A Comparative Study of Online vs. Onsite Training for Scrum Using Gamification

Wantana Sisomboon and Nuttaporn Phakdee*

Software Engineering, Faculty of Informatics, Burapha University, Conburi, 20131, Thailand

*Corresponding author E-mail: nuttaporn@go.buu.ac.th

ABSTRACT

Since the World Health Organization announced the 2019 coronavirus disease (COVID-19) in 2020, 77 countries have reported cases of COVID-19. Various organization have demonstrated the ability to reduce the COVID-19 virus transmission. In response, many software industries have transformed to having people work online. It was a big adaptation for the workers. Scrum is an agile project management framework which is widely used in many organizations. Use of online Scrum tools is increasing rapidly nowadays. For education, the Scrum approach was adopted for Software Engineering students to learn the current way of developing software projects. This research aims to compare the effectiveness of learning to develop software utilizing a Scrum framework by comparing between online and onsite groups of students. Gamification, which is the application of game elements in a non-game context, was adopted to enable this learning. Game components are used to increase team motivation and change behavior. Six research questions were produced to compare the difference between learning Scrum utilizing gamification online and onsite. The first four questions compare the differences between online and onsite students' opinions toward using the agile framework, based on four key values of the agile manifesto. The online group responded to change better than the onsite group, whereas there was no significant difference in the other three values. The fifth question examined negative Scrum activities. The result showed that the Velocity chart was the most difficult technique to understand. The fun game aspects were evaluated in the last question. The results show that for both online and onsite participants, the "Leader board" was the students' favorite fun game element.

Keywords: *Scrum, gamification, software engineer, COVID -19, online learning*

INTRODUCTION

The world is facing a particularly difficult situation as a result of the COVID -19 virus's global spreading. (Harish, 2020). A report from the UNESCO website shows that more than 1.2 billion children in 186 countries are being affected by school closures as a result of the crisis. Most of the curriculum has been converted to an online format, with unpredictable long-term impacts. In addition, numerous worldwide organizations have adjusted their working environments as a result of the COVID -19 situation. Even though the software industry's work environment began to be digitized before COVID-19, it was a big adaptation for the workers. Furthermore, the Deputy Prime Minister and the Minister of Public Health announced

Article history:
Received 25 February 2022; Received in revised from 20 May 2022;
Accepted 7 June 2022; Available online 25 June 2022.

the number of patients under investigation on January 13, 2020. Since that time, businesses in Thailand have allowed people to work from home and the education sector has changed to having learners study utilizing online tools.

Due to the current Coronavirus pandemic, Thailand's higher education institutions have decided to transition from traditional teaching to online learning. Nevertheless, many issues have arisen from students, instructors, the colleges, and the students' parents, as well as other considerations (Donitsa-Schmidt, 2020).

Gamification is a new strategy for engaging users and solving problems by applying game elements to non-game contexts. In recent years, 70% of the Global 2000 companies have implemented game-like solutions to encourage consumer engagement and staff performance. The practice known as gamification is becoming increasingly popular across all industries. (Hermanto, 2018)

Scrum is an agile method which is widely used in software organizations because Scrum is a simple, uncomplicated, and easy to implement methodology. With repetitive iterations and continuous feedback, it assists Scrum teams in delivering end products that meet customer requirements. Software Engineering students at the Burapha University begin learning software development following a Scrum framework in a course named "Opensource Software Development Camp" (OSSD). The camp introduces second year students to a challenging and fun-packed learning experience of software development utilizing open source tools for 7 days. The researchers have applied gamification. In Scrum software development, game components are used to increase team motivation and change behavior.

Various software development tools exist in the market. This research designed a custom framework to support teaching undergraduate students to do software development utilizing Scrum. This is call "Scrumification". The aim of this development is to make students have fun engaging with the four values of the Agile manifesto. Six research questions were raised to compare the significant differences toward development of software utilizing gamification between online and onsite environments. The first four questions were designed to evaluate behavior changes following four key values of the Agile manifesto. The fifth question examined negative Scrum activities. The fun game aspects were evaluated in the last question. The paper is organized as follows: Section 2 summarizes literature evidence about Gamification, the Agile framework, the Scrum software development approach, and user centered design. Section 3 discusses the research methodology. Section 4 shows an experiment that compares simulated gamification in Scrum context between online and onsite environments with a case study. Section 5 gives a review of the results and discusses the experiment. Finally, section 6 wraps up this project and makes some recommendations for future work.

LITERATURE REVIEW

The purpose of this section is to provide a review of past research about Gamification, Agile frameworks, Scrum software development approaches, and user centered design.

Gamification

According to Soesanto et al. (2021), gamification is a technique for introducing game elements into non-game contexts. It is the process of combining game mechanics, aesthetics, and game thinking in order to engage people, motivate action, promote learning, and solve issues. An important factor in successful gamification is the selection of game design elements. Game design elements specify the type of experiences that will be generated for the players. The following typical game elements were found in reviews of gamified participation tools (Sarah, 2016): 1) Achievement: Badges and rewards are common achievement indicators used in game applications. For accomplishing specific tasks, players receive a badge or a reward. The purpose behind badges and rewards is to encourage and motivate learners by recognizing their progress, while also making learning more interesting and fun. 2) Personalization: The example game elements in this group include profiles and avatars. A profile is a description of personality for someone or something. An avatar is a graphical representation of a player character or persona. 3) Points: Points are an abstract score associated with a player or team in games. 4) Time constraint: It refers to a number of factors that limit the amount of time to complete mission. 5) Competition: A leaderboard is an example game element in this category. The objective of a leaderboard is to show players their rank in a gamified system. A leaderboard is a tool for evaluating players against specific success criteria. 6) Conflict or Challenge: There should be some type of conflict in a game to provide a challenge for the player to overcome to make the work more engaging.

