

An Initial Usability Testing for Improving Acceptance Criteria in A Scrum Project: An Angkasa LMS case study

Dana S. Kusumo*, Nungki Selviandro*, Kusuma Ayu Laksitowening*, Veronikha Effendy*, Monterico Adrian*

*School of Computing, Telkom University, Jl. Telekomunikasi no: 1, Bandung, Indonesia

danakusumo@telkomuniversity.ac.id, nselviandro@telkomuniversity.ac.id, ayu@telkomuniversity.ac.id, veffendy@telkomuniversity.ac.id, monterico@telkomuniversity.ac.id

Abstract— A sprint backlog is a placeholder for user requirements written as user stories and acceptance criteria for user stories. However, there is no standard technique used to involve users in validating the requirements at the beginning of the sprint. In this research, we conducted usability testing, involving seven participants, for three given tasks before a sprint planning in the development of Angkasa Learning Management System (LMS), a pay-as-you-go LMS on top of a cloud service. In the sprint review, five participants were asked for feedback and selected the best sprint increments between the sprint A dan sprint B. For this research, both sprints were identical which the sprint B had the same user stories, but the sprint B was tested using usability testing. The results show that the user acceptances in the sprint review were increased and user requirements in the user stories were validated by the users. This research contributes to the studies of user experience practices in the Agile-based projects to increase acceptance of software products.

Keywords— Scrum, sprint backlog, acceptance criteria, usability testing

I. INTRODUCTION

According to scrum guide, there are no specific requirement engineering techniques used in a Scrum project. Traditionally, RE consists of elicitation, analysis, validation, and documentation of requirement. Requirement in Scrum are captured in product backlog [1][2][3]. However, there are no specific requirement engineering techniques used to form product backlog [4], so that the product backlog not always the results of validated requirement.

In a sprint, a sprint backlog is implemented into working features. The validation of the features is in the sprint review. However, there is possibility that the respective users don't accept the feature in the review. Therefore, if the user can be involved earlier in the forming of acceptance criteria, the acceptance of features in the sprint review can be increased. The initial user involvement in Scrum can only take place during the product backlog refinement

Even though, in Scrum, product backlog is not formal user requirements [4], it is a placeholder for user requirements [5]. A user story usually is used to express the user requirement [5][6]. Jeffries argues that user stories can be effectively

written [6] to confirm the acceptance of the user stories. However, there is no definitive technique that can fasten the confirmation process.

Therefore, this research aims to facilitate the confirmation of the writing of user stories by providing technique to test the user stories. This approach can fasten the acceptance of product increment because the acceptance criteria have been initially tested by user in before the sprint planning. Therefore, it is expected in the sprint review, the user will be more eager to accept the finished features' increment because she/he has tested the product backlog using the usability testing in product backlog refinement.

II. RELATED WORKS

Scrum is a framework that can be applied within an organization or team to generate value through adaptive solutions to complex problems [7]. Scrum is pretty much applied to software development projects. In Scrum, we are familiar with the terms Scrum events, Scrum team and Scrum artifacts.

Scrum events consist of [7]: (1) the Sprint, (2) sprint planning, (3) daily scrum, (4) sprint review, and (5) sprint retrospective. Scrum team consists of [7]: (1) developer, (2) product owner and (3) scrum master. While scrum artifacts consist of [7]: (1) product backlog, (2) sprint backlog and (3) increment. The following are the stages that describe the process carried out in SCRUM in outline [7]:

- The product owner is fully responsible for managing the product backlog efficiently.
- The Sprint process begins with the Sprint Planning, where the Developers together with the Product owner will discuss to select items from the Product backlog that will be included in the Sprint Backlog to be completed in a Sprint and determine the Sprint Goal.
- Every day in a Sprint, a Daily Scrum is held, where developers will communicate with each other regarding the progress of work in the sprint, including what will be done today and how it will be technically done. After that, the developer will work on their responsible parts. And so on until the sprint is completed.

