

# Gamifying research in software engineering

Mariusz Nowostawski  | Simon McCallum | Deepti Mishra 

Department of Computer Science, Faculty of Information Technology and Electrical Engineering, NTNU—Norwegian University of Science and Technology, Teknologivegen, Gjøvik, Norway

## Correspondence

Deepti Mishra, Department of Computer Science, Faculty of Information Technology and Electrical Engineering, NTNU—Norwegian University of Science and Technology, Teknologivegen 22, 2815 Gjøvik, Norway.  
Email: [deepti.mishra@ntnu.no](mailto:deepti.mishra@ntnu.no)

## Funding information

NOKUT under the Centre for Excellent IT Education (ExcITEd) (Norway), Grant number: 16/02049

## Abstract

It is a non-trivial task for research-centric courses in the software engineering curriculum to compete and engage students on the same level as the practical, software development courses. Practical software development courses and projects are inherently motivating to students, as they provide necessary elements such as agency, relatedness, and the strong sense of competence upon completing software engineering tasks. In contrast, reading research articles and technical white papers feels dry and non-engaging. Nevertheless, a well-balanced MSc programme curriculum covers both, learning through construction and practical courses as well as research courses. The main motivation for the development of game-centric approaches to the research aspects of curriculum is to improve students' interest and engagement with those courses. In this paper, we present the methodology and initial evaluations from three gamification strategies used in the Master's degree programme. These are: the game of reading and discussing research articles (GoRaD), the game of arguing and counter-arguing, and combining research and practice. The paper presents our experimentation and initial evaluations of the use of those strategies, as well as plans for future development and enhancements.

## KEYWORDS

academic education, gamification, postgraduate courses, programming, software engineering

## 1 | INTRODUCTION

Software engineering education [33] is an important subfield of software engineering focusing on education topics for software engineering e.g., how to better teach and train software engineering skills. It is the responsibility of software engineering education to prepare software engineering professionals by providing them with the skills to meet the expectations of the software industry [22–24]. Educators are incorporating simulation [2,25], games [16,26,27]; as well as different pedagogical approaches such as inquiry-based learning [19] in their courses. Software engineering education at postgraduate level, in addition to the aforementioned areas, also includes

research and preparation for research. Those aspects are qualitatively different from acquiring skills in developing software projects. A researcher needs deeper preparation in underlying principles, in problem formulation, and in validation of results [31] as well as a special kind of inquisitiveness and creativity [33]. For a typical software engineering student, learning by construction and software projects are inherently engaging. In contrast, the research elements pose specific challenges in software engineering education. Games are one method that can be used to immerse learners within an experience and create a lasting impression of the content [21].

The Master of Science in Applied Computer Science is an international 2-year degree programme focused on Web,

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2018 The Authors. *Computer Applications in Engineering Education* Published by Wiley Periodicals Inc.

Mobile, and Games themes. The MSc programme consists of a number of general, and research-specific courses, as well as software engineering projects. The degree programme provides specialization in three application domains of computer science, namely, Mobile/Wearable computing, Web Technologies, and Games/Gamification/Serious Games. Students typically pursue research in one or two of these areas. At the Masters level, a degree in software engineering and programming consists of both research and development activities. Students are taught how to conduct research within software engineering as well as the engineering and development practices, methodologies, and tools.

Our teaching methodology is inspired by Self Determination theory (SDT) [6]. We have observed that most software engineering projects are highly engaging and motivating for students. Learning by construction, i.e., the development of a software system is intrinsically motivating for our students. The projects provide the students with agency, i.e., the students can select or create their own projects, tools, and software engineering methodologies. Competence and relatedness are also relatively easily achieved in software development projects. Unfortunately, it is harder to engage students in the research aspects of the R&D elements of the MSc programme. Thus, most of our gamification work has been focused on gamifying the research components of a Masters level degree. Gamification can be defined as using game-based mechanics, aesthetics, and game thinking to engage people, motivate action, promote learning, and solve problems [11,12].

MSc degree courses include the traditional process of reading and reviewing research articles. This is to facilitate students acquiring necessary skills for comprehension of research, critical and logical thinking, and the skills of argumentation. Ideally, students would read the suggested material before the lecture, and discuss the various aspects in class. However, students' motivation to read the research papers before the lecture session is a continuous challenge. The desired outcome is that students should read the research papers, reflect on the content of the research, create relevant, and interesting questions about the research in the paper, and finally, write or verbally discuss the qualities of the research and the article's presentation.

This paper presents our work focused on gamification of the research components of a Software Engineering Masters level degree. The challenge for the gamification is to engage students in the research activities. To achieve the desired goals, we use three strategies in various courses, which put the students at the center of the activity. Those are: the Game of Reading and Discussing, oral/written argumentation exercises, and linking research and practice.

This paper is organized as follows: In the next section, literature review has been provided with respect to Self Determination Theory (SDT), inquiry based learning, and critical thinking. The subsequent section illustrates background

of this study along with description of the strategies used in various master level courses. Section 4 presents the evaluation of these strategies. The final section infers the conclusions, future work and limitations in this area.

