The Indonesian Journal of Computer Science
www.ijcs.net
Volume 13, Issue 5, October 2024
https://doi.org/10.33022/ijcs.v13i5.4413

IT Team Project Management Transformation Plan for TIC Company's IT Division

Fauzan Aldiansyah[1], Teguh Raharjo[2] , Anita Nur Fitriani[3]
fauzan.aldiansyah21@ui.ac.id[1], teguhr2000@gmail.com[2], anita.nur02@ui.ac.id[3]
[1,2,3]Faculty of Computer Science, Universitas Indonesia, Jakarta, Indonesia

Article Information

Received : 28 Sep 2024
Revised : 24 Oct 2024
Accepted : 30 Oct 2024

Keywords

Project Management, SDLC, Waterfall, Kanban, Scrum, Transformation

Abstract

Nowadays, organizations are increasingly reliant on information technology to maintain efficient and effective internal systems, stay competitive, and meet the evolving needs of their customers. However, organizations facing challenges often include delays in project delivery, difficulties in managing workflows, and issues with accurately tracking progress and performance. The Information Technology Division of one of the testing, inspection, and certification companies in Indonesia is responsible for application development but frequently encounters operational issues such as development delays, unaddressed requests, difficulties in tracking project timelines and completions, and unclear job description alignments. This study examines Waterfall, Kanban, and Scrum Agile methodologies through case studies and Design Science Research Methodology to address these issues. Scrum Agile, with its principles and practices, is selected as the preferred solution for resolving these recurring problems. The research proposes transitioning project management to Scrum Agile methods and demonstrates how these methods effectively address several existing challenges. This approach results in the development of a new system that implements Scrum Agile for project management within the department. This research can enhance the efficiency and effectiveness of application development and improve project visibility and control for the organization. Additionally, it provides learning through comparative analysis to identify appropriate new methods and systems based on the case study.

Attribution-ShareAlike 4.0 International License

A. Introduction

In today's fast-paced business environment, organizations are increasingly reliant on information technology to maintain efficient and effective internal systems, stay competitive, and meet the evolving needs of their customers. However, many companies, face common challenges in information technology implementation and development that can hinder their operational efficiency and overall success. These challenges often include delays in project delivery, difficulties in managing workflows, and issues with accurately tracking progress and performance. For organizations engaged in services these challenges can be significant.

The need to streamline processes, manage resources effectively, and deliver timely results becomes even more crucial. Inefficiencies in internal systems not only impact project timelines but also affect customer satisfaction and organizational effectiveness. Late delivery has been a common problem in the software industry for decades, with software projects often exceeding their deadlines by approximately 30% [1]. A study of 720 software development projects revealed that using an inappropriate methodology is the most critical factor leading to project delivery failures [2]. Research suggests that a large percentage of IT projects either fail before completion and one of primary reasons for these failures is the lack of appropriate assessment and risk management throughout the project [3].

One of the teams that experienced these issues is the Information Technology Division of one of the testing, inspection, and certification companies in Indonesia, which company's daily activities are related to inspection and testing, which produces certificates of assurance for its customers. This company provides various services and services to advance the economic value of the Indonesian business sector. In support of its operations, besides intensify the business and engineering, it also continually leverages and advances technology to enhance the company's value by boosting revenue, improving efficiency, reducing work time, and streamlining efforts. One of its divisions, the Information Technology Division, is responsible for developing applications and automation, as well as maintaining its applications, systems, and infrastructure. This division facilitates application developments and automations across throughout the company entities. Comprised of a team of 3 planners, 15 developers, 2 project manager, and 3 operational staff, the team has faced several issues in the past year. These problems mainly include delays in development, handling numerous unaddressed requests, inability to track project timelines and completions, and unclear job description alignment.

Currently, application development requests are directly sent to the planning team for analysis. Once the request document is prepared, it is passed on to the development team for system analysis and programming. However, the division is not operating efficiently at this stage because each developer is solely responsible for an entire application, a process known as monolithic development. This critical situation arises due to unclear ticket distribution, which fails to specify the tasks being handled. As a result, the project manager is unable to monitor the progress or ongoing work of the development team in real time.