Agile

In 2001, Kent Beck and sixteen Agile alliances, which were formed by independent-minded software practitioners, wrote the Agile Manifesto. The Agile Manifesto is a declaration that outlines the key values and practices that support the Agile methodology, with the objective of making development teams be more effective and to allow them to last longer. The Manifesto's creators listed the following key values of developing software. “1) Individuals and interactions over processes and tools, 2) Working software over comprehensive documentation, 3) Customer collaboration over contract negotiation, and 4) Responding to change over following a plan”. While planning is important, the Agile alliances also stress that it is necessary to accept that plans change. In general, the Manifesto emphasizes the importance of people and interactions over processes and tools (Kent Beck, 2001). There are multiple techniques in agile approaches such as eXtreme Programming, Crystal Clear, Agile Unified Process, Feature-Driven Development (FDD), Kanban and Scrum. eXtreme Programming focuses on practice, while Kanban and Scrum focus on managing processes and reducing waste.

Scrum

Among the various Agile approaches, the most successful approach is Scrum (Singh, 2020). Scrum emphasizes all stakeholders' involvement while developing the software. The focus is on delivering working software as soon as possible. Scrum is a method that emphasizes courage, focus, dedication, respect, and transparency as core qualities. In 2020, Ayunda et al. proposed in her research that a valid scrum framework consists of scrum teams, scrum events, scrum artifacts, and artifact transparency. Artifacts in Scrum represent activity or value. The Scrum method consists of four artifacts. These are user story, product backlog, Sprint backlog, and increment. In Scrum, five events are used; Sprint, Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective. A Sprint is a limited period of time in which a Scrum team works to complete a specific amount of work. Sprint planning is a Scrum event that begins the Sprint by outlining the work that will be done throughout the Sprint. Each team member answers the following three questions at the Daily Scrum: 1) What did you do yesterday? 2) What are your plans for today? And 3) Are there any obstacles in your work? The Sprint Review is an opportunity for the Scrum Team to reflect on the tasks done during a sprint and to plan the next steps. A sprint retrospective is a meeting held at the end of a sprint period during which Agile teams analyze the previous sprint and make improvements for the next. A Scrum Team consists of developers, Product Owner and a Scrum Master. The product owner is responsible for producing the schedule. The Scrum Master's role is to ensure the development team completes the sprint and delivers shippable products. The development team is responsible for making the product shippable. (p.318)

Scrum gamification

“Scrum gamification” describes the use of game elements in Scrum software development to increase team engagement and change behavior. Currently there are various tools which support Scrum software development with Gamification in the world market. Some examples are Bitrix24, Trello, ClickUp, Miro, Plan Street, GoodDay, and Status Hero.

In recent publications, discussions occur about adopting gamification in developing software utilizing the Scrum technique. For example, in 2017, Ulrich Schäfer (p.754) proposed a training Scrum with gamification: In this study, he discussed two teaching periods at a university of applied sciences with 110 students. Ulrich and his team compared the two versions and analyzed the results and found that Gamification which utilized the Minecraft technique assisted leaners to focus on the project while successfully learning the mostly new Scrum methodology. In addition, the game motivated leaners and assisted project teams in bringing together students from different backgrounds to form effective teams. The limitation of this study was that it did not support the submission of change requests to the teams.

In addition, in 2017 Sonja Hof proposed using gamification for teaching. The research found that successful agile software development requires collaboration and communication (p. 323). Agile values include respect, flexibility, transparency, and trust. The experiment adopted the Scrum Lego City game, which is a widely used game to teach Scrum in a one-day course. The online survey result showed that "what they liked best about the game aspect," was working in a team. It is obvious that participants prefer to experience the Agile approach directly in a project, since it makes the work more fun, and that they like team communication.

In 2019, Nitin Naik and his team proposed utilizing Trello to implement a Game-Based Learning (GBL) strategy for Agile Scrum. This approach was put into practice in four developmental steps. Trello was used to teach participants about Agile Scrum. Participants were split into groups and designed a strategy for creating a Trello Board for a specific project case study. They competed for rewards by displaying their Trello Boards in a game tournament. They used GBL's overall knowledge and experience in the creation of a real project. The research showed that students can more easily learn the Agile Scrum approach as a game, not as a complex methodology, with this Trello-based gaming activity. Finally, the researchers suggested applying this technique to other courses/subject areas as a topic of future work (p.1).

In 2021, Waraporn Jirapanthong wrote an analysis of literature on game project management based on agile approaches (p. 30). The experiment participation was from 41 students. The research aim was to learn whether or not agile methodology is more suitable for game development management for participants who enrolled in the course of game development project management. The research focused on the planning, management, and execution phases. The result was that no significant difference exists for learning progress between learners who utilized a traditional approach and those who used Scrum. This experiment measured utilizing Scrum and gamification in a project management subject. Observational studies would be required to make it possible to apply this to other courses successfully in the future.

User Centered Design

User-centered design (UCD) is an incremental design method in which designers attach great importance to users and their requirements throughout each stage of the process. The process consists of five fundamental activities. 1) Capture User Requirements: The goal of the research is to gain a better understanding of the product from the user's perspective, as well as to identify users' fundamental requirements and expectations. 2) User Research/Define User: Determine who will use the product, what it will be used for, and under what conditions it will be used. 3) Design UI: This phase can be completed in stages, starting with a rough concept and progressing to a finished design. 4) Prototype the UI Design and, 5) throughout the design process, prototypes must be assessed. The activities are carried out in an iterative way during the development phase. The workflow will not be completed until the system meets the requirement definition. (Bin Xiao et al, 2020).

