

Implementation of Scrum for Developing the Data Visualization Dashboard for the PLN UP3 Yogyakarta Project

Hestri Apriliani Widowati^{1*}, Andhik Budi Cahyono²

Program Studi Informatika, Universitas Islam Indonesia, Yogyakarta, Indonesia^{1*2}

Email: 20523074@students.uii.ac.id^{1*}, 105230101@uii.ac.id²

ABSTRACT

In the web application development process, selecting the right method is crucial to achieving goals effectively and efficiently. Previously, applications at PLN UP3 Yogyakarta were developed using the waterfall development method. Some projects that were developed without a structured method progressed slowly and were not well organized. Therefore, for the PLN UP3 data visualization dashboard project, the Scrum method was chosen. The Scrum method was selected for its flexibility, rapid development capabilities, and ease of implementation by all team members. The implementation of Scrum follows these stages: setting product goals, creating a product backlog, sprint planning, conducting sprints, producing increments, holding sprint reviews, conducting sprint retrospectives, and daily stand-ups. By using the Scrum development method, the SIBER project at PLN has been able to progress according to the specified timeline and successfully deliver all required features.

Keywords: Scrum Method, Web Application Development, Real-Time Dashboard, Data Visualization Dashboard.

INTRODUCTION

Currently, many organizations have digitized their products to enhance effectiveness in improving service quality (Sjödín et al., 2020), including PLN. PLN is currently making adjustments in information technology development to enhance efficiency and provide better services that employees can use to simplify their work, especially in data analysis. For example, PLN has developed several mobile web application-based products, such as PLN Mobile, to bridge the gap between officers and consumers, facilitating consumer interactions and complaint handling (Furtak, 2021).

PLN itself has several tasks to monitor power and maintain existing electrical infrastructure (Osunde, 2022). For power and voltage monitoring, PLN categorizes customers into two groups based on power consumption: those with low voltage and those with medium voltage (Pandrajū et al., 2022). By grouping these voltage levels, they can monitor the electricity status of each customer, categorized as (Miranda Filho et al., 2016): (1) needs attention, (2) needs repair, or (3) normal. Meanwhile, for the monitoring and maintenance of existing electrical infrastructure, PLN divides the tasks into several areas: customers and power; susut (energy lost), KWH sales, KWH production; saidi (duration of outage), saifi (frequency of outage), recovery time, GGN 100 KMs (interruptions or failures in the electrical network causing disconnection, calculated per 100 km of circuit), GGN reclosure (temporary interruptions in the electrical network); and HPL (addition of electrical poles).

The application created is called the *Sistem Informasi Bersama* (SIBER). SIBER is an information system that contains dashboards for visualizing data related to power

and voltage monitoring used by customers (the AMR "Automatic Meter Reading" dashboard) and the monitoring and maintenance of existing electrical infrastructure (the performance dashboard), all functioning in real-time (Raisin et al., 2020). The data visualization includes a geographic layout (map) to identify the location and electricity status of customers, a pie chart to display the number of statuses in each region, and line and bar charts to compare annual figures.

The method used during the development of the SIBER application is the Scrum framework. Scrum is a framework for developing and managing complex products. It relies on collaboration among Scrum teams to enhance the product, and through an iterative process, it aims to achieve the best final result. Scrum continuously adapts to necessary changes and evolves in line with technological advancements (Shafiee et al., 2020).

Previously, there were application dashboards developed without using a formal development method, as observed in Figure 1.