Monitoring data from 2023 to April 2024 shows that out of 17 application development and establishment requests, 11 applications surpassed the target deadline. Additionally, there are 10 applications still in the queue for development, with 5 of them being carryovers from requests made in 2023. The presence of this application queue also introduces uncertainty regarding when the requested applications will commence and the estimated completion time that needs to be communicated to the user. This happens because there are no clear time limits, and the work is not divided up well enough to make the workflow smoother. The number of applications in the queue also indicates a lack of accuracy in predicting the duration of application development based on timeboxing. For planners who deal directly with users, having time limits in place is important so they can give estimation about when projects will start and how long they will take to finish. Based on observations, there is a need for system improvement and refinement in the team, particularly in application development project management. This requirement is to accommodate the timely delivery of applications and provide clear information to users. Research in this case study is conducted to design a system within the division. The aim is to simplify progress tracking, job estimation, enhance efficiency, and provide estimated completion timelines, ensuring a prompt response to development requests from users.

B. Research Method

The methodology used in this research is the Design Science Research (DSR) methodology. Design Science Research is a problem-solving procedure to intensify knowledges through the creation of innovative artifacts. DSR resulted in the designed artifacts and design knowledge that provides a fuller standing in the reason of the artifacts supporting relevant application contexts.

DSR studies relevant problems in the real-world environment with various application domains. DSR create an innovation solution which extent knowledge for the topics by following the “routine design” which in most applications can refer to the DSR of form of the theories, frameworks, instruments, or design artifacts such as constructs, models, methods, or instantiations [4].

C. Result and Discussion

The results and discussion from this research using DSR presented as points given.

1. Problem Identification and Motivation

The traditional methodology for software development were often unsuccessful due to the rapid growth of the IT industry [5]. Based on direct observation on the division and interview with 3 planner, 15 developer, 2 project manager, and 3 operational staff, The problem is the team have not used the right software development process where there are often delays in targets and requests often increase during the coding process. There is also a need of ability to predict development time, resource allocation, and allocation for other projects based on the number of development requests. According to developers also stated that they were overwhelmed by the number of projects piling up and handling alone by one developer per project.

2. Define the Objectives for a Solution

The objective for the solution is to solve the problems described previously. One of the main objectives is to provide an estimated time for project completion so that allocation for subsequent projects can be arranged. Defining the objectives by implementing the DSR methodology, as shown in Table 1.

Table 1. Design Science Research Methodology

DSR Process	Process Implementation
Problem identification and motivation	Defines the specific research problem namely delays in development and inability of project tracking then justifies the goals of a solution
Define the objectives for a solution	Knowledge what is possible and feasible based on objectives that is project estimation
Design and development	Artifacts that are created contribute to the design provides the design of proposed method implementation
Demonstration	Demonstrates the use of the artifacts which is new method and system
Evaluation	Measures how well the new method support solution to the problem
Communication	The problems and the designed method and system are communicated to the relevant stakeholders, and journal publication

Based on interviews with the team, advice obtained that the development team needed to transform into new method in software development which were proven to be more effective in overcoming the obstacles. To generate the objectives of the solution, research was conducted to provide the team with new methods. The research involved a literature review to gather methods from various sources as references. From several references, SDLC methods were collected along with their advantages and disadvantages for consideration. Based on reference searches, there are several SDLC methods utilized by development teams, such as Waterfall, Iteration, Rapid Application Development (RAD), Extreme Programming, FAST, Prototype, Spiral, Zachmann Framework, End User Computing Satisfaction, Simple Additive Weighting (SAW), The Delone and McLean Methods, Agile Scrum, and V-Model [6]. Some frequently used methods include Waterfall, Agile Kanban, and Agile Scrum.

Based on references, an analysis was conducted on the advantages and disadvantages as considerations to provide solutions for the issues. A summary of the three commonly used SDLCs can be seen in Table 2.