RESEARCH METHODOLOGY

The Scrumification Framework

In 2020, Oscar Pedreira and his team proposed a software framework, a gamification model, and a gamification engine for implementing gamification in Software Engineering workplaces (p. 776). The gamification engine was produced to support a real software organization work environment. The features consist of a social network for the players, online chat, system notifications, participant challenges, and a virtual assistant. This might easily be used to create engaging software products in a variety of domains. A high-level view of the framework driven with the gamification engine shows it is the most important part of the framework because it receives and assesses all of the software engineers' behaviors. The engine has an integrated REST API that allows it to interface with other tools. This framework was produced to support real software organization work environments, but does not support learning the Scrum framework at the beginner level. Also, it does not support the Thai language need by my undegraded students.

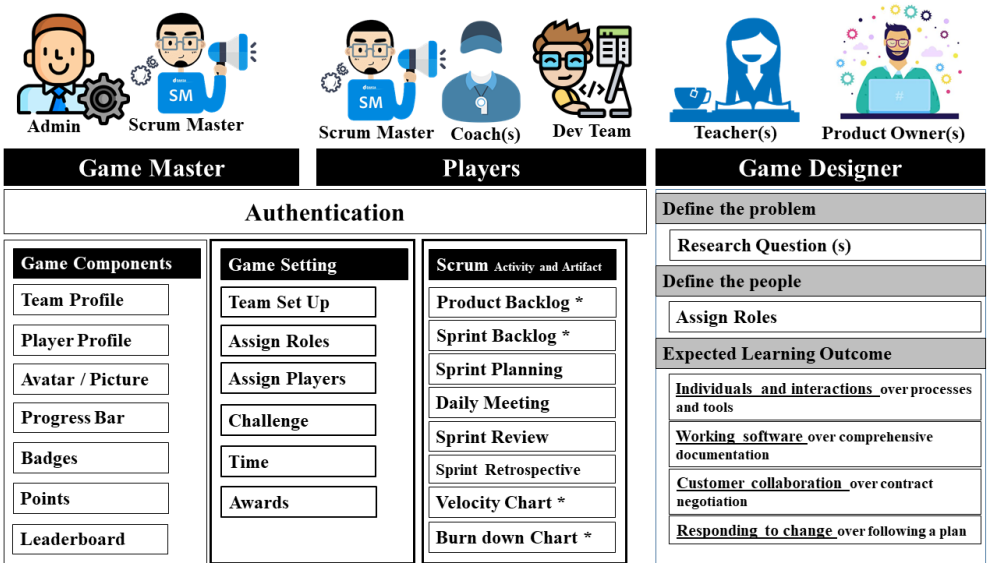

Figure 1 The Scrumification Framework.

Figure 1 shows the Scrumification framework applied to support beginners learning Scrum in an undergraduate course. It begins with a game designer group which consists of teacher(s) and product owner(s). They are responsible for creating games for serious situations. They consider how to make difficult learning be fun during the learning process. The scope of design consists of defining the problem, defining the people, and specifying the expected learning outcome. The details are presented next.

Designing the problem: To design the problem, the game designer created six research questions as follows:

Research question 1: Are there significant differences in the individuals and interactions in processes and tools between online and onsite players in learning Scrum utilizing Scrumification?

Research question 2: Are there significant differences in the working software writing comprehensive documentation between online and onsite players in learning Scrum utilizing Scrumification?

Research question 3: Are there significant differences in the customer collaboration for contract negotiation between online and onsite players in learning Scrum utilizing Scrumification?

Research question 4: Are there significant differences in responding to changes following a plan between online and onsite players in learning Scrum utilizing Scrumification?

Research question 5: What are the negative game elements that are difficult to understand via online learning?

Research question 6: What are the positive game elements that help make learning Scrum fun?

Define the people: The study divided actors for the Scrumification into two groups called “Game Master” and “Players”. The Game Master team has two people who are the “Admin” and “Scrum Master”. They work together to set up an environment for learning Scrum via Scrumification. The Scrum master is responsible for creating “Challenges” and selecting “Awards” for the learners. The game fun comes from the “Game Components” and the design of the challenge activities. The knowledge was created from “Scrum elements” as shown in Figure 2. Players were allowed to login and use the Scrumification to play games. Coaches are allowed to update status of “Scrum activities” such as Sprint planning, Daily Scrum, Sprint Review, and Sprint Retrospective (Activity the coach can update are represent with * mark) after a Scrum team completes its achievements. Scrum teams are allowed to see their own profile, the team profile, work progress from a progress bar, and the leaderboard. Team leaders are allowed to exchange “Points” for special items to support their work, such as expanded working time, or hiring a specialist to consult with for difficult coding.

Expected Learning Outcome: The main purpose of this experiment was to evaluate the "Game players" to see if they embraced the four key values of the Manifesto for Agile Software Development. The Agile Principles are a set of ideals that guide how individuals in the company interact with one another. These values and principles are critical to understanding agile project management correctly. The adaptation of the four values with 12 Principles was an expected learning outcome for this study (Kent Beck, 2001). The values are:

1) Individuals and interactions over processes and tools. Throughout the project, business people and development teams collaborated on a regular schedule. Face-to-face communication was the most efficient and effective way of transmitting information within a development team. Projects should be built around people who are passionate about their work. Space and support required for project members should be given, and project members are trusted to do the task. The development process should be driven by teams so they will be responsive to change to meet customer requirements.

2) Working software over comprehensive documentation. Teams should prioritize working software over comprehensive documentation. The Scrum framework focuses on delivering valuable software on time. Working software is the primary measure of progress.

3) Customer collaboration over contract negotiation. This process includes an end-user on the team in daily meetings to ensure the products meet the customer's business needs.