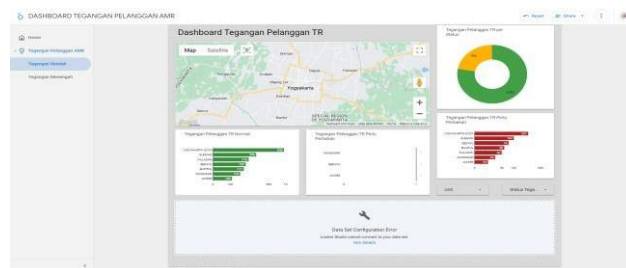


Figure 1. AMR Customer Voltage Dashboard

The dashboard in Figure 1 serves a similar purpose as the SIBER application, namely monitoring power and voltage used by customers. However, this dashboard was not developed to operate in real-time. Additionally, its development was very slow and missed the timeline deadlines. There is also a project at PLN that used the waterfall method, which had different functionality but experienced unstructured development and missed deadlines. This led to delays in implementing important features. The combination of unstructured development methods and the inflexibility of the waterfall method caused PLN projects to face obstacles in achieving goals effectively and efficiently. Therefore, adopting the Scrum method for the PLN UP3 Yogyakarta data visualization dashboard project is expected to address these challenges by providing a framework that is iterative, responsive to change, and ensures project completion according to the specified timeline.

RESEARCH METHODS

The method for developing dashboard applications can be seen in Figure 2. Development begins with setting product goals, creating a product backlog, planning sprints, conducting sprints, producing increments, holding sprint reviews, performing sprint retrospectives, and conducting daily scrums (Cano et al., 2021).

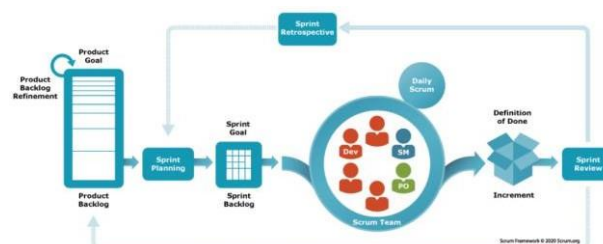


Figure 2. Scrum Method Development Steps

A. Product Goals

The first step in the Scrum methodology is to establish the product goal, which defines the objective of developing the product (Boghossian & David, 2021). The product goal provides a general description of what the development aims to achieve. In the SIBER project, the product goal is set by the product owner, the Assistant Manager of the Planning and Transaction Division of Energy. This target is determined based on the fundamental problems identified during the creation of the application or system and is discussed collaboratively with the team, product owner, and project leader or Scrum Master during meetings.

B. Product Backlog

Next, a list of all the work required for developing the product is created, known as the product backlog. The product backlog is dynamically managed by the product owner and includes all features, changes, and improvements. It is reviewed, updated, and refined periodically, such as during daily sprints or regular meetings, and weekly evaluations are also recommended.

The product backlog identifies the required requirements and organizes them into a to-do list for the development of the SIBER application (Rupp & Singh, 2020). The product owner determines the backlog based on the needs essential for developing the SIBER application, which are defined as features required in each dashboard. Additionally, the product owner prioritizes these features based on their importance and functionality. The prioritized product backlog is then submitted to the team during meetings.

C. Sprint Planning

After creating the product backlog, the team conducts sprint planning. During this stage, the team selects items from the product backlog to work on during the sprint and creates a detailed work plan for completing them (Grapenthin et al., 2015). In this phase, the team discusses with the project leader which features will be addressed in each sprint.

D. Sprint

Sprints are conducted within a limited timeframe as part of the project. For this project, sprints are carried out over a period of 2 months to work on the AMR dashboard and Performance dashboard with the team. The team meets regularly in daily stand-up meetings (daily scrums) to review progress on the application (Henriques & Tanner, 2020). Additionally, task division among team members is organized and monitored by the project leader.

E. Increment

At the end of each sprint, the team must deliver an increment, which is a version of the product that is either created or updated. This increment should be testable and usable by end users, specifically the Assistant Manager of AMR and Performance (Riecken, 2022). The resulting increment is then reviewed with the product owner and project leader during the sprint review.

F. Sprint Review

After completing the sprint, the next step is the sprint review. During this phase, the team examines the increments that have been produced and receives feedback from the product owner and project leader (Cooper & Sommer, 2018). Sprint reviews are held every Friday following the end of the sprint period.

G. Sprint Retrospective

After the sprint review, the next step is the sprint retrospective. This phase involves reflecting on the past sprints, identifying what went well, and pinpointing areas for improvement (Andriyani et al., 2017). The retrospective is conducted with the team, the product owner, and the project leader, focusing on the challenges encountered during the sprint and how to address them.

