students, it was their first contact with CBL methodology, Scrum framework and mobile application development.

According to the results of students' perceptions, we found that a teaching and learning environment based on practical experience, combining the Challenge-Based Learning framework with the Scrum process, was an effective model for undergraduate students to learn in a short time to be effective mobile application developers. We were able to obtain important information on mobile application development environments, demonstrating that a majority of participants agree and completely agree that Scrum helps to efficiently perform activities related to challenges, and that CBL enhances the use of Scrum concerning the quality of the solutions due to the big idea, essential question, challenge and research stages. We were able to confirm that CBL is effective in technology-rich learning environments [6], and we confirmed that ongoing student and professor reflection and publishing improves the learning process.

7. FINAL REMARKS

The main objective of this study was to empirically explore processes and practices aimed at integrating the Challenge-Based Learning framework and Scrum methodologies. Based on the findings of this study, we reach the following conclusions: Challenge-Based Learning is a successful framework for mobile software development. The results also demonstrate that Scrum can be successfully integrated with CBL for mobile application development. The results also revealed that the mobile application development projects require a documented process where agile processes such as Scrum can help.

Our research also contributes to the dialogue on new methodologies for application development learning. Combining CBL with Scrum not only improved the learning process, but also resulted in a new approach that was effective in the rapid development of high quality mobile applications.

The research strategy applied to conduct this study presents potential limitations due to the research method selected. In future studies examining CBL and agile integration, more empirical studies and surveys will be applied in order to collect other aspects related to the proposed integration approach.

8. ACKNOWLEDGMENTS

Afonso Sales is funded by CNPq-Brazil (Universal 470096/2013-6) and Paulo Fernandes is also funded by CNPq-Brazil (PQ 307602/2013-3).

9. REFERENCES

[1] J. Dewey. *Experience and Education*. Kappa Delta Pi., New York, NY, 1938.

[2] T. Dybå and T. Dingsøyr. Empirical studies of agile software development: A systematic review. *Information and Software Technology*, 50(9-10):833–859, 2008.

[3] D. S. V. C. Harleen K. Flora. A review and analysis on mobile application development processes using agile methodologies. *International Journal of Research in Computer Science*, 3(4):9–18, 2013.

[4] E. Hasnain. An overview of published agile studies: A systematic literature review. In *Proceedings of the 2010 National Software Engineering Conference*, NSEC '10, pages 3:1–3:6, Rawalpindi, Pakistan, 2010.

[5] A. Inc. Challenge based learning: A classroom guide. Technical report, Apple Inc., Cupertino, CA, 2012.

[6] L. Johnson and S. Adams. Challenge based learning: The report from the implementation project. Technical report, The New Media Consortium, Austin, TX, 2011.

[7] L. F. Johnson, R. S. Smith, J. Smythe, and R. K. Varon. Challenge-based learning: An approach for our time. Technical report, The New Media Consortium, Austin, TX, 2009.

[8] S. B. Kaleel and S. Harishankar. Applying agile methodology in mobile software engineering: Android application development and its challenges. Technical report, Department of Computer Science, Ryerson University, 2013.

[9] D. Kolb and R. Fry. *Toward an Applied Theory of Experiential Learning*. C. Cooper (ed.). Theories of Group Process, London: John Wiley., London, 1975.

[10] G. A. Lewis, N. Nagappan, J. Gray, D. Rosenblum, H. Muccini, and E. Shihab. Report of the 2013 icse 1st international workshop on engineering mobile-enabled systems (mobs 2013): 12. *SIGSOFT Software Engineering Notes*, 38(5):55–58, Aug. 2013.

[11] L. Naismith, P. Lonsdale, G. Vavoula, and M. Sharples. Literature review in mobile technologies and learning. Technical report, University of Birmingham, 2013.

[12] T. K. O'Mahony, N. J. Vye, J. D. Bransford, E. A. Sanders, R. Stevens, R. D. Stephens, M. C. Richey, K. Y. Lin, and M. K. Soleiman. A comparison of lecture-based and challenge-based learning in a workplace setting: Course designs, patterns of interactivity, and learning outcomes. *Journal of the Learning Sciences*, 21(1):182–206, 2012.

[13] Routledge and K. Paul. *The construction of reality in the child*. Basic Books Inc., New York, NY, 1954.

[14] C. Scharff and R. Verma. Scrum to support mobile application development projects in a just-in-time learning context. *Proceedings of the 32nd ICSE*, pages 25–31, 2010.

[15] K. Schwaber. SCRUM Development Process. In J. Sutherland, C. Casanave, J. Miller, P. Patel, and G. Hollowell, editors, *Business Object Design and Implementation (OOPSLA'95 Workshop Proceedings, Austin, Texas)*, pages 117–134. Springer London, 1997.

[16] A. I. Wasserman. Software engineering issues for mobile application development. In *Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research*, FoSER '10, pages 397–400, Santa Fe, New Mexico, USA, 2010. ACM.