## 2 | RELATED WORK

### 2.1 | Self determination theory in higher education

SDT posits that humans come with the needs for relatedness, competence, and autonomy [6]. Note, that in this article we use the term "autonomy" and "agency" to mean the same concept. Some authors from the field of Gamification and Game Design use the term "agency," whereas others focus on "autonomy." Even though those two terms are not perfect synonyms, for the purpose of the article we often use them interchangeably. When the needs for relatedness, competence, and autonomy (agency) are met by activities or social contexts, people will find the tasks meaningful and continue to participate in them [34]. However, when these needs are thwarted, individuals will become disaffected and withdraw [34]. The satisfaction of learners' needs for autonomy, competence, and relatedness facilitate them to move along the learner's autonomy continuum from dependence to autonomy [10]. Increased relatedness to peers and faculty and increased higher-order thinking assignments are found to be substantial predictors of educational outcomes relevant to literacy, critical thinking, and, especially, job preparation [3]. Danyaro et al. [5] evaluated the patterns of use and behavior of tertiary level students based on SDT towards the use of Web 2.0 as an alternative and supplemental e-learning portal. They revealed that students have an inherent desire of expressing ideas and opinion online openly and independently and this sense of freedom makes students feel more competent, autonomous, or participative and find learning to be less tedious [5]. Lamprinou and Paraskeva [13] described the design process, implementation, and evaluation of a meaningful gamified online course with the use of structural and content gamification to increase student intrinsic motivation based on the Self-Determination during the learning process. Laskowski et al. [14] focused on gamifying the study program and its various aspects and Hew et al. [9] conducted a study based on SDT with an additional element of game mechanics i.e., points, badges, and leader boards and found that the contribution of students increased in the discussion forums, but there was no significant difference on students' recall of factual knowledge. Further, they found that the use of game mechanics had a positive effect on motivating students to engage with more difficult tasks, and that the quality of artifacts produced by participants in the experimental groups were higher than those in the control groups [9]. Rosenkranz et al. [32] provided evidence for the motivating effects of competence and

relatedness in relation to medical students doing research and suggested that well supported compulsory research activities that incorporate group learning and elements of choice may promote motivation to do research, and potentially, careers in research, even in a research naive student body. Iosup and Epema [11] provides a comprehensive overview of using gamification in higher education. In their report, the authors have used social gamification techniques in two courses in one of the technical universities in Netherlands. One BSc course focused on Computer Organization and one MSc course focused on Cloud Computing. They found gamification to be correlated with an increase in the percentage of passing students, and in the participation in voluntary activities and challenging assignments. Gamification seems to stimulate interactions in the classroom.

## 2.2 | Inquiry-based learning

Inquiry-based learning is a question driven approach for active learning. Inquiry-Based Learning (IBL) is a student-centered pedagogy that focuses on questioning, critical thinking, and problem solving [18]. The basic idea of IBL is asking meaningful questions, which may be established by students, teachers, or by negotiation among them [15]. Inquiry based learning is often described as a cycle which implies five phases: ask, investigate, create, discuss, and reflect [28]. Students are engaged in the learning activities, and the focus is shifted from teacher-led to student-centered learning [29]. Perry and Richardson [30] presented the Master of Science teaching program in which graduate students are required to develop inquiry-based curriculum for their respective classrooms and laboratories from what they experience as they pursue a graduate degree in science teaching. Wu et al. [36] proposed a knowledge exploration assistant system for inquiry-based learning to support Computer Science university students with instant assistance with proper knowledge materials and knowledge exploration functions during their learning process. Marques et al. [19] developed a self-motivating and inquiry-based educational game to teach software visualization and visual programming to university-level students. Acosta et al. [1] presented the expansion of the SMILE project (Stanford Mobile Inquiry Based Learning Environment), called RecQuest, in which teachers propose a research topic and provide students initial reading material about that topic that further drives students to create and share questions and answers using mobile devices.

## 2.3 | Critical thinking: arguments and counter-arguments

Critical thinking is generally recognized as an important skill, and one that is a primary goal of higher education [7]. Critical thinking skills are interpretation, analysis, evaluation,

inference, explanation, and self-regulation [8]. According to Mason [20], much of the rhetoric regarding education and its reform revolves around teaching students to think and question critically. Wei [35] showed that students need to challenge their peers and ask intriguing and open-ended questions in order to promote construction of knowledge online and to facilitate the critical thinking of students through collaborative interaction. Furthermore, it was found that facilitation of critical thinking skills using wiki happened where students reflected on and synthesized credible information, explained their own ideas in the wiki and through incorporating others ideas [35]. Liu et al. [17] described an effective web-based learning strategy, peer review, and revealed that students not only performed better under peer review, but also displayed higher level thinking skills, i.e., critical thinking, planning, monitoring, and regulation. The most effective individual appears to be the strategic adapter who effectively constructs a project, adjusts to peers comments, and serves as a critical reviewer as well [17]. Cismas [4] described RWCT (Reading and Writing for Critical Thinking), a new research-based, instructional methods able to help students think reflectively, take ownership of their personal learning, understand the logic of arguments, listen attentively, debate confidently, and become independent, life-long learners by promoting active inquiry, student-initiated learning, and refinement of problem-solving skills, critical thinking, and cooperative learning.

## 3 | RESEARCH METHODOLOGY

### 3.1 | Background

Self Determination Theory (SDT), a motivational meta-theory, which is used in game design as well as general behavioral intervention analysis, suggests that Agency, Competence, and Relatedness are important psychological factors that influence the sense of intrinsic motivation and engagement. The structure of our degree, as well as the choice of learning methods reflects the fundamental principles of the theory, which we use as a theoretical basis for a gamification of the core research activities in the degree. Our teaching environment is structured such as to influence and increase the three main factors highlighted by the SDT: Agency, Competence, and Relatedness. The theory itself is at the core of the Games and Serious Games track, due to the track focus and teaching material used. Students are taught about SDT and other gamification theories and methodologies. However, the use and applicability of SDT is not limited to Serious Games track. It influences the other two tracks of our MSc programme: Mobile and Web. We use a number of methods to increase the intrinsic motivation of students, as well as their engagement through application of tactics and teaching methods that appeal to all three aspects of the

Self-Determination theory. One of the key features of good educational programme is engaging students with the teaching material. There are many techniques used to put the student in the center of the learning environment. We try to achieve this through the choice of exercises, engagement methods, and the way students work with the teaching materials. This includes both in-class activities and those conducted by students independently at home.