H. Daily Sprint

Daily meetings, often called daily scrums, are brief sessions to check progress, synchronize activities, and identify any obstacles that might hinder progress (Kalenda et al., 2018). These meetings are held every morning with the team, product owner, and project leader.

RESULTS AND DISCUSSION

A. Scrum Implementation

In its implementation, the Scrum method involves the development stages described in Figure 2, which include:

1. Product Goals

The SIBER application aims to create a real-time dashboard for monitoring power and voltage used by customers (AMR dashboard) and for monitoring and maintaining existing electrical infrastructure (performance dashboard). This goal was established by the product owner, the Assistant Manager of Energy Planning and Transactions at PLN UP3 Yogyakarta.

2. Product Backlog

This section contains the features and requirements needed to develop the SIBER application, along with the priority level of each feature. The list of features is explained in Table 1 and Table 2.

The product backlog is divided into two groups based on the dashboard development targets: the AMR dashboard and the Performance dashboard. This division is made because each dashboard has distinct goals and functionalities. The AMR dashboard is designed for the Energy Transactions division to address monitoring needs for power and electrical voltage used by customers. In contrast, the Performance dashboard is intended for the Planning division to manage the monitoring and maintenance of existing electrical infrastructure.

Priority on the product backlog is determined based on the importance of each feature's functionality, as shown in Table 1 and Table 2. The product owner establishes the priority for each feature, which is then discussed further by the team during meetings.

For the Performance dashboard in Table 2, it is necessary to distribute the content across four pages. Each page focuses on different data categories and visualizations. The first page displays customer power usage; the second page shows energy loss, KWH sales, and production; the third page covers outage duration and recovery time; and the fourth page focuses on the addition of electrical poles and transformers per customer.

The time sections in Table 1 and Table 2 are determined based on the estimated time required, which considers the team's experience, the framework needed for the application, and the time needed to address errors or issues encountered during application testing.

Table 1. AMR Product Backlog Dashboard

No.	Feature	Time	Priority
1.	Dashboard navigation bar	3 days	High
2.	Scorecards	7 days	Intermediate
3.	Customer map	14 days	High
4.	Customer table data	7 days	High

5.	Pie Charts	14 days	High
6.	Filter for scorecards, maps, and pie charts	14 days	Intermediate
7.	Footers	1 day	Low

Table 2. Product Backlog Performance Dashboard

No.	Feature	Time	Priority
1.	Distribution into 4 pages Customers & Power Susut, KWH Sells, KWH Production Saidi, Saifi, Recovery time, GGN 100 KMs, GGN reclosure HPL	1 day	High
2.	Navbar for each page	3 days	High
3.	Scorecard for each page	5 days	Intermediate
4.	Diagram for each page	35 days	High
5.	Filter for scorecard and year	14 days	Intermediate
6.	Footers	1 day	Low

3. Sprint Planning

This section outlines future sprint plans, including time estimates and priorities, as detailed in Table 3 and Table 4. Sprint planning begins with identifying the needs and priority features from the product backlog in collaboration with the project leader or Scrum Master. These features are then managed and allocated based on each team's ability to deliver during each sprint iteration.

The time estimates in Table 3 and Table 4 are based on the backlog estimates and serve as the foundation for sprint planning. The entire development team participates in discussions to ensure that these time estimates are realistic and achievable within the specified sprint duration. Experience and application testing are also considered when determining sprint planning times.

Sprint section selection is informed by the team's previous experience with similar features in different applications, allowing for more accurate time estimation and sprint division. Meanwhile, priority levels are derived from the backlog tables for each dashboard.

Table 3. Sprint Planning I (AMR Dashboard)

Sprints	Feature	Time	Priority
Sprint 1	Dashboard navbar, scorecard, footer	11 days	High, Medium, Low
Sprint 2	Customer map, customer table data	21 days	High, High
Sprint 3	Pie chart, filtering	28 days	High, Medium

Table 4. Sprint Planning II (Performance Dashboard)

Sprints	Feature	Time	Priority
Sprint 1	Page division, dashboard navbar, footer	5 days	High, High, Low
Sprint 2	Scorecard for each page, filtering	19 days	Intermediate, Intermediate

Sprint 3	Diagram for each page	35 days	High
----------	-----------------------	------------	------

4. Sprints

In these stages, the implementation of the upcoming sprints is carried out according to the previously prepared plans. The sprints are detailed in Table 5 and Table 6. During this process, the team begins to develop and test the application based on its features and time targets.