Table 2. Advantages and Disadvantages of SDLC Methods [6-11]

SDLC Method	Advantages	Disadvantages
Waterfall	• Requirement is clear before development starts • Phase needs to be completed before moving to the next phase • Linear model, easy to implement Resources required is minimal	• Many problems with one phase that never solved completely Requirement needs to be clear, and problems could arise if the requirement changes
Agile	• Workflow visualization	• Continuous framework

Kanban	• Task limitations/restrictions • Focus on workflow • Execution based on Work in Progress (WIP) limits • Minimal changes	• Constant work without planning • No defined roles and set ceremonies • Assessment and prioritization by choice Not stuck by schedule, items released when ready
Agile Scrum	• Iterative framework • Timeboxing by sprints • Defined roles • Has ceremonies • More flexible and requirements Mandatory assessments and prioritization	• Disruption in teamwork • Module integration issues • Release process Documentation

Based on the data from Table 1, Agile Scrum methodology has been identified as the optimal solution due to its inherent advantages that effectively address the challenges outlined earlier. By implementing Agile Scrum, this division aims to enhance efficiency, collaboration, and responsiveness in software development processes. The detailed findings and recommendations are documented in Table 3, reflecting the strategic alignment of Agile Scrum with the division's goals and operational needs.

Table 3. Problems and Scrum Resolutions

Problems	Scrum Resolutions
Each developer is responsible for the entire application (monolithic architecture)	Scrum encourages cross-functional teams where members collectively possess all the skills necessary to deliver the product
Unclear ticket distribution system	The project divided into tickets results in the distribution of tasks among developers
Work is not divided up efficiently among developers	Dividing tickets optimizes developers and reduces the chance of them being idle due to tasks being unassigned or waiting
Lack of ability to monitor progress effectively	The progress of application development can be tracked from tickets that are completed and those that are not yet finished
Uncertainty about application queue (no clear time limit) leads to inaccuracies in predicting application development duration	Timeboxing in Scrum helps establish clear limits on the progress of the project, and timeboxing with the number of tickets yet to be worked on can assist in estimating project completion

These advantages include flexibility in adapting to changing requirements, iterative development cycles that promote continuous improvement, and regular feedback loops through frequent daily meetings.

3. Design and Development

In this stage, this research designs a new method in the software development process. According to results of references and evaluation in the previous stage, the proposed new method for the team is transforming to traditional method to

Agile Scrum method, following the Scrum Guide that demonstrated by previous projects in other research studies. Scrum by frequent release resulted in the increased flexibility and Scrum process testing of product is integrated which enables continuous inspection of the product as it gets developed, and increased communication, decisions-making, and clear idea for items development [10]. By using agile, the project also visible and controllable [12]. Also, the concept of Scrum uses timeboxing, which can answer the main problem of project time estimation.

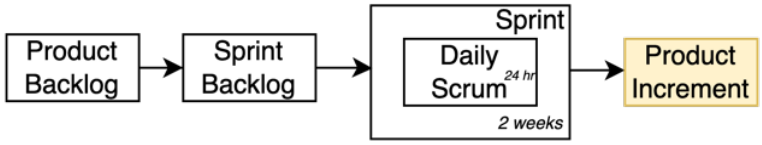

Figure 1. Scrum Guide [13]

3.1. Team

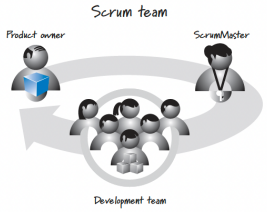

Figure 2. Scrum Roles [14]

In Scrum values and procedures, there is a component of team distribution. This team has 3 planners, 15 developers, 2 project manager, and 3 operational staff. Based on this, the team was designed to transform into 3 Scrum teams shown on Table 4.

