Poster: Challenge-Based Learning: A Brazilian Case Study

Alan Santos
Afonso Sales
Paulo Fernandes
School of Technology, PUCRS
Porto Alegre, RS, Brazil
{alan.santos,afonso.sales,paulo.fernandes}@pucrs.br

Josiane Kroll
University of Manitoba
Winnipeg, MB, Canada
josiane.kroll@umanitoba.ca

ABSTRACT

Mobile application development (MAD) has became, or is considering to be a part of the academic curricula in Computer Science courses. However, training students on mobile application development inherits the challenges of teaching software engineering where the target computer is a device that has a large number of features accessible by software. Furthermore, the most related experience in teaching students reveals difficulties in developing software engineering competencies. In this paper we present results from a case study conducted in four universities in Brazil. We have investigated the adoption of Challenge-Based Learning (CBL) framework and agile practices for training students in software engineering applied in mobile application development environments.

CCS CONCEPTS

• **Social and professional topics → Software engineering education**; • **Information systems** → *Mobile information processing systems*;

KEYWORDS

Challenge-Based Learning, Software Development, Mobile, Agile Practices, Software Engineering Education.

ACM Reference Format:
Alan Santos, Afonso Sales, Paulo Fernandes, and Josiane Kroll. 2018. Poster: Challenge-Based Learning: A Brazilian Case Study. In *ICSE '18 Companion: 40th International Conference on Software Engineering Companion, May 27-June 3, 2018, Gothenburg, Sweden.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3183440.3195061

1 INTRODUCTION

In spite of the accumulation of knowledge generated in recent years, the mobile application development (MAD) industry continues growing over the years. In this direction, the traditional development model seems not to be attractive anymore, as long planning and development cycles result in applications out of date [2].

The development software process transformation in the IT industry raises new questions that needs to be answered attend to the market changes. Therefore, these changes generate a need to reevaluate development processes and how to better prepare computer science students to work in this market [5].

In the academia, software engineering (SE) concepts are usually teach through theoretical disciplines, with lectures, supplementary readings, and theoretical exercises that span practical competence in a limited space of time [3]. In this sense, when exposed to the industry, students find a scenario in which learned techniques and methods may be little applied [6]. For Beckman *et al.* [1], one of the ways to increase the quality of teaching is by improving the teaching and learning processes through innovative didactic and strategies.

Active learning techniques such as Problem-Based Learning, Project-Based Learning (PBL) and Challenge-Based Learning (CBL) engage students and improve their performance positively. This type of techniques moves students beyond a watchful role in traditional classes to activities that engage students in real problems [7]. In this study, we focus on CBL approach. CBL aims to enable students to conduct research by integrating practices with theory and application of knowledge and skills, such as collaboration, problem solving and exibility, and taking action in the community context.

Another point that differentiates CBL from other practices is that it is effective in learning environments with technology [4], which allows for agility in the learning process as students skills are obtained through working in real world problems.

In order to contribute to the SE area and help students to improve their SE skills, this work presents results from a case study that combines the CBL and agile practices in mobile application development (MAD) environments. For this, an approach using CBL and agile practices was developed to evaluate how this junction might be adopted in MAD environments.

2 THE CASE STUDY

We conducted a case study in four universities located in the North, Northeast, Southeast, and South of Brazil. These universities provide training courses in MAD where students have the opportunity to develop iOS applications as part of their assignments. The training course is offered to graduate and undergraduate students and the only requirement for students is to have basic programming logic skills. This project is a pioneer initiative in the country and it is held in a unique learning environment, configured to provide a variety of work environments.

Each training course has one year duration, with class sizes ranging from 40 to 100 students organized in different teams. The course is divided into two main phases. The first phase focuses on the theoretical basis through workshops and small application

ICSE '18 Companion, May 27-June 3, 2018, Gothenburg, Sweden

ACM ISBN 978-1-4503-5663-3/18/05.
https://doi.org/10.1145/3183440.3195061

projects development, while the second phase focuses on more robust projects with a bigger application scope. In the second phase students are allocated to teams where they have six months to develop their projects.

The course curriculum includes Object Oriented Programming, User Interface Components, Model View Controller (MVC), data sources, navigation, animations and frameworks topics. Students are also trained in CBL and agile practices. Instructors played as experts for all students groups, overseeing projects and providing feedback. The size of student teams varies between two and five members, since the students are free to choose their own teams. The course requires 20 hours dedication per week (four hours a day). The course is facilitated by instructors, which have different levels of industry and academic experience, also it is important to remark that each university has different teachers running the program. All teachers have previous iOS development training and CBL training and most of them have some experience with agile software development.

