

Peel the Onion: Use of Collaborative and Gamified Tools to Enhance Software Engineering Education

Naomi Unkelos-Shpigel^(✉)

Information Systems Department, University of Haifa, Carmel Mountain,
31905 Haifa, Israel
naomiu@is.haifa.ac.il

Abstract. As software engineering and information systems projects become more and more of collaborative nature, project-based courses are an integral part of information system (IS) and system engineering (SE) curricula. Several challenges stem from the nature of these courses, the most significant are equal participation of all students, and creating projects of high quality and utility. Several mechanisms, such as gamification and collaborative tools, were helpful when dealing with these challenges. This paper present a teaching case, where several tools were used, based on the onion model for open source systems, and several motivation theories.

Keywords: Gamification · Collaboration · Education

1 Introduction

In recent years, the software and information industries has recognized the contribution of a collaborative projects, and projects are being developed according to agile manifesto [2], focusing on continuous review among practitioners, within the development team, and among teams. However, information system (IS) and software engineering (SE) students are not being trained during their studies to this type of collaborative work.

Project-based courses, where the students are required to develop a prototype as final assignment, are an integral part of IS and SE degrees curricula. The teachers of these courses face several challenges, such as ensuring equal participation of all students in the workload, and creating projects which will be both of high quality and utility, all in parallel to teaching a large amount of theoretical background. This paper describes the result of teaching several courses of this nature, mainly those requiring developing a prototype or a website, where several gamified and collaborative tools were used. The teaching method described in the paper can be used by other lecturers teaching similar courses. To actively receive feedback on their work, students were asked to test potential user's response to the prototype. To encourage interaction and collaboration among students, each group presented their project to other students in class. Each student was asked to share thoughts on other group projects, using an online form. This interactive method of experiencing other group's work, while

presenting their own work, resulted in much positive feedbacks about the projects and the assessment method.

This paper presents the summary of a teaching case of a course in the discipline of software engineering. The teaching method drew inspiration from the onion model for OSS (Open Source Systems) [1], and motivation theories such as SDT – Self Determination Theory [7], and the flow theory [9]. These motivation theories discuss how to motivate employees to take active part in the work, and to encourage them to strive for more productive behavior, mainly by encouraging intrinsic and extrinsic motivation [7], and by achieving a state of flow, where the worker is immersed into the task [9]. The Kahoot! Application was used each lecture, to test students' knowledge from the previous lecture, and to present the results to all the class in a gamified manner. GoogleDocs were used to perform additional collaborative exploratory assignment during class. Google forms were used for a collaborative peer review.

Leveraging on the principles of collaborative and gamified tools for education, several questions: (1) How can we promote software engineering students productive behavior via using collaborative and gamified tools? (2) What are the benefits of embedding collaboration and gamification techniques in software development education?

The next section presents the background for the teaching case. Section 3 details the teaching method. Section 4 presents students' responses to the teaching method, and Sect. 5 discusses the conclusions and further evaluation.

2 Scientific Background

2.1 Collaborative Tools in Education

Collaborative tools have been used in high education for several decades [3], and in recent years, Web 2.0 tools are used in education, as part of an ongoing effort to enhance motivation among students. However, teachers' perception towards the use of these tools are still ambiguous, as many teachers do not believe these methods can help to raise student motivation.

Several attempts were made to use several tools [6], where students worked with various tools during the semester, and concluded that these days it is better to use experimental teaching, where students take an active role in class, rather than frontal teaching. However, additional research is in order in order to inspect collaborative work and its outcomes in SE and IS education.

2.2 Gamification in IS and SE Education

Persuasive technologies, and specifically Gamification, were acknowledged as changing employees' motivation and behavior. Gamification is defined as “the integration of Game Mechanics in non-game environments to increase audience engagement, loyalty and fun” [5, p. 2]. Gamification research from the last few years is targeted on using gamification mechanisms for changing behaviors of specific populations. In the context of IS and SE students [4, 6, 8]. In example, in order to engage

software engineering students in development, documentation, bug reporting, and test coverage, using social rewards. The students who used these systems showed statistically proven improvement in their work results. A good example is [4], which identified three types of activities needed to be performed when engaging gamification into software engineering: analysis, integration, and evaluation, and found that students performing these activities had better results in software engineering.

These teaching cases and their outcomes are a strong motivation for using collaborative and gamified tools in SE education. The teaching method is described in the next section.