4) Responding to change in following a plan. Changes in requirements are acceptable, especially if they occur late in the development process. Agile processes take advantage of change to help customers gain a competitive advantage. The shorter timeframe should allow for the addition, deletion, or update of features in the next iteration.

The Mixed UAT-Scrum Development Process

Development was divided into two phases. The first phase focused on design. The second phase was gamification tool development and testing. User-centered design (UCD) was adopted as a method of integrating user research and feedback into the design process. Figure 2 shows the 5 Stages of UCD.

Capture User Requirements: After analysis of relevant business and user problems, the framework was designed as shown in Figure 2. The aims of development were to 1) support all users in Scrum roles to complete software utilizing a Scrum framework, and 2) create game components and a game setting.

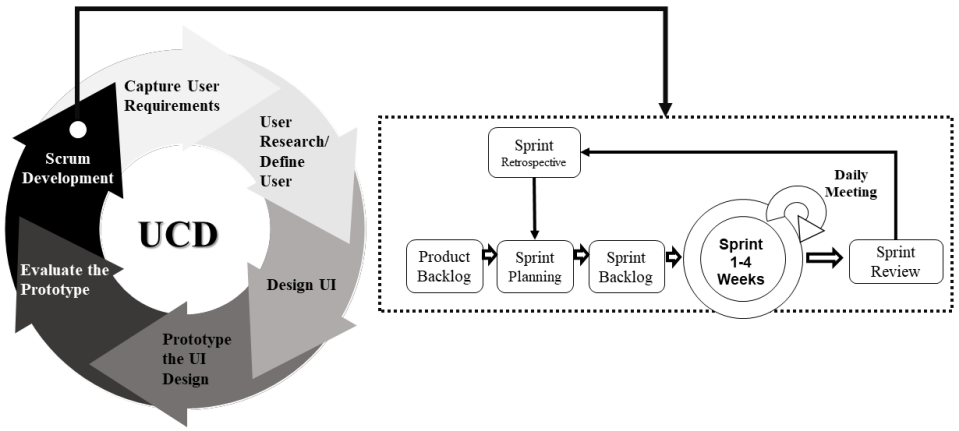

Figure 2 The Mixed UAT-Scrum Development Process.

User Research/Define User: First, User personas represent typical users whose aims and characteristics represent the expectations of a larger group of users. Wanda, between 18-21 years old university student, represented the player with no-experience as a programmer and without any knowledge of the Scrum framework. The requirements were selected from Wanda's needs. However, at the end of stage 3, more users were defined, Admin and Scrum Master were assigned to set up an environment for fun games involving Scrum learning. A Scrum master is also authorized to design the challenges and rewards in the game. Coach(s) are responsible for reporting the achievements representing solving each challenge in each team.

Design UI: The requirements were written following the standard of the ISERL laboratory. Then the requirements were transformed to UI design. A tester reviewed the UI design to ensure that it followed the ISERL laboratory standards and validated it based on the requirements. Figure 3 is an example of the results from UI design.

Prototype the UI Design: A prototype was created following the UI design. The Prototype consists of 8 modules: 1) Player Management, 2) Scrum Activities Management, 3) Points Management, 4) Shopping Mall Management, 5) History of Activities, 6) Group Management, 7) Rewards Management, and 8) Kanban Board.

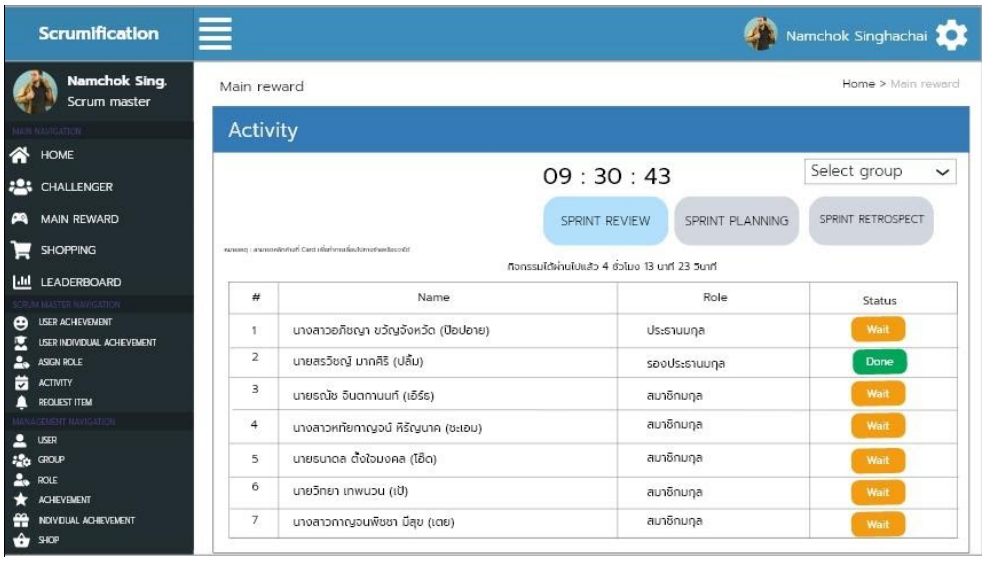

Figure 3 The Scrumification Development Process.

Evaluate the Prototype: An evaluation was conducted by five specialists and twenty-five second year university students to measure user satisfaction. In the first stage, the prototype was demonstrated to five specialists from three companies. One specialist had received a Scrum certificate and all of them had more than 10 years of experience with Scrum frameworks. In addition, 2 specialists had experience with gamification. Feedback from specialists was collected after demonstration and redesign. In the second phase, the prototype version 1.0 was released and demonstrated to 25 university students. The age of the students averaged 20 years old. They had one-year of experience with Scrum frameworks and gamification. The questionnaire consisted of 20 questions which covered topics of content (3.97), UX/UI design (4.04), flexibility (4.04), and benefits (4.06). A 1-5 rating scale (1 - Very Dissatisfied to 5 - Very Satisfied) was utilized to measure user satisfaction. The results of the evaluation show an average score equal to 4.03 which is considered positive.