Table 4. Scrum Roles [6-11]

Current Roles	Scrum Roles	Scrum Team 1	Scrum Team 2	Scrum Team 3
Planner	Product Owner	1 person	1 person	1 person
Project Manager	Scrum Master	2 persons		
Developer	Developer	5 persons	5 persons	5 persons
Operational Staff	Tester and Deployer	1 person	1 person	1 person

The team distribution is based on the current work in the division and transform following the scrum guide [13]. The planners become product owners who will maximize the value of the product and create effective backlogs. The project manager becomes the Scrum Master for both teams who will ensure that the understanding and implementation of Scrum is appropriate and effective. Developers working collaboratively within the Scrum framework to create usable increments. Also, operational staff who are people responsible for infrastructure and operations in the division will have additional work and responsibilities as testers on the Scrum team.

The Scrum process starts with a request for production or development from users that come from other divisions through the division helpdesk. The Head of the Division will assign the work to team 1, team 2, or team 3 based on load and allocation. After the request is made, Scrum Events are then executed.

3.2. Scrum Events

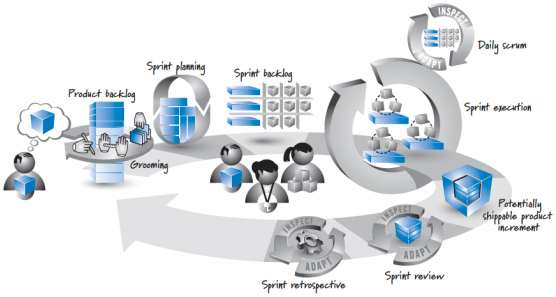

Figure 3. Scrum Framework [14]

3.2.1. The Sprint

Sprints operate in general where the Sprint consists of a duration of two weeks, trying to complete the tasks given according to office working days and hours [14]. Sprints are carried out to complete the sprint goal proposed in the sprint planning. Before the sprint begins in sprint planning, it needs to be clarified that the project results are in the form of product goals which are then distributed into items called product backlog items by the product owner. For the team, it is proposed that the media to be used is an online board where items will then be displayed on the TV available in the room.

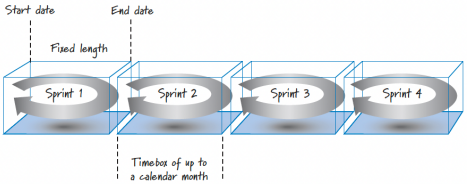

Figure 4. Scrum Timebox [14-15]

3.2.2. Sprint Planning

Sprint planning is a meeting to make the agenda of what will be done is this sprint, That the product owner will determine the purpose and mission of each feature that already designed in the product backlog and part of team members will discuss the how many hours will be spent to finish the backlog item. The items selected to be worked on in this sprint will be documented in the Sprint Backlog with the value to be achieved being the Sprint Goal. Due to its working days is started on Monday, then the sprint planning is carried out on Monday with a maximum time of four hours [13]. To determine the time needed to work on these items, poker planning will be used [14].

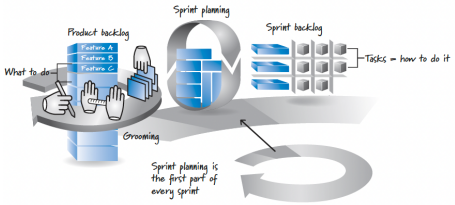

Figure 5. Sprint Planning [15]

3.2.3. Sprint Planning

Daily Scrum is to check the progress toward Sprint Goal and adapt the sprint backlog as necessary if there are any needs. The Daily Scrum held for 15-minutes maximum [13]. The Daily Scrum will be a progress monitoring that can improve communication among scrum team members that will unleash the project [16]. The Daily Scrum will be held every working day, standing by, for each team. The daily scrum implementation time will be sequential for each team because the scrum master handles two teams and must be present in this event.