- After the sprint is completed, the Developer will present the resulting value to stakeholders at the Sprint Review event. During this event, the Scrum Team and stakeholders will discuss the results achieved and discuss possible changes if needed. In this process, it is possible for the product owner to update or improve his/her product backlog.
- Sprint retrospective is conducted after the Sprint review event is done. This event was attended by the Scrum team to discuss internal problems that occurred during the sprint process that had occurred, and to find solutions to these problems. During this event, it is also possible to select items into the sprint backlog for the next sprint.
- Next sprint started

From the description of the process, the quality of the Product backlog becomes quite crucial. No matter how good the value generated from the Sprint event is, if the main ingredient, namely the product backlog, is unclearly defined, it is very likely that stakeholders will reject the value presented in the Sprint review. By improving the quality of the product backlog, it is expected to increase the level of acceptance from stakeholders in the Sprint Review.

Requirements are anything that must be fulfilled. In terms of software, the requirements are the descriptions of what should be implemented by the software and how the software should behave [8]. It is a collection of needs from the perspective of both users and stakeholders [9].

Software requirements can be categorized into business requirements, user requirements, and functional requirements. Business requirements portray how the company/organization expects the software can bring advantages to their business, while user requirements describe the tasks that must be provided by the software to support user goals. Functional requirements define how the software must behave under certain circumstances [8].

The functional requirements must be aligned with the user requirements. Therefore, the business requirements could be achieved. This alignment is crucial for the software development success [8]. Therefore, it is essential to identify and document the software requirement in an appropriate way [10].

Requirement engineering is an approach to specify and manage software requirements systematically. It aims to gather relevant requirements, compromise them among the stakeholders, document them accurately, and manage them systematically. The objective is to avoid failure and minimize the risk that the developed software doesn't answer what users and stakeholders' needs [10].

To help improve the usability of a product, it is common sense that usability testing needs to be conducted in a software development process. The test should not be done just once at the end of the development process; however, it should be conducted as early as possible in the software development phases; in this case, it could be done in the requirements analysis phase. The usability testing can also be seen as an activity to improve the process by which the products are

designed and developed; in this case, the objective is to avoid repeating the same problems in developing a similar product [11].

Generally, in usability testing, the participants involved in the testing need to represent the actual users. For example, we may ask them to test some of the many tasks that users will be able to do with the software product that is being developed; this case is typically applied particularly in functionally rich software and complex software product. In general, the testing can be done by recording the participant's activity while performing the usability test (i.e., completing the given tasks). The testing result is then used as input to improve the software [12].

III. RESEARCH METHODOLOGY

For the purpose of this research, two identical sprints were compared (Sprint A dan Sprint B). The aim of this comparison is to observe the use of usability testing before the sprint planning. This scrum project was part of Angkasa LMS (Learning Management System) development. The Angkasa LMS is a pay as you go LMS on top of cloud. This scrum project has focused on the order services of The Angkasa LMS.

The sprint A was started from the sprint planning and proceeded until the sprint review. The sprint B was started with a sprint backlog tested by usability testing and then followed by a sprint event until the sprint review. The sprint backlog for both sprints was the same because we wanted to compare and analyse the two sprints. The sprint backlog had three features as follows:

- Product information
The user story of this feature is *"As a user, I want to get information about the types of products or services provided by Angkasa LMS so that I can determine the suitability of Angkasa LMS products with my needs"*.
- Pricing information
The user story of this feature is *"As a user, I want to get information on the price of Angkasa LMS services so that I can find out the budget that must be spent to use Angkasa LMS services"*.
- Buying package
The user stories of this feature are *"As a user, I want to register to Angkasa LMS so that I can become Angkasa LMS customer"*, *"As a user, I want to login so that I can use Angkasa LMS services"*, *"As a user, I want to choose payment method so that I can adjust with my preferred payment method"*, *"As a user, I want to explore offered Angkasa LMS package so that I can choose Angkasa LMS package that suit to my needs and budget"*, *"As a user, I want to check out before paying the package so that I can make sure that I have chosen the right package and payment method"*, *As a user, I want to see the details of the invoice detail so that I can double check that the order is correct, monitor payment status, and document proof of order"*.

Both sprints have the same development team consists of two UI/UX designers and one senior developer. Therefore, we expected that the feature increments had the same product quality because of the same development team.

In this research, the steps of the sprint planning dan the sprint event in the sprint A dan sprint B were different. The sprint A had normal sprint event sequence: the sprint planning and then the sprint event. In the following are the steps for the sprint B.

1. After initial sprint backlog definition, the product owner conducted a usability testing for three features. For the usability testing, there were modification related to above features as follows: adding three sentences in Product Information, adding two information about Pricing, and adding a Buy Now button in the landing page. The aim of these additions is to test whether more information and direct navigation will ease potential customers to decide and buy the Angkasa LMS services.

