2018 International Symposium on Educational Technology

A Gamification Technique for Motivating Students to Learn Code Readability in Software Engineering

Qing Mi, Jacky Keung*, Xiupei Mei, Yan Xiao, W. K. Chan
Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong
Email: {Qing.Mi, xpmei2-c, yanxiao6-c}@my.cityu.edu.hk, {Jacky.Keung, wkchan}@cityu.edu.hk

***Abstract*—Code readability is one of the important software quality attributes that computer science students need to learn in their programming classes, unfortunately most of the students do not have the necessary work experience or background to appreciate the importance of code readability. Traditional methods of learning code readability tend to be less than interactive and practical in the classroom environment. With the advent of gamification technique, this study introduced a new interactive teaching method and implemented as GamiCRS, an online platform for students to learn code readability. The focus was on incorporating game-based mechanisms to enable students with positive attitudes towards a more interesting learning process. A complete incentive and reward model is proposed in the study together with a combination of both intrinsic and extrinsic motivators identified. To ensure its dynamic efficacy, a field experiment was carried out to compare GamiCRS with its non-gamified counterparts and to evaluate learning outcomes. The empirical results show a positive effect towards the application of GamiCRS in the classroom environment. As many learning activities in software engineering are typically challenging and seldom amusing, gamification can thus be applied as a compelling addition to supporting a wider variety of teaching tactics.**

***Index Terms*—Code Readability; Gamification; Education; Crowdsourcing; Motivation; Technology Acceptance Model**

I. Introduction

Software development is a complex activity that requires a group of developers working together. When collaborating, it is necessary for each developer to maintain a high level of code readability, in an attempt to minimize the time and effort others have to spend reading and understanding it.

Code readability is of central concern to developers [1]. However, there is a lack of effective ways to educate novices on the importance of code readability. The problem inspires this research into building an engaging, interactive environment for novices (especially computer science students) to learn code readability.

Gamification is defined as the application of typical game design elements (GDEs) into non-game contexts [6]. When employed properly, it can be a very powerful tool to improve engagement [9]. With the approach of bringing game-based mechanisms into software engineering (SE) education, we propose GamiCRS, a novel gamification system that aims to build student understanding of code readability. In GamiCRS, we attempt the combination of both intrinsic and extrinsic motivators to form a systemic incentive mechanism. We begin by inspiring students' extrinsic motivation through various GDEs. At this stage, all students' endeavors and accomplishments are meticulously tracked and rewarded with points and badges. While keeping them in the reward loop, we gradually facilitate students' understanding of the importance of code readability, and ultimately internalize its regulation so they can be self-motivated and self-directed to perform the learning activity, instead of trying to escape it or just get through it.

To validate the potential of GamiCRS, a field experiment is designed and conducted. The results show that GamiCRS is effective in motivating and retaining students to learn code readability, which is supported by a sound theoretical and experimental evidence. This study can serve as a significant reference for future development of gamification applications in the context of SE education.

II. GamiCRS: A Gamification System for Code Readability Study

This section first details the design process of GamiCRS based on a psychological theory. Next, a general overview of GamiCRS is provided.

A. Intrinsic and Extrinsic Motivation

Motivation is the process that initiates and energizes goal-oriented behaviors, which is the core element driving individuals to accomplish a task. Motivation varies not only in amount (how much motivation), but also in nature (what type of motivation) [13], which can be divided into two categories:

- **Intrinsic Motivation**: the doing of an activity for its inherent satisfactions (e.g., competition, cooperation, sense of belonging, love, and aggression [12]), which occurs when an individual enjoys the activity or regards it as an opportunity to learn or actualize the potentials [3].
- **Extrinsic Motivation**: the doing of an activity for some separable consequences (e.g., financial incentives, classifications, levels, points, badges, awards, and missions [12]), which can be used to inspire an individual's interest to pursue a behavior.

Research has shown that people's performance varies according to whether his/her motivation stems from intrinsic or extrinsic. Generally, intrinsic motivation offers intense, long-term engagement and results in high-quality behaviors, whereas extrinsic motivation is relatively less effective yet indispensable in the scenario that the activity is in itself neither engaging nor rewarding to the individual, some external rewards are thus introduced as a great way to foster participation,

*Correspondence author.