Scrum Software Development: Development of the Scrumification Web Application adopted the Scrum technique in the development and testing process. The development took 4 sprints (4 weeks per sprint). The web application consists of the same 8 modules which were in the prototype. Figure 4. shows an example of the leaderboard feature in the Scrumification website application.

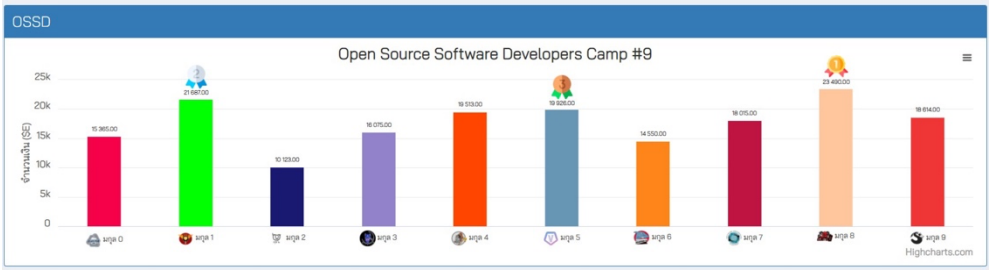

Figure 4 Leaderboard: An example of a feature in the Scrumification web application.

After the development was completed, the system was tested. 50 test cases were designed for automated tests. 41 test cases passed without defects and 9 failed. Those 9 test cases were for the Kanban Board module which was not completed due to the high complexity of code and a lack of time.

CASE STUDY

This section outlines a controlled experiment which involved two groups of Scrum learners. One was assigned online study, while the other practiced in a real, onsite environment.

Participants

Onsite group: The Scrum masters were two third-year students and two teaching professors. The Scrum teams were formed by dividing the 75 second-year students into ten groups. Each group consisted of six to nine students chosen randomly. The Coach(s) were twenty experience programmers.

Online group: The Scrum masters were two third-year students and two teaching professors. The Scrum teams were formed by dividing the 86 second- year students into ten groups. Each group consisted of eight to nine students. chosen randomly. The Coach(s) were five project managers from five software development companies.

The experiment environment

Working start with a seven days' software development camping for second year students called Opensource Software Development (OSSD) camp. When learners started working on real projects, they utilized a Scrum workflow and the Scrumification web application to manage their project development. The development was divided into 6 sprints. The duration of each sprint was one day as show in Figure 5.

Tools

Onsite group tools: A physical Kanban board was utilized for monitoring work progress. 10 groups sat at a round table while coding their product. During the Daily Scrum event, all team members stood in a circle and answered the three questions one by one. The outcome in each sprint was presented in a face-to-face

meeting with the product owner. A Sprint Retrospective meeting was held at a round table.

Online group tools: The online groups used “Zoom” for contact with product owners and for formal meetings such as “Sprint Review”, “Sprint Retrospective”. “Discord” was used for daily scrum meetings, monitoring team member work progress, and consultants. The online Kanban board called “Trello” was adopted for monitoring team and individual work progress.

Online and Onsite Work Processes

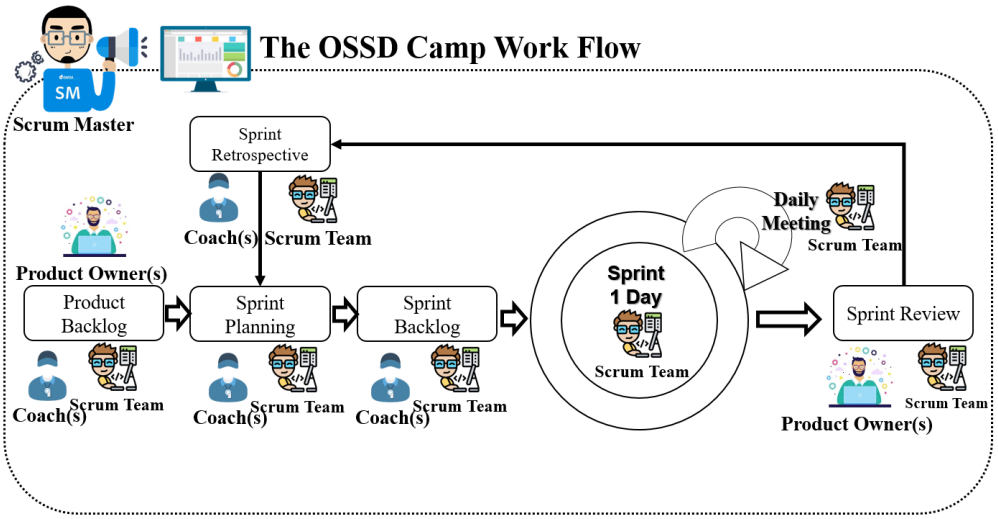

Figure 5 The Online and Onsite Working Environment.

Scrum workflow: Each day, the Scrum team kicked off the sprint with sprint planning. A coach and his/her scrum team selected tasks which specified what could be delivered within the sprint then prioritized them and placed them in order on a Sprint backlog board. The onsite group placed their tasks on a Kanban board so everyone in the team could see what was happening easily. The board was divided into three parts (according to work status) which were “TODO”, “DOING”, and “DONE”. A task was written on a sticky note, then the Scrum team member could move tasks to the different area to show team work progress obviously on the board. The online group used “Trello” which is a collaboration tool that organizes team software development projects into boards.