### 3.2 | The degree structure

The MSc programme takes 2 years split into four semesters. Semester 1 is dedicated to Scientific Methodology and Communication course, plus various introduction courses within the Games, Mobile, and Web tracks. In addition, Semester 1 contains the Applied Computer Science project course. The Applied Computer Science course is the self-motivating, group project focused on development of a usable prototype for external stakeholders. The other courses are generic in nature or research focused. Semester 2 contains the remaining research introduction courses (research-based), Experts in Teamwork (teamwork-focused course) and the Integration Project course. The Integration Project course is a development course that pushes students to integrate multiple technologies in a single, coherent project. Semester 3 is a preparation for the MSc project, and it includes Advanced Project Work, Project Planning, and specialization courses pertinent to students' MSc topic. All subjects in Semester 3 are research focused. The final semester (S4), involves thesis work and thesis writing. In these first two semesters, the study material is broader, and applicable to the wider audience of the class. Semesters 3 and 4 focus on individual activities. The students work on their own projects and focus on their own area of expertise. The gamification elements within the degree are used mostly in Semesters 1 and 2, where students need to engage with each other.

### 3.3 | Questioning and question quality

In GoRaD—subsequently renamed as EduLab—and Essay counter-arguing (Figure 1), the fundamental aspect that we are focusing on is questioning. Questioning is, broadly speaking, essential for any software engineer, where most of the activities are focused on either: questioning, or answering questions. One of the highlights of this phenomenon is through the StackOverflow systems, where questioning and answering questions is a fundamental aspect of the system itself. Debugging is another example where properly formulated questions lead to insightful answers, whereas purely formulated questions distract and lead astray. Each set of values entered while debugging can be considered a question. Entering the values that reveal the nature of the bug lead to finding the bug much faster.

The quality of an answer is often related to the quality of the question, and therefore, students, or software engineers in general, should be encouraged to improve their questions quality. The challenge for educators is how to improve question quality?

In our curriculum, we focus on three dimensions of questions, namely: (1) is the question answerable; (2) is the question relevant and interesting to the domain of inquiry; and (3) how succinct is the question. There are many other possible dimensions in which question quality can be assessed. In the context of the aforementioned tools and game-like systems, we have focused on those three aspects.

### 3.4 | Teaching method example: GoRaD

The Game of Reading and Discussion (GoRaD) is the result of multiple iterations of designing and testing various gamifications of the core analysis activity. The challenge we are addressing through the game is to engage software engineering students in critical reading of research articles and technical specifications before the class. The first iteration was adding the requirements that students submit questions about the research papers and technical documents set for the next session, and score the quality of each other's questions (Figure 2). Note, students often need some time to get accustomed to the system, and “play” with it, by posting low-barrier entry questions, and we often spend one or two lectures in “play” mode introducing the student to the system and encouraging the discussion, questions (Figure 2). The next iterations included writing reflection statements about the research and grading each other's reflections. Finally, the present form allows questions, evaluations, reflections, and discussion to be conducted through the tool and for these interactions to be scored and become part of the social interactions in the course. See Figure 3 for example of question scoring and discussions during one of the cryptocurrencies seminars for which the system was used.

The GoRaD system has gone through three iterations of technology, initially being written in PHP and SQL. After the initial release, the system has been re-written in JavaScript and Angular 1 framework, on the front-end and Firebase-based backend. The latest implementation uses Typescript and Angular 4<sup>1</sup> Google Material Design UI<sup>2</sup> for the front-end. On the backend, we use JavaScript together with Loopback.io<sup>3</sup> REST framework with Node.js, and MongoDB as a datastore. The process of creating the current version, as well as the methodologies used, testing frameworks, and integration tools are also used as an example of Software Engineering practices for students. The methodology uses an issue tracker, version control system (Git) with branching, automated building

<sup>1</sup><http://angular.io>

<sup>2</sup><http://material.io>

<sup>3</sup><http://loopback.io>

## EduLab

**NTNU**  
Norwegian University of  
Science and Technology

## IMT4306 2018

IDI Gjovik   Organizer: NTNU   Venue: Usually C026   Hashtag: imt4306-2018

^

IMT4306 Introduction to Mobile/Wearable Research.

Add New sub-event (+)   Edit This Event

### Sessions

07:25-07:25 **IMT4306 2018 essays #2**  
NTNU  
Essays session #2. All essays for second round should come under this category.

07:27-07:27 **IMT4306 2018 essays #3**  
NTNU  
IMT4306 2018 essays #3. All essays from round #3 should come under this subevent.

### Essays

- ... Evaluating chal ... (4632 characters) View
- ... Overview of pri ... (9935 characters) View

**FIGURE 1** Screenshot of the system for MSc course in Mobile/Wearable research. Students are required to submit three rounds of essays which are subsequently peer-reviewed and counter-argued by other students in the course

system, deployment, and unit testing. This allows students to not only experience the tool itself, but also, contribute to the issue tracker as well as the development of various features of the system itself. The various technology stacks were chosen at the time of development, therefore, Angular 1 was used initially, and subsequently updated to the new version of the framework. The original Firebase storage turned out to be limiting, and the decision was made to move the backend storage to MongoDB for flexibility and robustness. The Loopback framework seemed a natural choice to provide Restful API in declarative form, as opposed to using more imperative library to achieve the same effect.