978-1-5386-7209-9/18/$31.00 ©2018 IEEE
DOI 10.1109/ISET.2018.00062

with an underlying goal of developing people's real interest in the course of being involved in a certain activity [3].

B. Incentive Mechanism

To foster voluntary participation and active engagement, a complete incentive mechanism is proposed with the combination of both intrinsic and extrinsic motivators.

1) Applying GDEs as Extrinsic Motivation: Game design elements (GDEs) act as the toolkit for gamification, which can be used as a great way to motivate certain behaviors. Among all relevant GDEs, the classic PBL triad (Points, Badges, and Leaderboards) is adopted in GamiCRS to reward students through different dimensions. The primary reasons are as follows: 1) PBL is applied in most gamification projects as they are simple (most students have no difficulty in understanding their rules and regulations) yet significantly effective. 2) Although a highly game-like application can produce a better user experience, it requires a lot more time and effort to implement. PBL is employed as a reasonable compromise.

- **Points**: a way of providing quick feedback in GamiCRS. As rewarding students with points for their actions may help to change or boost certain attitudes and behaviors, we equate the core activities in code readability study with points. It should be noticed that not all behaviors are created equal. More points are rewarded for those behaviors requiring more effort and attention. Considering that students may focus on repetition of non-useful gaming behaviors to get plenty of points at one stroke, a ceiling is set on the maximum points that students can be given per day to discourage such meaningless behaviors.
- **Badges**: a visual representation of achievements in GamiCRS. Students are rewarded with badges for reaching specified thresholds, accomplishing particular tasks, and so forth. There are a total of 20 badges available in GamiCRS, which can be further categorized into 4 groups (i.e., basic, standard, premium, and professional). In particular, basic ones take the role of instruction to show what's possible in the system as they are extremely easy to get, while the others are used to provide various goals for students to strive toward.
- **Leaderboards**: a scoreboard displaying the ranking of the leading students in GamiCRS. The presence of a leaderboard can bring students' achievements forward, which is a powerful motivator to trigger friendly competition [11]. However, if the gap between a student and the champion is too great, he/she may feel frustrated and lose motivation towards the activity. One way to prevent this scenario is to simply involve multiple leaderboards across a variety of contexts and time frames. Thus, 5 long-term (i.e., *Total Points*, *Total Ratings*, *Total Uploads*, *Total Badges*, and *Total Login Days*) and 3 short-term (i.e., *Daily Points*, *Daily Ratings*, and *Daily Uploads*) leaderboards are carefully introduced in GamiCRS.

Note that the introduction of GDEs aims to fulfill students' basic psychological needs, but this motivation is directed toward GDEs rather than the task itself. In other words, the intrinsically motivating GDEs are essentially extrinsic motivators with regard to the non-game task [14].

2) Applying Crowdsourcing as Intrinsic Motivation: Extrinsic motivation can strengthen engagement, yet it should not be overused, because excessive external rewards might deteriorate individuals spontaneous interest in the activity [3]. In many cases, intrinsic motivation should be encouraged.

According to a pilot survey in our class, the majority of students would like to receive judgments from experienced colleagues or trusted peers on whether their source code is readable. While we always strive to activate students' intrinsic motivation, we facilitate this by incorporating a crowdsourcing approach into GamiCRS.

As shown in Table I, Doan et al. [8] identified four basic roles of human users in a crowdsourcing system. Based on Table I, a novel dual role setting is proposed in GamiCRS. Specifically, not only can students evaluate readability level of code snippets as *Perspective Providers*, they are also granted the capabilities of uploading their own code snippets as *Content Providers* and then receiving a crowdsourcing feedback from a multitude of human experts, which makes GamiCRS novel in this regard. Although the dual role setting limits the possible application scope of GamiCRS, it allows the activation of individuals' intrinsic motivation.

TABLE I
FOUR BASIC ROLES OF USERS IN A CROWDSOURCING SYSTEM [8]

Role	Description
Slaves	Humans help solve the problem in a divide-and-conquer fashion, to minimize the resources of the owners.
Perspective Providers	Humans contribute different perspectives, which when combined often produce a better solution.
Content Providers	Humans contribute self-generated content.
Component Providers	Humans serve as components in the target artifact, such as a social network.