2. This testing aims to rapidly analyse and validate user stories based on our testing participants' input. To gather information as the input of usability testing, we used a maze (<https://t.maze.co/60719301?guerilla=true>). In this research, we had total 7 testing participants (see Table 1).

3. In the sprint planning, the updated sprint backlogs were selected for the current sprint.

4. The next, is the sprint event for the sprint B.

Then for both sprints, these following steps were conducted.

1. Sprint reviews were conducted for the feature increments. The incremented features, and user feedback were collected, compared, and analysed.

2. In the sprint reviews, we conducted A/B Testing to compare and evaluate the feature increment between Sprint A dan Sprint B. The A/B testing was considered as a sprint review because in this testing, the participants were also asked to give inputs and select the better increment of the three tasks between the sprint A dan sprint B. Of seven participants in the usability testing, only five agreed to participate in the A/B testing. We asked the participants to compare and select the best features from Sprint A (<https://angkasa.rg.telkomuniversity.ac.id/clone1>) and Sprint B (<https://angkasa.rg.telkomuniversity.ac.id/clone2>). We informed that one of the sprints was updated but we did not give further information that the input for the update was from one of the participants.

IV. RESULTS AND DISCUSSION

In this section, we present the research results and discuss the results. The purpose of our experiment is to compare and analyse between normal sprint backlog (Sprint A) and tested sprint backlog using the usability testing (Sprint B).

A. Results

The results obtained from the usability testing in Sprint B are presented in Table 1. The usability testing was conducted using a maze tool (<https://t.maze.co/60719301?guerilla=true>) before the Sprint B's Planning. There were three tasks as

usability testing for three features: product information, pricing information and buying package.

TABLE 1. THE RESULTS OF THE USABILITY TESTING USING MAZE TOOLS

Maze question	Answer
Total participants	7
Participant's roles	University (Head of Elearning Services), Educational Foundation (Digital Transformation Division as VP, AVP, IT Manager) High School (teacher and structural roles)
LMS user	Active (teaching, research, training, and service monitoring)
Have you installed a LMS?	4 out of 7
Have you had an interest to use a LMS?	All participants said yes
Feedback on task product information	The description of product and service information was too short. There was no customized package information. There were no interoperability features in product package information. There were typos. There were a few asymmetric UI design. There were non clickable navigations and buttons.
Feedback on task pricing information	The upgrade package should be added. It was not clear whether the service duration one time charge, monthly or annually. The duration of pricing information was too short.
Feedback on task buying package	The product information for checkout was too short. The checkout page needed more information, for example: duration of services and time information for payment. The invoice was too simple.

Of seven participants, only one participant whose feedback was used to update the sprint backlog for the Sprint B. We considered the feedback from this participant and did not request follow-up feedback from other participants because this participant is the VP Digital Transformation Division in a leading educational foundation in Indonesia covering from basic to higher education. His main role includes transforming digital environment in all schools and universities in the educational foundation. Based on this updated sprint backlog, the developer built updated product, pricing and buying package features for the Sprint B.

The following is the results of the A/B Testing for the Sprint A (<https://angkasa.rg.telkomuniversity.ac.id/clone1>) and the Sprint B (<https://angkasa.rg.telkomuniversity.ac.id/clone2>). Only five

out of seven participants in the usability testing, including the VP, agreed to participate in the A/B Testing.

TABLE 2. THE RESULTS OF THE A/B TESTING IN THE SPRINT REVIEW

Question and Input	The Testing Results
Which was the best product information feature (Sprint A/Sprint B)	Total selection for: Sprint A: 0 Sprint B: 5
Which was the best pricing information feature (Sprint A/Sprint B)	Total selection for: Sprint A: 0 Sprint B: 5
Which was the best buying package feature (Sprint A/Sprint B)	Total selection for: Sprint A: 1 Sprint B: 4
Input for product information feature	It is better to add information about favorite packages/products. The interface for this feature in the Sprint B is better. It is better to display the interface of the LMS. Information about LMS product must be detailed, especially for new LMS users. The label caption of Start Now and Learn More are confusing because the provided information for these two buttons quite similar. It is better to improve and add more detail information about product.
Input for pricing information feature	It is better to have annual subscription prices (with discounted prices compared to monthly price). It is better to add more detail information about products especially the difference features and limitation among products.
Input for buying package feature	It is better to have chatbot services and top-up information so that when the services end, the services can be renewed automatically. It is better that the customer can download the invoice.
General input	There are symmetric alignment problems among HTML element/division tag.