GoRaD was initially a simple online system that allows students to post questions related to research article, and see questions posted by others. We have subsequently added the ability to vote on questions submitted, and later, changed it to weighted allocation of points across multiple submitted questions each week before the class. The system supported

students' engagement with research papers by providing Competence measures related to scoring points based on the quality of the questions students asked. Other students allocate points to the questions you ask, and you can allocate points to questions others formulated. The lecturers for the course include their own questions, and as all questions are anonymous to other users, the lectures can also engage in trying to write high quality, high-scoring questions.

The use of other students in the course as part of the review and scoring system of the game supports the need for Relatedness, as the other students in the group are integral to performing well. There is also a high score list to provide an incentive for students to have the best question in each week, and an overall list for the semester. Supporting Agency is easy within a games context as the player is the focus of the game, and the interaction of the player with the rules and context of the game is the core of the difference between games and film. Within a teaching context, where

## Questions

Sort latest up ↻

Sort best up 👍

This is not a question, but a statement 👍<sup>3</sup> 🗨️<sup>0</sup>

emoji test 🤪 👍<sup>0</sup> 🗨️<sup>2</sup>

## Comments

Sort latest up ↻

Sort best up 👍

The "dark" part of Norway is a bit further north than where we are right now. 👍<sup>4</sup> 🗨️<sup>0</sup>

Zero Escape: Virtues Last Reward is a game entirely based around the Prisoners Dilema. I recommend it.

👍<sup>2</sup> 🗨️<sup>0</sup>

The entire Zero Escape series is recommended to play. The games are 999, VLR and ZTD (all are shortened titles). All games will have dilemmas you will have to go through, and there will always be lessons to be learnt in each of the games.

👍<sup>1</sup> 🗨️<sup>0</sup>

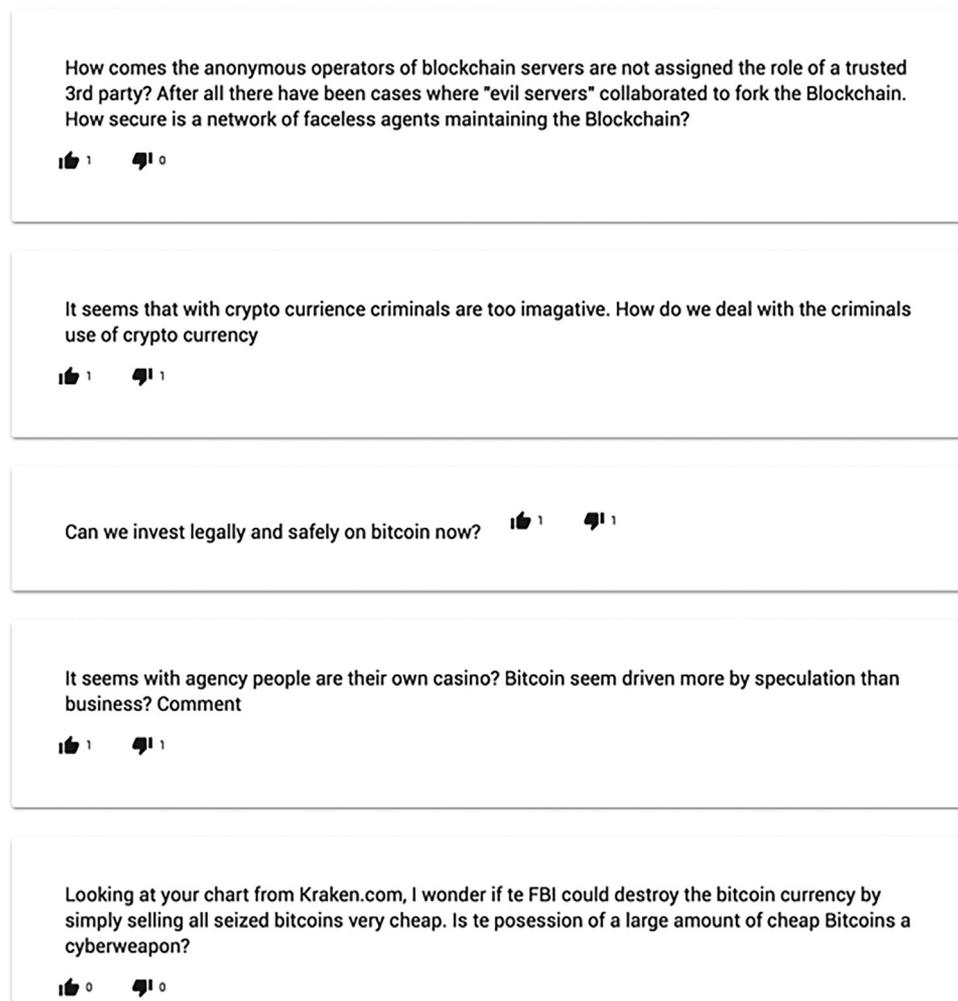
**FIGURE 2** Example comments/questions for one of the courses that the system was used for

there are learning outcomes that must be achieved, agency is naturally lowered, as the student is not making the choice. To counter this we allow student to choose both the area of the Mobile or Serious Games that they want to focus on, and also to suggest papers that they have found that are interesting. We provide students with a default option for the research paper to cover. The importance of Agency is not in making a choice, but in the feeling that choices are available.