C. General Overview

GamiCRS is implemented as a web-based application that can be accessed anywhere, anytime for the convenience of students. Corresponding to the dual role setting, GamiCRS consists of two main functions: rating code readability and uploading code snippets.

Immediately after logging in, students are brought to the main page that provides necessary guidance as to what is possible within the system. Students can then navigate to other parts of GamiCRS, or they can begin to rate the readability level of a certain code snippet. Figure 1 provides a general overview of the readability rating page, which is composed of four major modules:

- **User Profile** (the top left portion of Figure 1): the basic information of a student, for instance, the current points and the acquired badges.
- **Points Leaderboard** (the bottom left portion of Figure 1): a vertical structure displaying the ranking and the current points of the top 10 leading students.

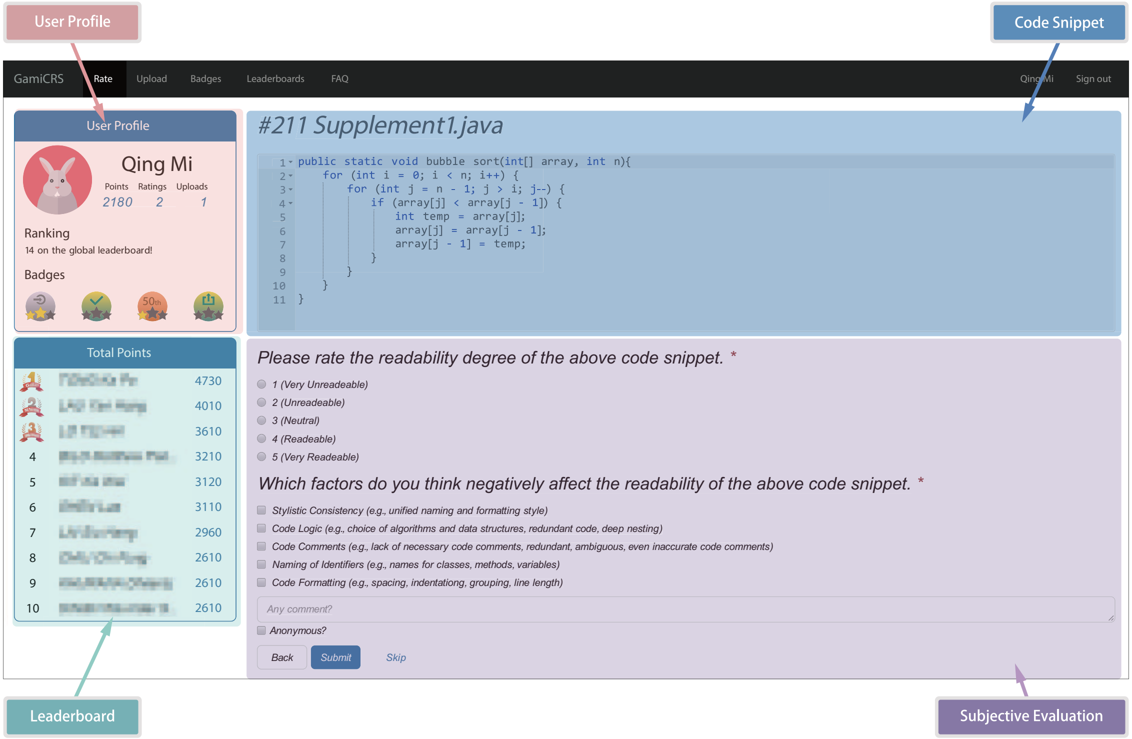

Fig. 1. Readability rating page in GamiCRS displaying the user profile, the points leaderboard, a certain code snippet and a five-point Likert scale.

- **Code Snippet** (the top right portion of Figure 1): a piece of source code (rendered with syntax highlighting) subject to evaluation.
- **Subjective Evaluation** (the bottom right portion of Figure 1): a five-point Likert scale which measures students' perceptions on how easily the code snippet can be read and understood. In addition, students are allowed to provide feedback in the form of comments (optional).

To foster intrinsic motivation and sustain engagement in the long run, GamiCRS supports and encourages students to upload their own code snippets. After submitting a piece of source code, students may well receive a crowdsourcing feedback on its readability level.