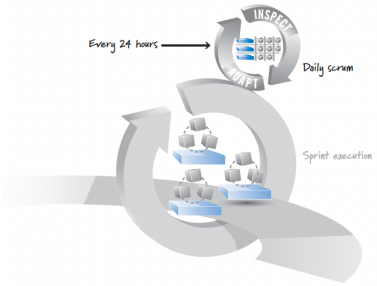

Figure 6. Daily Scrum [14]

3.2.4. Sprint Review

Sprint Review event is to inspect the outcome of the Sprint where the Scrum team presents of the work to the stakeholders [13]. The Scrum Team reviews what has been done and collaborates on what to do next, allowing the product backlog to be adjusted to meet new opportunities or developments. The sprint review has a maximum time of 2 hours and is presented on the last day of each sprint to the Division Head who will review and provide input for development due to the Division Head as the person responsible for the product and correspond with the user's division head. Presenting the increment results of the sprint will involve an application that has been deployed in the company's development environment so that it can be tested directly by various parties, especially stakeholders.

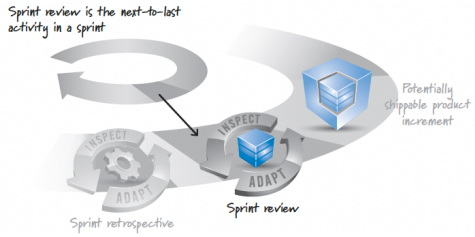

Figure 7. Sprint Review [14]

3.2.5. Sprint Retrospective

Sprint Retrospective is an activity to increase quality and effectiveness that will be taken as advice to next sprint and events. This activity will be where all members of the team will provide opinions regarding obstacles from the sprint as well as recommendation for upcoming sprints. This event will be held after the Sprint Review which is at the last day of the sprint and maximum 1.5 hours. The process will use a dedicated board where all team members can put suggestion tickets, then the scrum master will lead the process to discuss the items.

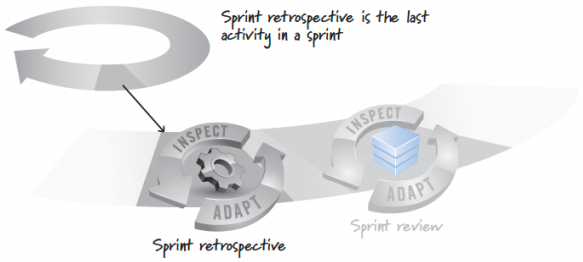

Figure7. Sprint Retrospective [15]

Overall, the designed scrum framework for the team presented in Table 5.

Table 5. Scrum for Information Technology Division

Event	Timebox	Information	Process
Sprint	2 weeks	Using online board	Sprint on Scrum Guide
Sprint Planning	Max 4 hours		
Daily Scrum	30 minutes	Every morning working day	Progress monitoring, sequences for Scrum Master
Sprint Review	Max 2 hours	Presented to the Division Head	Using increments that have been added to the application in the development environment

3.3. Demonstration

Based on the large number of requests for application development in the information technology division, users often feel that their requests are not being addressed. On the contrary, this happens because the team has not been able to develop the requested application as soon as possible, and one of the main problems is that the team cannot provide an estimate of the time when the application requested by the user will start to be developed, this is due to the development in the division is not yet organized and there is no time estimation, this leads to the users feel not being responded by the team.

The framework that has been designed has an estimate of the time and work that will be done by the team, so that the load and resource allocation will be more effective. As a helpdesk and Division Head, it can also provide an estimated time for development of user's application requests lead to the users get informed and certainty. Also, the division does not feel like they are constantly being chased regarding development requests.

3.4. Evaluation

The framework was designed using the agile scrum method that uses sprints which product goal are distributed into items and timeboxed. This will be a solution to the problems described previously. As an evaluation, further interviews or questionnaires are also required, as well as evaluation of the timeliness of development, also user satisfaction with the information technology division if this agile scrum framework is implemented.

3.5. Communication

Implementing this new method may not be easy, there will be many things that need to be considered and done before finally carrying out this transformation. One important thing is related to communication. The plan that has been designed needs to be submitted to the division head for review and approval. Then, comprehensive outreach is needed to all parts of the division to be aware of the transformation that will be carried out, starting with an explanation regarding the benefits that will be achieved if this framework is implemented. Then communicate concerns from all division members and discuss the related concerns. Until the transformation is ready to be implemented.