### 3.5 | Teaching method example: arguing and counter arguing

Critical thinking, the ability to argue your point of view, formulating clear objectives, and logical reasoning are

teaching objectives in multiple courses on the MSc level. All those skills require practice. In addition to standard lectures and assignments, we have experimented with two additional ways of engaging students that enhance their agency as well as relatedness. The students are required to postulate an argument, and then present it, in oral or written form. After formulating their own argument, students have to prepare counter-arguments for other students' statements. There is a friendly confrontation of postulating a thesis, or counter-thesis, and then engaging in discourse to defend or counter-argue another student thesis. These sessions are either verbal presentations, or through our online system that allows students to submit their essays and reviews, comments and counter arguments in written forms. The students are empowered to provide reviews



**FIGURE 3** Example questions from the lecture on Bitcoin and crypto-currencies

and scores for their peers, and the system encourages peer-to-peer activities between the students directly. Students' contributions are read and evaluated by the peers from the class, which provides strong link to Relatedness. Agency is inherent in the task itself. The students are free to choose their own topics, as well as arguments for or against a given thesis. The essays and counter arguments are limited to 1,200 words, and the reviews to 600 words.

For arguing and counter-arguing in written form an extension to the GoRaD system has been added. This extension allows students to submit essays and reviews, as well as counter-arguments (Figure 4). The system allows students to question each other submission, obtain direct feedback, and peer-review each other work (Figure 5).

### 3.6 | Teaching method example: combining research and practice

A number of courses in the Software Engineering MSc programme address the objective of combining the research elements and practice. The purpose is to use the university

curriculum to provide students with the concept of levelling up, in their use of research and state-of-the-art investigations prior to undertaking software development tasks. The practical elements are always adjacent to more research-focused courses. In particular, the research specialization courses are augmented with learning by construction courses that utilize the knowledge and understanding developed through the research. Students engage in research-based construction tasks from the Integration Project, in Semester 2. The research element there is minimal, and the focus of the course lies in software development and in development methodologies. Nevertheless, students are not given a specification of how they should solve the problem at hand. They have to propose their own solutions.

The second instantiation of the same structure, with a much stronger focus on the research elements is the Advance Project Work. The research element here is expanded, and students need to conduct literature review and analyze more thoroughly the state-of-the-art.

The final course that is lined up for this particular objective is the MSc project itself. Here, the balance is shifted

## Essays

- *Evaluating chal ...* (4632 characters)
- *Overview of pri ...* (9935 characters)
- *ng Viability of Of ...* (6011 characters)
- *ang Mobile applicat ...* (4806 characters)
- *apa Wearable Techno ...* (4680 characters)
- *Stra ... Mobile Performa ...* (0 characters)
- *kog Cross-platform ...* (6746 characters)
- *høl ... Security threat ...* (4498 characters)
- *Ler ... The Importance ...* (5195 characters)
- *viado ... Personality of ...* (4952 characters)
- *Hole ... Monetizing and ...* (0 characters)
- *Software Engine ...* (8354 characters)
- *usten Using wearables ...* (5689 characters)
- *vik Ethical challen ...* (10141 characters)
- *ng Mobile Cloud Ga ...* (4887 characters)
- *apa Wearables assis ...* (4208 characters)
- *Stra ... Mobile Phones a ...* (6969 characters)
- *Mobile Exercise ...* (8156 characters)
- *Progressive Web ...* (0 characters)
- *Ler ... Using Smartphon ...* (6298 characters)

## Questions



FIGURE 4 Sample essays submitted by students

even more from the construction, and software development side, towards the research and research methodologies.

One could say that software engineering students are leveling up through those three courses, to obtain the necessary skills and capabilities in research-driven software design and development.

## 4 | EVALUATION OF METHODS

It is worth noting, that the MSc programme graduates, almost exclusively go to the workforce in the software development

industry, not in academia. The percentage of graduates that proceed into academic career after finishing their MSc degree, for our campus students is less than 5%. This means, that for the majority of students, the MSc curriculum must provide adequate industry preparation, with relevant practices and know-how, to benefit the software development community at large. Our choice of techniques and strategies to improve student engagement takes this into account. For example, the skill of reading and formulating critical assessment of white papers and scientific publications is one of the necessary skills for any senior software developer or system architect. Requiring those skills is a time-intensive process that requires practice, in the same way as many other aspects of software development skills-base. The games or game-like incentive systems have been designed to provide long-lasting effects. However, the evaluation and success metric provide us with their own challenges that we describe in this and the following section.

In our MSc courses, we typically have between 10–20 students (Table 1). Setting up randomized control groups with such small class sizes has not been practical. Therefore, we have used previous years that has not used the system as a baseline for comparisons. We have used qualitative (student self-evaluations, testimonials) and quantitative evaluation methods (number of student participating, number of questions asked, quality of questions, and so on) that do not provide objective metrics for the evaluation of impact of our system. The use of different student years as control is not ideal, as it does not account for confounding factors such as exact learning material, the average skill level or motivational aspects as well as other curriculum aspects affecting individual courses. Nevertheless, we believe that the qualitative and subjective evaluations do help in providing evidence and future direction of this line of educational improvements.

### 4.1 | GoRaD

The key measurement of success for this type of engagement is the actual interaction with the students. Below, we present both quantitative numbers for interactions as well as qualitative feedback from students. Given the limited number of students undertaking MSc degree programme, it has been challenging to provide solid research results as to the effectiveness of the proposed strategies. Nevertheless, we obtained feedback and evaluation from students, through discussions and qualitative evaluations.

The use of a system that actively rewards reflections and provides feedback for improving analytical skills is popular among the Applied Computer Science students. The game aspects do not overpower the importance of analysis and reflection, so the students can still see that the core activity in the course is learning to be a researcher, rather than merely being good at the GoRaD game.

The GoRaD game has been used for 4 years (2015–2018), in both, Mobile and Serious Games specialization courses.