From the above Table, we can summarize that Sprint B was selected almost by all participants and only one participant selected buying package feature in the Sprint A.

B. Discussion

Based on the usability testing results, in the maze's heatmap feature show that 5 participants cannot find the product information (they navigated to main feature information) and only 1 participant can find the product information. Based on this result, the researchers decide to merge product information and main feature information into

product information as sprint B's increment. Because in the maze's heatmap feature show that 2 participants successfully find the pricing information, the pricing information still use the same design as the usability testing design added with monthly charge for the package price. In the usability testing in task buying package, only 2 participants tried the task and all of them did not successfully complete the task because some navigations and buttons were not set properly in the maze's tools. As the result of the usability testing for this task, we only added monthly charge for the package price in the process of buying package.

Table 2 shows that almost all participants in the A/B testing agree that product increments in the Sprint B is better than in the Sprint A. As stated by the testing's participants in Table 2 that compared to the increment in the Sprint A, the increment in the Sprint B has more information needed for the potential users (for Product and Pricing information's increment. For the task of buying package (Table 2) with only small changes in the Sprint B, only one participant opted for the Sprint A. Based on the results in Table 2, the usability testing can increase acceptance rates for the user stories without changing the user stories. Thus, the benefits of the usability testing are to increase users 'acceptance in the sprint review.

As the testing in the form of prototyping testing, so that the testing can be done rapidly and can get quick feedback. This mechanism is similar to build measure learn in the lean start-up [13]. Therefore, the second benefit of the usability testing is to use validated learning from the usability testing's feedback to implement user stories whose user stories were validated as the results of the usability testing. From the perspective of requirement engineering, the usability testing validated user's needs [10] in the early software development using prototyping testing approach.

The sprint review, as shown in Table 2, indicates feedbacks that can be used to improve the features. The feedbacks do not reject the acceptance criteria. We argue that because the A/B testing's participants were the usability testing's participant, their initial understanding of the built features makes the feedback in the Sprint Review advanced. With our approach, the users' participation shows positive result and still accordance with the Scrum by adding users' participation before the Scrum planning [14]. Based on Patton [14], as our product development is a consumer commercial, the Angkasa LMS used two practices of UX in the Scrum project [14]: using prototype as specification and increasing user validation by adding the usability testing in the beginning of the sprint.

V. CONCLUSION

In this research, we have conducted usability testing before the sprint planning for building three features in the development of Angkasa LMS. The features are product information, pricing information and buying package. The Angkasa LMS is a moodle based LMS on top of the cloud providing a pay as the go LMS services so that users can get affordable LMS services based on their LMS usage. The aim

of the test is to get feedback for these three features before the sprint begun. The user participants yield some benefits: increasing acceptance in the sprint review, validating user requirement in the user stories, increasing of users' understanding on the requirement and of users' participation on the Scrum project. The usability testing functions as card to facilitate conversation between product owner and his/her users thus can make better user stories and acceptance criteria. Further work is required to implement more UX practices in Agile-based projects.

ACKNOWLEDGMENT

The authors thank to Kedaireka 2021 grant from the Ministry of Education and Culture of the Republic of Indonesia for supporting this research.