III. Experimental Setup

It is assumed that the incorporation of gamification techniques and infrastructure into the context of code readability study can produce psychological (e.g., satisfaction and fun) and behavioral (e.g., participation and performance) benefits for students. To scientifically test this hypothesis, a field experiment is conducted in a real-life environment to compare GamiCRS with its non-gamified counterparts, which lasts over a period of 2 weeks involving a class of 161 undergraduate students from the City University of Hong Kong.

A. Participants

The experiment is performed in a large undergraduate course *CS3342 Software Design*, held by the City University of Hong Kong, which aims to introduce the fundamental principles and practice of software process and object-oriented software development methodologies. The course lecturer is the second author. The primary reasons for choosing CS3342 in this research are as follows: 1) A total of 161 students has enrolled in the course. The class size is large enough to well support our experiment. 2) Since having familiarity with Java programming language (or equivalent) is the prerequisite for enrolling in CS3342, it is likely that all students have some programming experience, which is necessary for our experiment.

To avoid carryover effects, and lower the chances of participants suffering fatigue or boredom from a long experimental duration, we carefully plan a between-subjects design [2]. All participants are randomly and evenly allocated to the treatment and control group. After successfully logging into GamiCRS with the pre-allocated *Username* and *Password*, participants are automatically directed to different interfaces according to the group they belong to. The treatment group (N = 81) has various GDEs displaying in the user interface to encourage their participation, while the control group (N = 80) does not. All other respects are treated identically to eliminate the impact of environmental variables.

B. Procedures

The experiment spans a period of 14 days from 13th February to 27th February. To reflect the natural behaviors and avoid creating time pressure, students are permitted to access GamiCRS at their leisure. The specific process of the experiment is described as follows:

- **13th February**: Students are told that they are invited to attend a code readability study and they are free to withdraw from the experiment at any time they want.
- **20th February**: Students are reminded that the experiment is still in progress, and their support and cooperation would be greatly appreciated.
- **27th February**: A post-experiment questionnaire is given to students to collect their perspectives on GamiCRS as well as demographic information (i.e., age, gender, and years of Java development experience).

Note that the experimental procedures are introduced to students, but not the true objective, this is to protect our experiment from the observer or Hawthorne effect [10].

C. Post-Experiment Questionnaire

After the field experiment, a questionnaire approach is utilized in the treatment group to investigate students' impressions and attitudes toward GamiCRS based on Technology Acceptance Model (TAM) [5], one of the most robust theories for predicting and/or explaining user acceptance. Specifically, our post-experiment questionnaire is structured with four TAM-based Likert-subscales modified to the context of this study.

- **Perceived Ease of Use (PEOU)**: the extent to which students believe that using GamiCRS is easy (i.e., free of physical and mental effort).
- **Perceived Usefulness (PU)**: the extent to which students believe that using GamiCRS can enhance their performances and help achieve their goals.
- **Attitude Toward Game Design Elements (ATGDE)**: an overall assessment representing students' attitudes toward the effects of game design elements.
- **User Satisfaction (US)**: the extent to which students consider GamiCRS meets their needs and expectations.

The objective of the post-experiment questionnaire is basically to understand students' perceptions and behaviors on GamiCRS. To fully capture the concept being assessed, each subscale is composed of a series of Likert-type items that represent similar statements, which is enumerated in Figure 2. Students are required to specify their level of agreement with each statement on a scale ranging from 1 (Strongly Disagree) to 5 (Strongly Agree).

IV. Results and Discussion

At the end of the field experiment, the treatment group produces slightly more ratings as compared to the control group. The number is 385 versus 355.

A total of 73 valid responses is received through both online and written questionnaires, giving a response rate of around 90%. We begin by summarizing the demographic characteristics of the participants. The population comprises 54 (74%) males and 19 (26%) females. Approximately 89% of the participants report to have Java development experience (Median = 1, Mean = 1.56, SD = 1.44). As all participants are undergraduate students, their age varies in a small range from 18 to 26 years old (Median = 20, Mean = 20.59, SD = 1.49).