## Final thoughts

From the solutions and research discussed in this essay, it has shown that from a technological standpoint that replicating a fully immersive user experience. The issue is that this is not yet possible with commercial VR equipment. Even with a Kinect or additional HTC Vive sensors it would not be accurate enough to reflect real world interaction. As mentioned in [2] wearing an excessive number of sensors is cumbersome, and can create a bad user experience as it further limits your movement. There is very little research done on this topic. The proposed solution such as [2] discussed in this essay, were one of very few research papers discussing the topic of immersive user-experience during social interaction in VR. Other than this a large portion of research done exceeds the realm of wearable technology research and enters the disciplines such as gamification and general graphics programming.

## References

- Samantha Anderson "You're the center of the universe: A UX guide to designing virtual reality experiences" [Online Article](#) (2018).
- Daphne Economou, Ioannis Doumanis, LEMONIA ARGYRIOU, NEKTARIOS GEORGALAS "User experience evaluation of human representation in collaborative virtual environments" [Personal and Ubiquitous Computing](#) (2017).
- Steven Hickson, Nick Dufour, Avneesh Sud, Vivek Kwatra, Irfan Essa "Eyemotion: Classifying facial expressions in VR using eye-tracking cameras" [Computer Vision and Pattern Recognition](#) (2017).
- Shu-Yu Chen, Lin Gao, Yu-Kun Lai, Paul L. Rosin, Shihong Xia "Real-time 3D Face Reconstruction and Gaze Tracking for Virtual Reality" [IEEE VR](#) (2018).

[Back to the Event](#)
[Edit This Essay](#)
[Hide Reviews, Comments and Questions](#)

## Reviews

- [ten Review of "Creating realistic immersive user experience to simulate social interaction in virtual reality"](#)

## Overall Evaluation I generally think this essay was quite good. Except from a few grammar mistakes and that the title could have been a bit more precise. I think it is well structured and that the [...]

[View](#)
[Add Review](#)

Enter your Question \*

Add Question

## Questions

Sort latest up

Sort best up

Enter your Comment \*

Add Comment

## Comments

Sort latest up

Sort best up

I will review this essay!



**FIGURE 5** Example essay and review submission. The screenshot depicts final section of the student essay, the referencing subsystem, and the submitted review by another student (part of peer-review)

The student groups during that period consisted of groups between 6–12 students. The number of questions asked, ranged from 3 to 9 per week. There are total 242 questions and comments posted across all three courses. On good weeks, the participation would reach close to 80%, and on less-engaging weeks, as low as 30%. However, without the game we have observed typical engagement in the range between 10% to 30%, and we have confirmed with every lecturer using the system the improved engagement overall with the use the system, as compared to previously observed typical student engagement. The typical student engagement was low, and students would not read the pre-assigned articles ahead of lecture. Instead, they would wait for the discussion in class,

and read the articles after the class. However, if none of the students read the article before the class, the discussion in-class would take a form of lecturer monologue. In general, the use of the GoRaD system improved the student engagement in the class. The discussions have grown from 0%–30% active students in the class without using the game, to somewhere

**TABLE 1** Student counts per course, per year

	Serious games	Game technology	Mobile technology
2016	18	18	19
2017	10	8	10

between 30% and 80% of the class when the game has been used. Most students would read the article before the session and post their questions. Quote from students:

*The questions of others made me think of certain aspects of the article that I have not thought of myself.*

*It has been fun to ask questions and see how others like my question.*

## 4.2 | Arguing and counter-arguing

The essay writing and counter arguing form of student engagement through the use of GoRaD system has been used in 2014–2018, in a form of oral confrontations between two students. In 2016–2018, we have added the written form of the same idea, based on essay writing, reviewing, and written form of counter-arguing.

In general, the essay writing and arguing is a popular task that students enthusiastically engage in. Students have enjoyed the engagement (Relatedness) and freedom to argue their own chosen case (Agency), as well as gained confidence and competence in expressing their opinions in written form. In general, each student is required to write approx.  $2 \times 1,200$  words for the main essay and their main counter argument. In addition, they have to write approx.  $3 \times 600$  words of reviews. The reviews range from shorter, more generic ones to longer and fully developed arguments for and against the main thesis provided in the essay. In 2016, the average length of the essay in the class was 1,185 words, with the standard deviation of 147 words. In the 2017, the average length of the essay was 1,291 words, with the standard deviation of 328 words. The slightly longer average and larger deviation in 2017 suggests a more diverse student group, and does not have statistical significance.

In 2016–2017, only one lecturer have used the system, and the lecturer has recorded positive feedback on the activities from the students. The students were engaged and enthusiastic in formulating and expressing their opinions. Quote from students:

*It has been great to read what others thought of my essay. I have got lots of positive feedback, and it make me feel good.*

*The improvement suggestions were great. I have not thought of those things before.*

## 5 | CONCLUSION, FUTURE WORK, AND LIMITATIONS

In this article, we have described three main strategies used to engage and motivate students in the software engineering

MSC programme. Our students specialize in Programming of Web, Mobile and Games technologies. The three strategies used are: the game of reading and discussing (GoRaD), the system for argumentation, reviewing and counter arguing, and the combination of theory and practice in software development courses.

Our initial experiments and testimonials from students provide encouraging results. The results suggest that the use of various gamification elements in software engineering curriculum can provide beneficial encouragement to students, improving engagement and general satisfaction from combining theory- and research-based inputs to the student software engineering practices.