REFERENCES

- [1] N. Bolloju, S. Alter, A. Gupta, S. Gupta, and S. Jain, "Improving Scrum User Stories and Product Backlog Using Work System Snapshots," *AMCIS 2017 Proceedings*, Aug. 2017, [Online]. Available: <https://aisel.aisnet.org/amcis2017/SystemsAnalysis/Presentations/7>
- [2] V. T. Heikkilä, D. Damian, C. Lassenius, and M. Paasivaara, "A Mapping Study on Requirements Engineering in Agile Software Development," in *2015 41st Euromicro Conference on Software Engineering and Advanced Applications*, Aug. 2015, pp. 199–207. doi: 10.1109/SEAA.2015.70.
- [3] E.-M. Schön, J. Thomaschewski, and M. J. Escalona, "Agile Requirements Engineering: A systematic literature review," *Computer Standards & Interfaces*, vol. 49, pp. 79–91, Jan. 2017, doi: 10.1016/j.csi.2016.08.011.
- [4] T. Sedano, P. Ralph, and C. Péraire, "The Product Backlog," May 2019, pp. 200–211. doi: 10.1109/ICSE.2019.00036.
- [5] K. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*, 1st edition. Upper Saddle River, NJ: Addison-Wesley Professional, 2012.
- [6] J. Ronald E., "Essential XP: Card, Conversation, Confirmation." <https://ronjeffries.com/xprog/articles/expcardconversationconfirmation/> (accessed Nov. 30, 2021).
- [7] K. Schwaber and J. Sutherland, "The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game.(2020)," Available: *scrum.org*, 2020.
- [8] K. Wiegers and J. Beatty, "Software Requirements, 3rd Edition." <https://www.pearson.com/content/one-dot-com/one-dot-com/us/en/higher-education/program.html> (accessed Nov. 30, 2021).
- [9] A. Aurum and C. Wohlin, *Engineering and Managing Software Requirements*. Springer Science & Business Media, 2005.
- [10] K. Pohl and C. Rupp, *Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam - Foundation Level - IREB compliant*, 2nd edition. Santa Barbara, CA: Rocky Nook, 2015.
- [11] S. Riihiäho, "Usability Testing," in *The Wiley Handbook of Human Computer Interaction*, John Wiley & Sons, Ltd, 2018, pp. 255–275. doi: 10.1002/9781118976005.ch14.
- [12] J. S. Dumas and J. C. Redish, "A Practical Guide to Usability Testing: ., Redish,," <https://www.amazon.com/Practical-Guide-Usability-Testing/dp/1841500208> (accessed Nov. 30, 2021).
- [13] T. R. Eisenmann, E. Ries, and S. Dillard, "Hypothesis-Driven Entrepreneurship: The Lean Startup," Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 2037237, Mar. 2012. Accessed: Nov. 25, 2021. [Online]. Available: <https://papers.ssrn.com/abstract=2037237>
- [14] J. Patton, "Twelve Emerging Best practices for Adding UX Work to Agile Development – Help your organization focus on successful outcomes." https://www.jpattonassociates.com/emerging_best_agile_ux_practice/ (accessed Nov. 29, 2021).



Dana Sulistiyo Kusumo received BEng in Informatics from the Telkom University, Indonesia, MEng in Information Technology from Bandung Institute of Technology, Indonesia and a Ph.D. degree in Computer Science and Engineering from the School of Computer Science and Engineering, The University of New South Wales, Australia. Currently, he is a lecturer at the School of Computing, Telkom University, Indonesia. He is also a member of the Software Engineering Research Group at the School of Computing, Telkom University, Indonesia. His research interests include Software Engineering and Information Architecture.



Nungki Selviandro received both BSc and MSc degrees in Computer Science from the University of Indonesia, Indonesia, and a Ph.D. degree in Computer Science from the Department of Computer Science, The University of York, UK. Currently, he is a lecturer at the School of Computing, Telkom University, Indonesia. He is also a member of the Software Engineering Research Group at the School of Computing, Telkom University, Indonesia. His research interests include Software Engineering, Model-Driven Software Engineering, Assurance Cases, and XAI.



Kusuma Ayu Laksitowening received her bachelor's degree in Informatics from Telkom University, Master's degree in Information Technology from Bandung Institute of Technology, and Ph.D. degree in Computer Science from Universitas Indonesia. Currently, she is a lecturer and head of the Software Engineering Research Group at the School of Computing, Telkom University, Indonesia. Her research interests include Software Engineering, Personalized Learning, and Learning Analytics.



Veronikha Effendy received her bachelor's degree in Informatics Engineering from Telkom Institute of Technology (STITelkom) in 2003, and Master's Degree in Informatics from Telkom University, Bandung, Indonesia, in 2013. She has worked in IT industry for 8 years, and now she is a lecturer and researcher in School of Computing Telkom University. She is also a member of the Software Engineering Research Group at the School of Computing, Telkom University, Indonesia. Her current research interests include Human Computer Interaction, Software Engineering and Data mining.



Monterico Adrian is a Research Associate at School of Computing, Telkom University, Indonesia. He received his master and bachelor degree in Informatics Engineering from Institut Teknologi Bandung, Indonesia. His current research interests are software engineering, database, programming, and mobile technologies. He is also a software developer and IT consultant.