For both groups, before the code development started, Scrum teams stood and conduced their Daily Scrum within 15 minutes. Everyone answered these three questions: 1) What did you do yesterday? 2) What will you do today?, and 3) Are there any obstacles that the team need overcome? At the end of the day, product outcome was demonstrated and signed for acceptance by the Product Owners who were the real project owners. The teams ended their sprints with a sprint retrospective,

which is a periodic meeting held at the end of a sprint cycle to address what went well and what can be improved for the next sprint cycle.

Gamification: For Scrum events, which included Sprint Planning, Sprint Backlog, Sprint Review, Sprint Retrospective, and Daily Scrum, teams earned 10,000 Points after they completed their achievement in each sprint. A coach updated team status in the system if there finished within a time limit. After the session of sprint review, each team received 10,000 points for a deliverable user story and then the Scrum master collected information and updated team rewards in the system. The leaderboard showed the total points of all teams and the top three leaders were marked on the board. The Scrum master analyzed "Burn down charts" of each team and decided to provide awards of "Top Task Killer" to the winner. Awards of "The Best Quality Product" and "The best Teamwork" were considered from various aspects by stakeholders. Various challenges were produced to improve team participation, work collaboration and to make work more fun, such as requirement changes, limiting working time, cutting budgets, etc. In addition, points could be exchanged to enable important activities to support their work, such as hiring a specialist, expanding working time, and buying items to decorate the project presentation booth for the onsite group.

RESULTS AND DISCUSTION

The results of the data analysis are presented in this section, together with a discussion of the findings with reference to the aim of the study.

Data collection

Data was collected for statistical analysis from feedback on specific aspects of learning experiences of the two sample groups. An online questionnaire was a primary step for collecting the data from 75 second-year students for the onsite group registered in education year 2018 and the 86 second-year students in the online group from education year 2020. The survey was distributed at the end of the course. Altogether 62 students from the onsite and 74 from the online sample groups completed the survey. Students were requested to provide qualitative feedback on their learning experiences. The questionnaire was divided into three parts, each of which contains multiple questions. Part 1: Information about the participants. Part 2: Learner satisfaction with scrum activities and game elements. A 5-Point Likert Scale was adopted with 5-very Satisfied, 4-Satisfied, 3-Neutral, 2-Dissatisfied, and 1-Very Dissatisfied. Finally, Part 3 contained open-ended questions for learner suggestions and comment. In addition, data was collected from product owners' interviews. On the last day of the experiments, the researcher held focus group interviews. The ten product owners for the onsite group were programmers with 10 years of experience in a university research laboratory and 1 person from a software company, while the online group invited five people from software companies who had more than 10-years' experience act as coaches and product owners. Finally, the experiment was observed by two researchers who were Scrum masters in the learning camp.

Result and Discussion

Research question 1: Are there significant differences in the individuals and interactions over processes and tools between online and onsite players in learning Scrum utilizing Scrumification?

A comparison was made focusing on the aspect of individuals and interactions over processes and tools between the online and onsite sample groups utilizing a two-sample equal variance sample t-test with a significance level of.05. It was based on results from the questionnaire survey of 62 students for the onsite experiment and 74 answers from the online experiment. Researchers found average scores for the onsite group = 4.394 and average scores for the online group = 4.530 with t = 1.150 and p value =.252. That means there was no significant difference between these two sample groups. In addition, the result was found from working process observation and group interviews of coaches and product owners. The Scrum events included sprint planning, daily Scrum, and Scrum retrospective which served to encourage team collaboration and communication with each other more often. In addition, game elements called achievements motivated the two groups to communicate and work together regularly.

Research question 2: Are there significant differences while working to write comprehensive documentation between online and onsite players learning Scrum utilizing Scrumification?

Point values in the Sprint velocity chart are show in Figure 6., which measures work that has previously been done and was utilized to look for significant differences in this aspect. The result found that the percentage of completed tasks for the onsite group was slightly higher with average scores of 15.932 compared to the online of 14.360, but t =-1.072 and the p – value = 0.298 which is less than the alpha, 0.05. The result of these two-sample groups was not significantly different. Although in this experiment two sample groups were assigned to create and maintain a massive of documents such as system specifications, use case diagrams, user case descriptions, ER-diagrams, user stories, burn down charts, and test plans, using the methods of Scrum make this work incremental and iterative. The online and onsite groups planned the work of each iteration to be improved upon in subsequent iterations and completed work was delivered throughout the project incrementally. Tasks were prioritized before the start of each sprint so working on documents did not obstruct the onsite and online team's progress.

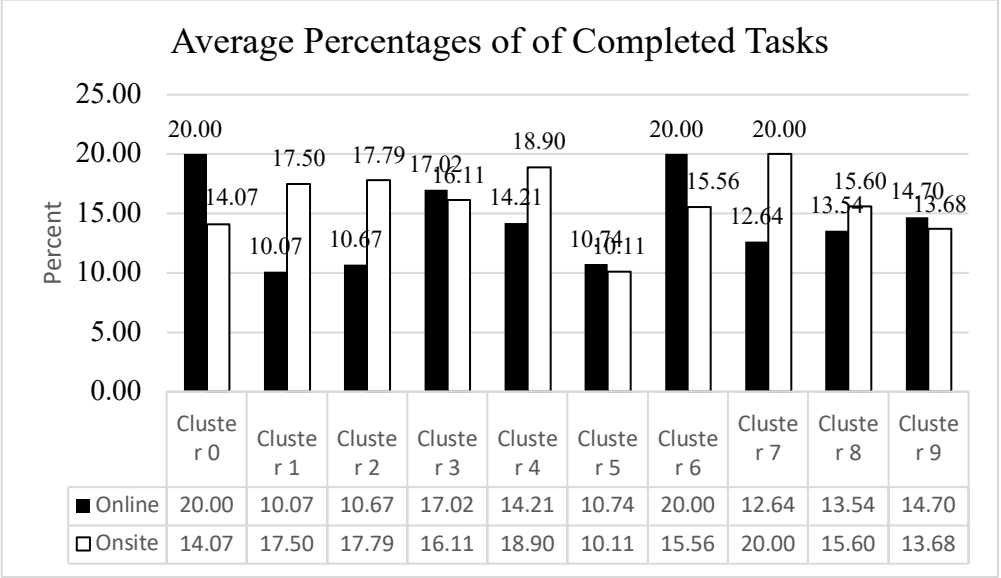

Figure 6 Average Percentages of Completed Tasks

Research question 3: Are there significant differences in customer collaboration for contract negotiation between online and onsite players learning Scrum utilizing Scrumification?