The generic framework we have worked within, Self Determination Theory (SDT), has proved itself as a valuable frame of reference, and provided helpful dimensions on which to focus during the design of the gamification system. The Agency, Relatedness, and Competence are valuable constructs in designing engaging and inherently motivating games. The use of quantified methodologies has been a challenge, given the low number of students in our MSc programme. We have used informal interviews and unstructured feedback from students. This however, provides only circumstantial evidence, and is not sufficient for making generalizable claims as to the effectiveness of the particular gamification methods used. We are planning to engage larger, multi-campus MSc programmes, and this would allow us to compare larger number of students.

The resulting engagement has been improved in class, however, we do not know if students reading of white papers and technical documentations has long-term effects, or, if those in-class effects wear off. One of the potential future areas of study is the analysis of students' engagement in professional and semi-professional dissemination platforms, such as LinkedIn, Medium, open source projects (e.g., on GitHub), and others, to see if the game elements have improved their engagement within the professional software development communities.

## ACKNOWLEDGMENTS

The authors would like to thank the editor, associate editor and reviewers for their efforts in improving the quality of this article. This work was partially funded by NOKUT under the Centre For Excellence in IT Education (ExcITed), project number 16/02049.

## ORCID

Mariusz Nowostawski  <http://orcid.org/0000-0002-2809-8615>

Deepti Mishra  <http://orcid.org/0000-0001-5144-3811>

## REFERENCES

- O. C. Acosta, P. A. Behar, and E. B. Reategui, *Content recommendation in an inquiry-based learning environment*, in: 2014 IEEE Frontiers in Education Conference (FIE) Proceedings, 2014, pp. 1–6.
- R. Atal and A. Sureka, Anukarna: A software engineering simulation game for teaching practical decision making in peer code review. 1st International Workshop on Case Method for Computing Education (CMCE 2015), 2015, pp. 63–70.
- M. R. Beachboard et al., Cohorts and relatedness: Self-determination theory as an explanation of how learning communities affect educational outcomes, *Research in Higher Education*, **52** (2011), 853–874.
- S. C. Cismas, *Rwct implementation in electrical engineering*, in: 2011 7th International Symposium on Advanced Topics in Electrical Engineering (ATEE), 2011, pp. 1–4.
- K. U. Danyaro et al., *An evaluation of the usage of web 2.0 among tertiary level students in Malaysia*, in: 2010 International Symposium on Information Technology, **1** 2010, pp. 1–6.
- E. L. Deci and R. M. Ryan, Self-determination, *The Corsini Encyclopedia of Psychology*, Wiley, Hoboken, NJ, 2010.
- E. P. Douglas, *Work in progress: What is critical thinking?* In: 2012 Frontiers in Education Conference Proceedings, 2012, 1–2. <https://doi.org/10.1109/FIE.2012.6462337>
- P. A. Facione, Critical Thinking: A Statement of Expert Consensus for Purposes of Educational Assessment and Instruction. Research Findings and Recommendations, 1990, pp. 112.
- K. F. Hew et al., *Engaging Asian students through game mechanics: Findings from two experiment studies*, *Comput. Educ.* **92–93** (2016), 221–236.
- P. Hu and J. Zhang, *A pathway to learner autonomy: a self-determination theory perspective*, *Asia Pacific Educ.* **18** (2017), 147–157.
- A. Iosup and D. Epema, *An experience report on using gamification in technical higher education*. In Proceedings of the 45th ACM technical symposium on Computer science education, 2014, pp. 27–32. ACM.
- K.M. Kapp, *The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education*, Pfeiffer, San Francisco, CA, 1st edition, 2012.
- D. Lamprinou and F. Paraskeva, *Gamification design framework based on SDT for student motivation*, in: 2015 International Conference on Interactive Mobile Communication Technologies and Learning (IMCL), 2015, 406–410.
- M. Laskowski, *Implementing gamification techniques into university study path-A case study*. In Global Engineering Education Conference (EDUCON), 2015 IEEE, 2015, pp. 582–586. IEEE.
- P. Levy, O. Aiyegbayo, and S. Little, *Designing for inquiry-based learning with the learning activity management system*, *J. Comput. Assist. Learn.* **25** (2009), 238–251.
- J. E. N. Lino et al., *Project management game 2d (pmg-2d): A serious game to assist software project managers training*. In: 2015 IEEE Frontiers in Education Conference (FIE), 2015, pp. 1–8. <https://doi.org/10.1109/FIE.2015.7344168>
- E. Z.-F. Liu et al., *Web-based peer review: The learner as both adapter and reviewer*, *IEEE Trans. on Educ.* **44** (2001), 246–251. <https://doi.org/10.1109/13.940995>
- M. Liu et al., Using a web enhanced, inquiry-based learning module to increase cultural awareness among middle school students E. Pearson, P. Bohman, *Proceedings of EdMedia: World conference on educational media and technology 2006*, Association for the Advancement of Computing in Education (AACE), Orlando, FL USA, 2006, 1408–1415.
- B. R. C. Marques, S. P. Levitt, and K. J. Nixon, Video games as a medium for software education, in: 2012 IEEE International Games Innovation Conference, 2012, pp. 1–4. <https://doi.org/10.1109/IGIC.2012.6329850>
- M. Mason, *Critical thinking and learning*, *Educ. Phil. Theo.* **39** (2017), 339–349. <https://doi.org/10.1111/j.1469-5812.2007.00343.x>
- J. McGonigal, *Reality Is Broken: Why Games Make Us Better and How They Can Change the World*, Penguin Group, London, UK, 2011.
- A. Mishra and D. Mishra, *Industry oriented advanced software engineering education curriculum* *Croat. J. Educ.* **14** (2012), 595–624.
- A. Mishra and A. Yazici, *An assessment of the software engineering curriculum in turkish universities: IEEE/ACM guidelines perspective*, *Croat. J. Educ.* **13** (2011), 188–219.
- A. Mishra, N.E. Cagiltay, and O. Kilic, *Software engineering education: Some important dimensions*, *Eur. J. Eng. Educ.* **32** (2007), 349–361.
- E. Navarro and A. Van der Hoek, *Simse: an interactive simulation game for software engineering education*. in: Proceedings of the 7th IASTED International Conference on Computers and Advanced Technology in Education, 2004, <http://www.ics.uci.edu/~emilyo/papers/CATE2004.pdf>
- M. Paasivaara et al., *Teaching students scrum using lego blocks*, in: Companion Proceedings of the 36th International Conference on Software Engineering ICSE Companion 2014, ACM, New York, NY, USA, 2014, pp. 382–391.
- D. Parsons, *Creating game-like activities in agile software engineering education*. In Proceedings of the Australasian Software Engineering Conference. 2014.
- M. Pedaste et al., *Phases of inquiry-based learning: Definitions and the inquiry cycle*, *Educ. Res.* **14** (2015), 47–61.
- M. Pedaste and T. Sarapuu, *Developing an effective support system for inquiry learning in a web-based environment*, *J. Comput. Assist. Learn.* **22** (2006), 47–62.
- V. R. Perry and C. P. Richardson, *The new mexico tech master of science teaching program: an exemplary model of inquiry-based learning*, in: 31st Annual Frontiers in Education Conference Impact on Engineering and Science Education. Conference Proceedings (Cat. No.01CH37193), **1** 2001, pp. T3E–1–4. <https://doi.org/10.1109/FIE.2001.963917>
- D. E. Perry, A. A. Porter, and L. G. Votta, *Empirical studies of software engineering: A roadmap*, in: Proceedings of the Conference on The Future of Software Engineering ICSE '00, ACM, New York, NY, USA, 2000, pp. 345–355. <https://doi.org/10.1145/336512.336586>
- S. K. Rosenkranz, S. Wang, and W. Hu, *Motivating medical students to do research: A mixed methods study using self-determination theory*, *BMC. Med. Educ.* **15** (2015), 95. <https://bmcmededuc.biomedcentral.com/articles/10.1186/s12909-015-0379-1>
- M. Shaw, *Software engineering education: A roadmap*. in: Proceedings of the Conference on The Future of Software Engineering ICSE '00, ACM, New York, NY, USA, 2000, pp. 371–380. <https://doi.org/10.1145/336512.336592>
- E. A. Skinner and J. R. Pitzer, *Developmental Dynamics of Student Engagement, Coping, and Everyday Resilience*, Springer US, Boston MA, 2012, 21–44.