Universities were selected following convenience criteria. The data collection consisted of primary sources through interviews. We conducted 32 semi-structured interviews with students from different teams. Each university provided a facilitator to support the interviews process and all interviews were transcribed.

3 RESULTS

Our results show the adoption of CBL combined with agile practices helps in rapidly identifying problems, constant feedback, development speed, project organization and transparency, communication, product delivery, flexibility and performance, iterative planning and continuous improvement, and improve quality.

Agile practices support CBL implementation specially for MAD. These type of environment is dynamic in terms of constant platform and constant requirements updates, which requires a set of flexible software practices. Since CBL is based on active learning approaches, agile practices fit through the dynamics of flexible software development environments.

We also found that different agile practices were adopted by the participants on the project development, such as: automated builds (15.62%), using of burndown chart (21.87%), daily meeting (81.25%), collocation (81.25%), continuous integration (34.37%), Kanban (68.75%), pair programming (46.87%), refactoring (25%), Test driven development (TDD) (3.1%), small releases (43.75%), and iterative planning (59.37%).

When comparing CBL with traditional teaching methods, participants report that it teaches them how to learn. However, the mentor will help teams to reflect where to go. In this sense, mentors (also instructors) need to have a good knowledge in order to be able to support different students demands.

At the end of the training course, we found that students have improved their learning, according to their own perception of application development and agile development. Using a scale between 0 and 10, in average, students assessment score 2.37 before the course starts and score 7.76 at the end of the course. This result shows that students have increased their knowledge, considering a standard deviation of 2.32 (before training) and 1.36 (after training).

We also identified six recommendations to support this approach. To do so, we asked the students if they could do something differently next time, and they pointed the following:

- Promoting team engagement;
- Adoption of version control system early in the process;
- Promote team reflections during the learning process;
- Keep focus on the learning process;
- Reduce failure by reducing teams exchange; and
- Adoption of the right tools and more agile practices to support the projects learning and execution.

The analyze of student and instructor role has shown that their roles are more active compared to the traditional teaching approaches. It happens because instead of students following instructors content exposure, students need to have attitude, which causes students engagement on knowledge construction. Due to the fact that our approach exposes a situation where students have to build their own knowledge, it increases their engagement. The instructor focuses more on steering a knowledge path.

4 CONCLUSION AND FUTURE WORK

In this study, we present results of a case study conducted in four universities located in Brazil. Our study investigated the adoption of CBL combined with agile practices for training students in MAD environments.

Our results show that our approach which combines CBL and agile practices support the learning process in SE. Thus, we conclude that our approach can be used as a training approach for developing mobile applications, teaching agile practices, as well as SE practices. In addition, the user experience and how students evaluated their learning at the end of the training course, provide us a set of recommendations to support our approach.

As future work, we are planning to expand the evaluation of our approach not limited to MAD environments, but also to other application development domains. Future studies will look for better understand how our approach can improve the learning and teaching process on other software development environments. Additionally, we are looking for explore how our approach can help us to promote a change in the current teaching practices adopted in classroom.

REFERENCES

[1] H. Beckman, N. Coulter, S. Khajenoori, and N.R. Mead. 1997. Collaborations: closing the industry-academia gap. *Software, IEEE* 14, 6 (Nov 1997), 49–57.

[2] Barry Burd, João Paulo Barros, Chris Johnson, Stan Kurkovsky, Arnold Rosenbloom, and Nikolai Tillman. 2012. Educating for Mobile Computing: Addressing the New Challenges. In *Proceedings of the Final Reports on Innovation and Technology in Computer Science Education 2012 Working Groups (ITiCSE-WGR '12)*. Haifa, Israel, 51–63.

[3] Shihong Huang and Damiano Distante. 2006. On Practice-Oriented Software Engineering Education. In *19th Conference on Software Engineering Education and Training Workshops (CSEETW'06)*. 15–18.

[4] L. Johnson and S Adams. 2011. *Challenge Based Learning: The Report from the Implementation Project.* Technical Report. The New Media Consortium, Austin, TX, USA.

[5] Laura Naismith, Peter Lonsdale, Giasemi Vavoula, and Mike Sharples. 2013. *Literature Review in Mobile Technologies and Learning.* Technical Report. University of Birmingham.

[6] Mohammad Nauman and Muhammad Uzair. 2007. SE and CS Collaboration: Training Students for Engineering Large, Complex Systems. In *20th Conference on Software Engineering Education and Training (CSEET'07)*. 167–174.

[7] M. Nichols, K. Cator, and M. Torres. 2016. *Challenge Based Learning Guide.* Digital Promise, Redwood City, CA, USA.