Answers from 62 students from the onsite and 74 from the online group showed average scores for the online group were equal to 4.541 and the average score for the onsite were equal to 4.371. The average value of the online group was slightly higher, but t = 1.268 and p value = 0.207. This means the two-sample groups were not significantly different in the aspect of customer collaboration for contract negotiation. The concepts of Scrum supported customers and Scrum teams working together, and customers were engaged and collaborated throughout the development process. In the experiment, both the online and onsite groups met with the product owners who acted as project customers in every sprint. The online group utilized Zoom, while the onsite group held face-to-face meetings for reviews at the end of the sprints to ensure that the product met the business needs of the customer.

Research question 4: Are there significant differences in the responding to changes between online and onsite players learning Scrum utilizing Scrumification?

The experiment used pooled sample proportions to calculate the p-values of the two sample groups. The information was gathered from survey questionnaires of the two sample groups. The questions asked about opinions towards change in the Scrum process during the experiment. 53 percent of the online group accepted the changes made during the development process, while only 38 percent from the onsite group accepted changes during the work. The results show that the difference in responding to changes between online and onsite players is 0.455 with t = 2.130 and

p value = 0.033. It shows that the two samples are different. The online group responded to change better than the onsite group.

Research question 5: What are the negative scrum activities that are difficult to understand in Scrumification?

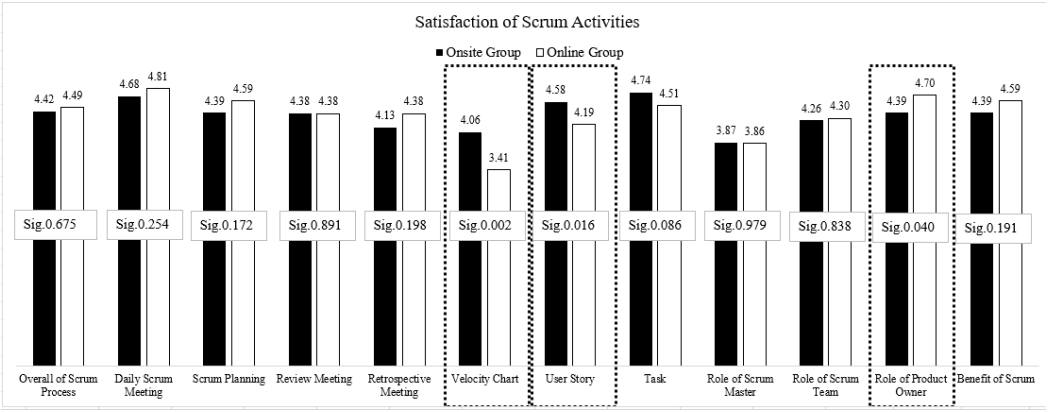

Figure 7 Comparison of Player's Satisfaction with Scrum Activities.

The average score of Scrum activities satisfaction (1 dissatisfied – 5 very satisfied) is show in Figure. 7. The results show that there are three Scrum activities with Sig. (2-tailed) values less than 0.05. It assumes that these three activities are not different: Velocity, User Story, and Role of Product Owner.

In addition, the satisfaction result shows that participants in the two groups almost never created or understood velocity charts and user stories. Velocity and user stories are Scrum artifacts. Data from the researcher's observation found that user stories were created by Scrum teams together with coaches or product owners. They were created once at the begin of the first sprint. Velocity charts were created at the end of each sprint, but that task was assigned to only one or two members. The low frequency of participants using these scrum activities had a negative effect on the learning outcome.

Research question 6: What are the positive game elements that make learning Scrum fun?

The result from Figure 8 showed that the "Leader board" was the favorite game element for both the onsite and online groups. In second place was "Challenge". The onsite group was more satisfied with "Reward and Points" than the online group, while the online group was satisfied with "Badges" more than the onsite group.

In addition, data was collected and analyzed from group interviews. The results from the interviews found that gamification made the Scrum framework more enjoyable, motivating, and engaging for all parties involved.