D. Conclusion

This research goals are to design new methods to transform Information Technology Division of one of the testing, inspection, and certification companies in Indonesia development process from traditional to new refined system. This plan is designed to overcome problems the team faces in dealing with user's requests and daily operational. Research conducted and the agile scrum method was chosen because of proven previous works and research that can solve the problems mentioned. The design follows the Scrum Guide, which is adapted to the conditions in the company. The future work of this research is an evaluation of the implementation of the design. This will be a consideration regarding the effectiveness of the planned framework for resolving the problem.

E. Acknowledgment

The author declares no conflict of interest. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

F. References

[1] E. Kula, E. Greuter, A. Van Deursen, and G. Gousios, "Factors Affecting On-Time Delivery in Large-Scale Agile Software Development," IEEE

Transactions on Software Engineering, vol. 48, no. 9, pp. 3573–3592, Sep. 2022, doi: 10.1109/TSE.2021.3101192.
[2] A. Q. Mohabuth, "A Study of the Identification of the Factors that Lead to Time Delays in Software Development," International Journal of Computer, vol. 25, pp. 141–148, 2017, [Online]. Available: http://ijcjournal.org/
[3] P. Gharibnejad and B. Ostadi, "Evaluating and prioritizing the failure factors and cause of delays in IT projects using FMEA: Towards project continuity," 2020.
[4] J. vom Brocke, A. Hevner, and A. Maedche, "Introduction to Design Science Research," 2020, pp. 1–13. doi: 10.1007/978-3-030-46781-4_1.
[5] M. Gaborov et al., "Journal of Applied Technical and Educational Sciences jATES Comparative analysis of agile and traditional methodologies in IT project management," vol. 11, no. 4, pp. 1–24, 2021, doi: 10.24368/jates.v11i4.279.
[6] D. Trivedi, "Agile Methodologies," International Journal of Computer Science & Communication, vol. 12, no. 2, 2021, [Online]. Available: https://www.researchgate.net/publication/356924683
[7] Oktaviano, H. R., & Firmanto, Y., "PENGEMBANGAN SISTEM INFORMASI ASET TETAP PEMERINTAH KABUPATEN X: SYSTEM DEVELOPMENT LIFE CYCLE (SDLC) TEKNIK WATERFALL," 2021. [Online]. Available: https://jimfeb.ub.ac.id/index.php/jimfeb/article/view/7462
[8] N. Faizah, N. Santoso, and A. A. Soebroto, "Pengembangan Sistem Aplikasi Manajemen Proyek menggunakan Kanban Framework," 2019. [Online]. Available: http://j-ptiik.ub.ac.id
[9] P. A. G. Permana, "Scrum Method Implementation in a Software Development Project Management," 2015. [Online]. Available: www.ijacsa.thesai.org
[10] M. Ali, "Benefits to Organizations after Migrating to Scrum," 2017. [Online]. Available: https://www.researchgate.net/publication/317066055
[11] S. Kute and A. S. Thorat, "A Review on Various Software Development Life Cycle (SDLC) Models," International Journal of Research in Computer and Communication Technology, vol. 3, 2014, [Online].
[12] A. Nabilah, "The benefit of implementing agile scrum in project management," 2019, [Online]. Available: https://www.researchgate.net/publication/338139300
[13] K. Schwaber and J. Sutherland, "The Scrum Guide The Definitive Guide to Scrum: The Rules of the Game," 2020.
[14] K. S. Rubin, Essential Scrum. New Jersey: Pearson Education, Inc. , 2012.
[15] R. S. Pressman, Software Engineering: A Practitioner's Approach, vol. 7. 2010. [Online]. Available: www.mhhe.com/pressman.
[16] O. C. Resmi Rachmawati, Deyana Kusuma Wardani, Wifda Muna Fatihia, Arna Fariza, and Hestiasari Rante, "Implementing Agile Scrum Methodology in The Development of SICITRA Mobile Application," Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 7, no. 1, pp. 41–50, Feb. 2023, doi: 10.29207/resti.v7i1.4688.