3 Collaborative and Gamified Teaching Method

3.1 The Method

This section describes the teaching method, used in an advanced software engineering course, intended for third year undergraduate students. The course consisted of 60 students, who were working in teams of four, developing a software product. All the teams received the same assignment – to build a plugin for Eclipse IDE, which provides a gamification tool for performing code review. The programming methodology was agile, so the students worked in three iterations. First, the students built a simple example of their project, and planned the iteration and their outcomes. The teams also implemented all the internal structure of their projects, mainly classes and APIs. At the next iteration, the teams completed the GUI of the system. At the last iteration, the teams performed final product integration, along with integration tests of the components.

The teaching method presented in this paper was inspired from the onion model for OSS projects [1]. This model aims to achieve quality of the open source software product by differentiating between contributors types - the core development team (core of the onion), contributing developers (first layer), bug reporters (second layer), and users (third layer). Each of these groups has its own contribution to the product, according to the definition of its role. In this teaching case, the onion model was an inspiration a model that reflecting the use of collaborative and gamified tools in education:

Student level – At the beginning of the lecture, after a brief recap of the previous lecture, students participated in a Kahoot! quiz. Each quiz included 5 questions regarding the material of the previous lecture. The students were asked to participate using their personal id number, and answered the quiz using their personal computers or smartphones. During the quiz, the students accumulated points, according to the correctness of their answers, and how fast they answered. Finally, a winner was chosen automatically by the application.

Team level – Toward the end of the semester, the students learned about cloud computing solution for software products. The students were given an assignment during class, where a case study was presented to them (based on a true story), and they were asked to provide a cloud computing solution, explaining why they chose this particular solution. They were instructed to work using Google Docs, where each

student used their laptop or smartphone, while the document is written collaboratively by the team. They were told that the documents will be checked after the lesson is over.

Course level – In the middle of the semester, in order to encourage interaction and collaboration among students, each team presented their project to other students in class. Each student was asked to share thoughts on other group projects, using an online google form. As there were 15 teams in the class, each student was asked to review at least seven other teams, in order to receive credit for the assignment. In this interactive method of experiencing other group’s work, each student experienced both the role of presenter of their team work, and a reviewer of other teams’ work (Fig. 1).

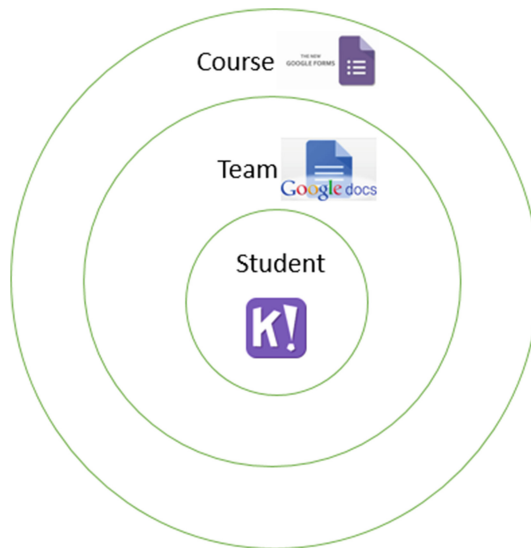


Fig. 1. The onion model for tools used by students

3.2 Analyzing the Teaching Model from a Motivational Perspective

Student level - According to the theory of flow, the use of Kahoot! supports the five elements of flow [9]: *Clarity* – The questions and game scoring are simple and clear; *Centering* – players feel they are in the center, gaining individual points as they answer the questions; *Choice* – As the game includes multiple-choice questions, choice is part of game participation; *Commitment* – each student wants to answer the question correctly as they perceive the success in answering correctly to be directly linked to succeeding in the course; *Challenge* - the game provides challenge to all the stakeholders in the process, when they are required to improve the quality of their work in order to earn additional individual and team points.

Team level - Using Google Docs in the context of team task supports the group flow elements [9]: *A compelling, shared goal* – all the players have the same goal of getting a high team score; *A sense of being in control* – since each player sends their work when they choose, each of them has full control on their progress in the game; *Blending*

egos – since there is a team goal, along with the personal goal, all the players' egos are blended to achieved a higher team score; *Equal participation* – each of the players are allowed to participate in the writing the document equally; *Familiarity* – all the players in the same team are personally familiar; *Constant, spontaneous communication* – the Google Docs environment allows all the players to communicate with each other; *The potential for failure* – the players are aware of the fact that if the result of their work will not promote the overall product, it will considered as a team failure.