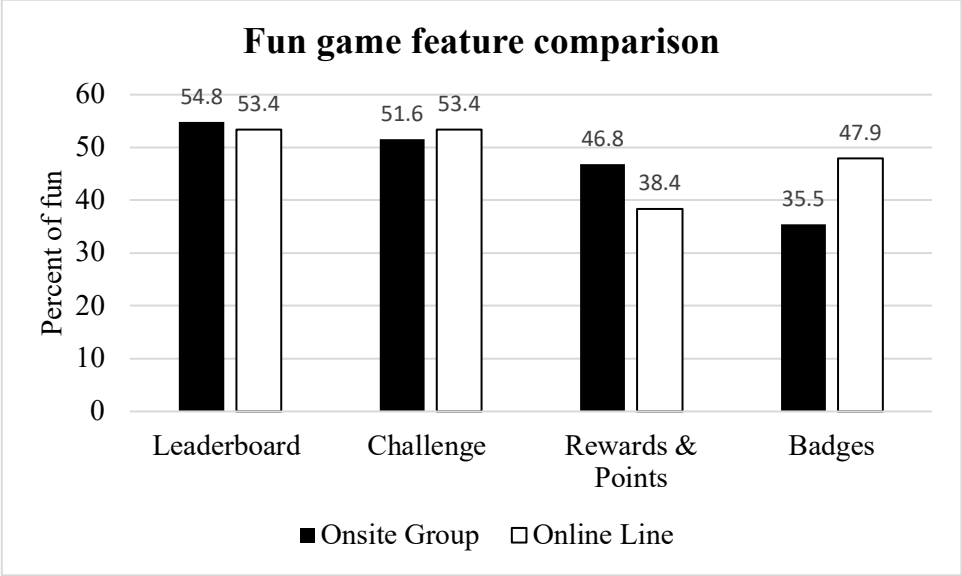

Figure 8 Comparison of Game Features between the Online and Onsite Groups.

CONCLUSION

The researchers began the research by collecting data from research papers and then produced the Scrumification framework. Three groups of people, Game designers, Game Players, and Game Masters, are involved. The scope of design consists of defining the problem, managing the people, and determining the expected learning outcome. The Game driven by the web application was called "Scrumification". The web application was designed with User-centered design (UCD), which is a method of incremental design in which designers put the emphasis on users and their needs at every stage of the process. The application was adopted for experiments with two groups of participants. The first group were students who attended a software development camp (OSSD#7) onsite for 7 days. The second group was made up of online learners who attend the OOSD#9 during the COVID-19 crisis. An online questionnaire was distributed at the end of the course. The survey was completed by 62 students from the onsite sample group and 74 students from the online sample group. The questions covered six research questions. The first four questions were designed to evaluate the differences between online and onsite student feelings toward utilizing the Scrum framework and following the four values of the Agile Manifesto. The result showed the online group was responding to change better than the onsite group, while other comparisons had no significant differences. The fifth question was designed to analyze the negative Scrum activities and the answer showed that the Velocity chart was the most difficult thing to understand. The last question was used to evaluate the fun game elements. The results showed that the "Leader board" was the favorite fun game element for both the online and onsite participants.

On the other hand, currently Scrumification which is the main tool for this experiment, was not completed. The next experiment will include the analysis of the performance of the Kanban board. In addition, there are many tools to support learning Scrum in the current market. For future experiments we should consider comparing our experience utilizing Scrumification with other market tools and apply gamification in other ways. Game elements to motivate the learners should be consider in selecting tools in order to avoid biases and misleading the learners about the Agile/Scrum concepts.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to the Faculty of Informatics, Burapha University, which provided funding for this research. I would like to thank their Software Engineering Department for letting me be part of this experiment.

REFERENCES

Ayunda, P. L., & Budiardjo, E. K. (2020). Evaluation of Scrum Practice Maturity in Software Development of Mobile Communication Application. 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE), 317-322, doi: 10.1109/IC2IE50715.2020.9274625

Bapat, H. B., & Hole, S. Y. (2020). A comparative study of online and offline mode of management education. Journal of Archaeology of Egypt/Egyptolog, 17(7). https://www.researchgate.net/

Beck K. (2001). Manifesto for Agile Software Development. https://agilemanifesto.org/

Schmidt, D. S., & Ramot, R. (2020). Opportunities and challenges: teacher education in Israel in the Covid-19 pandemic. Journal of Education for Teaching. 46(3). 1-10. doi: 10.1080/02607476.2020.1799708.

Hermanto, S., Kaburuan, E. R. & Legowo, N. (2018). Gamified Scrum Design in Software Development Projects. 2018 International Conference on Orange Technologies (ICOT), 1-8, doi: 10.1109/ICOT.2018.8705897.

Hof, S., Kropp, M., & Landolt, M. (2017). Use of Gamification to Teach Agile Values and Collaboration: A multi-week Scrum simulation project in an undergraduate software engineering course. The 2017 ACM Conference, 323-328. doi: 10.1145/3059009.3059043.

Jirapanthong, W., (2021). The Study of Active Learning on the Course of Game Project Management with Agile. JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY, 11(1), 30-38. https://doi.org/10.14456/jist.2021.4

Naik, N., & Jenkins, P. (2019). Relax, It's a Game: Utilising Gamification in Learning Agile Scrum Software Development. 2019 IEEE Conference on Games (CoG), 1-4, doi: 10.1109/CIG.2019.8848104.

Pedreira, O., García, F., Piattini, M., Cortiñas. A., & Cerdeira p. A. (2020). An architecture for software engineering gamification. Tsinghua Science and Technology, 25(6), 776-79. doi: 10.26599/TST.2020.9010004.

Schäfer, U. (2017). Training Scrum with gamification: Lessons learned after two teaching periods. Conference: 2017 IEEE Global Engineering Education Conference (EDUCON) , 754-761, DOI: 10.1109/EDUCON.2017.7942932

Singh, J., Dhindsa, K. S. & Singh, J. (2020). Performing Reengineering using Scrum Agile Framework, 2020 Indo-Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN), 33-35, doi: 10.1109/Indo-TaiwanICAN48429.2020.9181328.

Soesanto, R. (2021). Gamification for Student Achievement in Classroom: In Search of Requirement for Student Achievement Application. International Journal of Innovation in Enterprise System. 5. 90-99. doi: 10.25124/ijies.v5i02.138.

Thiel, S. (2016) A Review of Introducing Game Elements to e-Participation. 2016 Conference for E-Democracy and Open Government (CeDEM), 3-9, doi: 10.1109/CeDEM.2016.14.

Xiao1 B., Su, K., & Xiao, X. S. (2020). A Requirement Engineering Approach to User-Centered Design. Journal of Physics: Conference Series. 1453.