Next, Cronbach's alpha (a measure of internal consistency or reliability) [4] is used to provide evidence that the Likert-type items form an internally reliable subscale (i.e., the Likert-type items are homogeneous and measuring the same concept). The value of Cronbach's Alpha for each of the subscale is shown in Table II. Among a wide variety of standards proposed to interpret what makes a *good* alpha coefficient, we follow the rules of thumb as given in DeVellis et al.'s work [7], which is considered *Excellent* for >0.90, *Very Good* for >0.80, *Respectable* for >0.70, *Minimally Acceptable* for >0.65, *Undesirable* for >0.60, and *Unacceptable* for <0.60. It can be observed that all of our results are far above the threshold 0.6, demonstrating a reliable item-scale correlation coefficient.

TABLE II
Internal Consistency of the Post-Experiment Questionnaire

Subscale	Cronbach's Alpha	95% Confidence Boundaries		Mean	SD
		Lower	Upper		
PEOU	0.75	0.66	0.85	3.3	0.59
PU	0.85	0.79	0.91	3.5	0.71
ATGDE	0.90	0.87	0.94	3.2	0.73
US	0.86	0.80	0.91	3.1	0.78

After that, the response distribution of the post-experiment questionnaire is provided in Figure 2. The stacked bar chart is the most widely used approach for visualizing Likert results, with a position further to right indicating a higher level of agreement on the particular statement.

Since students' needs and attitudes should be put at the center, we value their feedback from the post-experiment questionnaire as an important measurement. As shown in Figure 2, when asked if they find using GamiCRS is easy and useful, most of the students express a positive stance, which makes it possible to promote long-term behavioral changes with the help of GamiCRS. Then the impact of isolated GDEs is identified to help researchers and practitioners to prioritize which GDEs to use in their future work. It is noted that points and leaderboards are more effective than badges in motivating students to participate. Although the PBL triad plays an active role, students put more emphases on the usefulness of the gamified system, which can serve to guide future working directions. Also, students express a slightly more neutral response concerning the US aspect, indicating that GamiCRS is not as attractive as what they expect. To improve GamiCRS, it would be valuable to conduct trials with different mechanisms, such as the inclusion of challenges (e.g., a non-deterministic rewarding mechanism) and/or explicit collaborations.

V. Conclusion and Future Work

This paper presents the design, implementation, and evaluation of GamiCRS, which is a novel system that incorporates gamification techniques and infrastructure into the context of code readability study. Specifically, the study provides the following contributions:

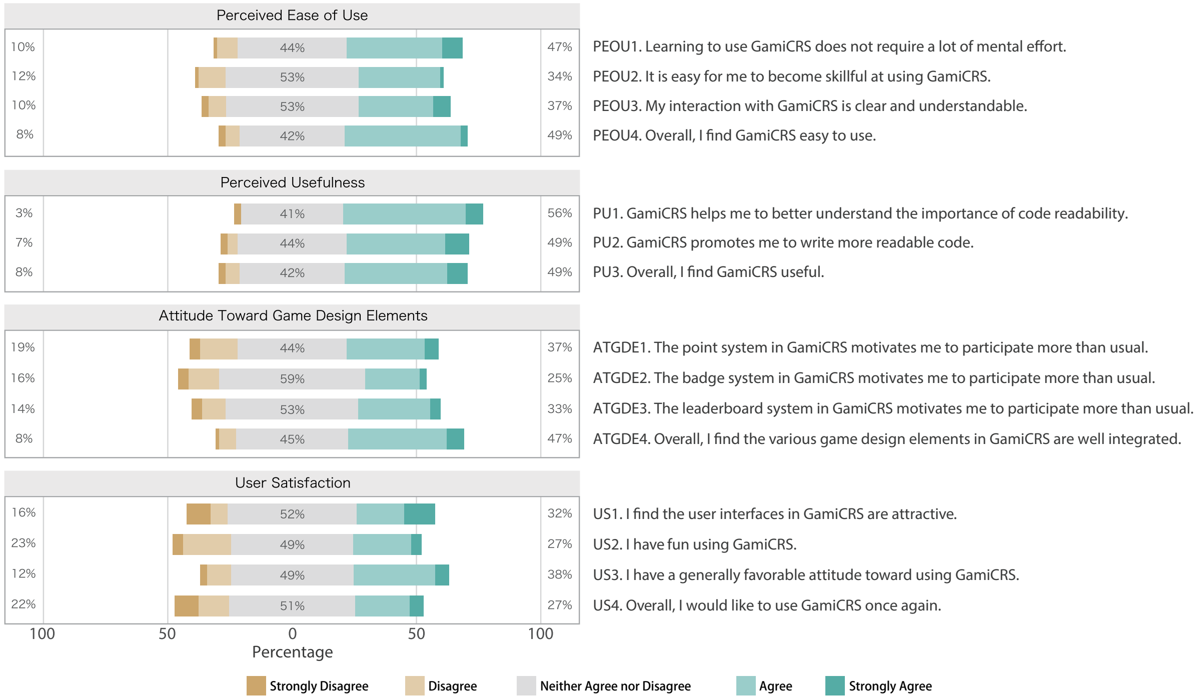

Fig. 2. Responses to the post-experiment questionnaire.

- A reusable gamification system is proposed for code readability study (GamiCRS), in which a novel incentive mechanism is presented to help foster active engagement.
- GamiCRS is implemented as a web-based application. As compared to prior work, this paper contributes a more detailed implementation process in a systematic manner.
- A large-scale field experiment is performed to validate the practical feasibility of GamiCRS and to help delineate future working directions.
- Based on the Technology Acceptance Model, a questionnaire approach is utilized to investigate students' impressions and attitudes toward the acceptance of GamiCRS. The results show that GamiCRS is effective in motivating and retaining students to learn code readability.

We hope that our study can interest and encourage further research regarding gamification, which is a promising methodology for changing behavioral patterns and for supporting a wider field of possible applications in SE education. As future work, we will investigate to what extent students' gender or programming experience affects their attitudes toward GamiCRS. Additionally, we will replicate our experiment with longer code samples. This is because the main challenge in program comprehension is in reading lengthy codes.

Acknowledgments

This work is supported in part by the research funds of City University of Hong Kong (project nos. 9042499, 6000485, and 9678149) and GRF of Research Grants Council of Hong Kong (project nos. 11208017, 11201114, 11200015, and 11214116).

References

[1] R. P. L. Buse and W. R. Weimer. Learning a Metric for Code Readability. *IEEE Transactions on Software Engineering*, 36(4):546–558, jul 2010.
[2] G. Charness, U. Gneezy, and M. A. Kuhn. Experimental methods: Between-subject and within-subject design. *Journal of Economic Behavior & Organization*, 81(1):1–8, jan 2012.
[3] D. Coon and J. Mitterer. *Introduction to psychology: Gateways to mind and behavior with concept maps and reviews*. Cengage Learning, 2012.
[4] L. J. Cronbach. Coefficient alpha and the internal structure of tests. *Psychometrika*, 16(3):297–334, sep 1951.
[5] F. D. Davis. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3):319, 1989.
[6] S. Deterding, D. Dixon, R. Khaled, and L. Nacke. From game design elements to gamefulness. In *Proceedings of the 15th International Academic MindTrek Conference on Envisioning Future Media Environments - MindTrek '11*, page 9, New York, New York, USA, 2011. ACM Press.
[7] R. F. DeVellis. *Scale development: Theory and applications*, volume 26. Sage publications, 2016.
[8] A. Doan, R. Ramakrishnan, and A. Y. Halevy. Crowdsourcing systems on the World-Wide Web. *Communications of the ACM*, 54:86, apr 2011.
[9] J. Hamari, J. Koivisto, and H. Sarsa. Does Gamification Work? – A Literature Review of Empirical Studies on Gamification. In *2014 47th Hawaii International Conference on System Sciences*, pages 3025–3034. IEEE, jan 2014.
[10] B. Kantowitz, H. Roediger III, and D. Elmes. *Experimental psychology*. Nelson Education, 2014.
[11] A. Marczewski. *Gamification: a simple introduction*. Andrzej Marczewski, 2013.
[12] C. C. I. Muntean. Raising engagement in e-learning through gamification. *The 6th International Conference on Virtual Learning ICVL 2011*, (1):323–329, 2011.
[13] R. M. Ryan and E. L. Deci. Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions. *Contemporary Educational Psychology*, 25(1):54–67, jan 2000.
[14] D. Wilson, C. Calongne, S. B. Henderson, and B. Henderson. Gamification Challenges and a Case Study in Online Learning. *Internet Learning Internet Learning Internet Learning*, 4(2), 2015.