35. G. W. Wei, *The use of wiki to facilitate critical thinking*, in: Proceedings of IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), 2012, pp. H3C-1–H3C-3. <https://doi.org/10.1109/TALE.2012.6360348>
36. J. W. Wu, J. C. R. Tseng, and W. N. Tsai, *A knowledge exploration assistant system for inquiry-based learning*, in: 2013 Second International Conference on E-Learning and E-Technologies in Education (ICEEE), 2013, 19–24. <https://doi.org/10.1109/ICeLeTE.2013.6644341>



**M. NOWOSTAWSKI**, associate professor, has an extensive background in Applied Computer Science, specifically, in distributed and high-performance computing, cloud and mobile technologies and multi-agent systems. He has graduated with PhD in 2008 in the area of self-evolving virtual machines, and compu-

tationally modelling the processes of life. Beside academia, he has work commercially on high-end network appliances with Sun Microsystems and Oracle, programmed high-end GPUs as well as the modern mobile devices. Recently, Mariusz's work focuses on distributed, decentralized and mobile-only systems in the context of blockchain and distributed ledger technologies. Mariusz has contributed hand-on programming expertise and software architecture skills for large complex software systems. In the curriculum development Mariusz was responsible for programming, mobile, and cloud-specific coursework, and currently leads and manages the BSc in Programming study program.



**S. MCCALLUM**, associate professor, has been lecturing in Computer Science for 20 years, and game development for 14. His PhD is in Computer Science from the University of Otago, on the area of “Catastrophic Forgetting in Neural Networks.” He taught in every course in the undergraduate CS degree at the

University of Otago between 1997 and 2007 and teaches

courses across the 5 years of undergraduate and Masters at NTNU in Gjøvik. He has also worked commercially in game development in Norway for 2 years. His research interests include: Serious Games for Education and Health; Virtual, Augmented, and Mixed Reality; and Computer Science Education. He has worked in many funded projects from local, national and European Union level, including H2020 projects.



**D. MISHRA** is an associate professor in the Department of Computer Science at the Norwegian University of Science and Technology (NTNU) since 2016. She has an extensive international experience and earlier worked at Monash University Malaysia, Atılım University Turkey and various

institutions at India. She received her PhD in Computer Science with thesis in the field of Software Engineering. Her research interests include Software Quality, Software Process, Agile Methods, Requirement Engineering, Software Measurement and Metrics, Communication, Coordination and Collaboration Issues in Software Development, Software Testing, Software Engineering Education and Information Systems. She has published over 35 research papers in Journals with around 40 research papers and book chapters in conferences at international and national levels. She is also an editorial board member and reviewer of many reputed journals and guest editor of various special issues.

**How to cite this article:** Nowostawski M, McCallum S, Mishra D. Gamifying research in software engineering. *Comput Appl Eng Educ.* 2018;26: 1641–1652. <https://doi.org/10.1002/cae.21994>