The course level – This method offers rules and regulation (in the form of a scheduled task), to encourage students to ask for review, and to share their knowledge with peers outside of their team, thus creating extrinsic motivation. Thus, the method supports the theory of SDT – Self Determination Theory, which can be used to encourage intrinsic motivation of employess engaged in SE tasks [7].

Additional factors in this task are in correlation with the principles of SDT: Team work was done under uncertainty; Having the students reviewing each other, expressing competence and autonomy; Students promoted their work to get review, thus expressing relatedness to their work.

4 Students' Responses to the Teaching Method

As mentioned before, at the end of the course, the students filled out a summary questionnaire, where they were asked about their opinions about the different tools used in the course, and whether they think the tools were helpful for them in understanding what is SE. Several comments for each level are presented below:

Student level

"I really liked the fact that it summarized an entire topic with a few questions"

"Helped to encourage student participation in the class"

"Nice and refreshing, very original, helped to understand the material"

Students were very enthusiastic about using the Kahoot! application. At the beginning of the lecture, after a brief recap, they were happy to pull out their smartphones and participate in the game. Interestingly enough, though the material was presented to them again a few minutes before, some of them still got the answers wrong. However, they still enjoyed the game and applauded the winner of each game. At the end of the semester, they asked for the questions, saying that reviewing the questions would help them during exam preparation.

Team level

"A collaborative tool that enhanced participation of the students"

"High usability for us as users"

"Great tool. This is the how we should learn nowadays"

Students had trouble at first to login to the joint document. When they started working, most of the teams fully participated. Some teams even stayed after the lecture to continue and work on the document. Students found the tool very helpful, completed the task quickly and successfully, and reported that they continued to use this tool later on, working on the project documentation and tasks in other courses.

Course level

“The interactive way of learning helped me in constructing the prototype”

“I learned that having a simple and easy to use application is highly important, so the users keep on using it”

“Reviewing other teams’ prototypes helped us to come to improve our product”

Students responded very positively to the task, and spent more than an hour reviewing other teams. About 500 reviews were written, while teams were trying to draw other students to review them. Students reported they enjoyed the task, compared themselves to their peers and received many insightful ideas for improving their work. Each group got the reviews their work received, anonymously, so they could use them and improve the prototype. Students reported this experience was unprecedented in their studies. They said it was the first time they felt how is it like to be on both sides of the aisle –developers who need to defend their work, and reviewers who inspect other teams’ work, and to write a critique.

5 Expected Contribution and Further Evaluation

In this teaching case, students used several collaborative and gamified tools, encouraged to participate in the student level, team level, and course level. According to their feedback, they felt the use of tools enhanced their learning experience, and assisted them in the development of the project. According to their responses, this experience of collaborative work, and reviewing the work of peers, was a good preparation for their future roles in industry, and to their overall performance in the course.

Future qualitative research is needed in order to explore the effect each tool has on the student, the collaboration within the team and between the teams. Quantitative research can be beneficial to understand if students’ performance in the final exam was in correlation with their achievement in the various levels of the model.

References

1. Aberdour, M.: Achieving quality in open-source software. *IEEE Softw.* **24**(1), 58–64 (2007)
2. Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Kern, J.: Manifesto for Agile software development (2001). http://academic.brooklyn.cuny.edu/cis/sfleisher/Chapter_03_sim.pdf
3. Brown, S.A.: Seeing Web 2.0 in context: a study of academic perceptions. *Internet High. Educ.* **15**(1), 50–57 (2012)
4. Dubois, D.J., Tamburrelli, G.: Understanding gamification mechanisms for software development. In: Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering, pp. 659–662. ACM (2013)
5. Deterding, S., Khaled, R., Nacke, L., Dixon, D.: Gamification: toward a definition. In: CHI 2011 Gamification Workshop Proceedings, pp. 12–15 (2011)

6. El-Masri, M., Tarhini, A., Hassouna, M., Elyas, T.: A design science approach to Gamify education: from games to platforms. In: Twenty-Third European Conference on Information Systems (ECIS), pp. 26–38 (2015)
7. Ryan, R.M., Deci, E.L.: Self-determination theory and the facilitation of intrinsic motivation, social development, well-being. *Am. Psychol.* **55**(1), 68 (2000)
8. Sheth, S.K., Bell, J.S., Kaiser, G.E.: Increasing student engagement in software engineering with gamification, pp. 43–47. *ACM* (2012)
9. Unkelos-Shpigel, N., Hadar, I.: Gamifying software development environments using cognitive principles. In: *CAiSE Forum 2015* (2015)