Sprints are divided to break down the work into more manageable segments that can be achieved within a specified timeframe, allowing the team to focus on tasks more specifically. Time estimates and priorities are derived from the product backlog and the plans outlined in the sprint planning.

Table 5. Sprint (AMR Dashboard)

Sprints	Feature	Time	Priority
Sprint 1	Dashboard navigation bar	3 days	High
	Score cards	7 days	Intermediate
	Footers	1 day	Low
Sprint 2	Customer map	14 days	High
	Customer table data	7 days	High
Sprint 3	Pie charts	14 days	High

Table 6. Sprint (Performance Dashboard)

Sprints	Feature	Time	Priority
Sprint 1	Distribution into 4 pages	1 day	High
	Navbar for each page	3 days	High
	Footers	1 day	Low
Sprint 2	Scorecard for each page	5 days	Intermediate
	Filter to scorecard and year	14 days	Intermediate
Sprint 3	Diagram for each page	35 days	High

5. Increments

In the increment stage, the results of each sprint begin to materialize. The outcomes of the work are detailed in Table 7 and Table 8. Each feature from the dashboard will be tested for quality and functionality in collaboration with the product owner and project leader or Scrum Master. The team will document which features have been successfully developed and present these results in future meetings with the product owner and project leader.

The success criteria in Table 7 and Table 8 are based on whether the features meet the needs and requirements specified in the product backlog. Validation and testing with the product owner and project leader ensure that the features are appropriate and free from issues.

Table 7. AMR Dashboard Increment

No.	Feature	Results
1.	Dashboard navigation bar	Success
2.	Scorecards	Success
3.	Customer map	Success

4.	Customer table data	Success
5.	Pie Charts	Success
6.	Filter to scorecards, maps, and pie charts	Success
7.	Footers	Success

Table 8. Performance Dashboard Increment

No.	Feature	Results
1.	Distribution into 4 pages Customers & Power Susust, KWH Sells, KWH Production Saidi, Saifi, REC_TIME, GGN 100 KMs, GGN_REC HPL	Success
2.	Navbar for each page	Success
3.	Score card for each page	Success
4.	Diagram for each page	Success
5.	Filter to score card and year	Success
6.	Footers	Success

6. Sprint Review

Review results from the Assistant Manager of Energy Planning and Transactions at PLN UP3 Yogyakarta include:

- a. All features are functioning well.
- b. Each dashboard operates in real-time and is integrated with the database.
- c. The dashboard design is clear and informative.

Sprint review meetings are held every Friday with the product owner and project leader or Scrum Master. This day was chosen because employees are generally more relaxed than on other weekdays. Feedback and input from these reviews are used to make adjustments in the next sprint.

7. Sprint Retrospective

In these stages, each team provides feedback and evaluations on performance while implementing the Scrum method. This stage concludes at the end of each sprint. The results of the sprint retrospectives show that, although all planned features were successfully completed as expected, there were some issues during implementation. One notable problem was ineffective team collaboration, which led to delays in some tasks. This issue can be addressed by enhancing communication and ensuring greater transparency and clarity of tasks. Sprint retrospectives are conducted with the product owner and project leader or Scrum Master.

8. Daily Sprint

During this project, daily sprints have significantly contributed to the progress of the development team. Notable achievements include the successful completion of the initial project setup and the resolution of server resource access issues. Additionally, the team effectively implemented the selected framework and addressed bugs in each existing view.

B. Data Visualization Dashboard

The final dashboard design, illustrated in Figure 3 and Figure 4, showcases the completed data visualization dashboards. Figure 3 demonstrates that all functions

REFERENCES

- Andriyani, Y., Hoda, R., & Amor, R. (2017). Reflection in agile retrospectives. *Agile Processes in Software Engineering and Extreme Programming: 18th International Conference, XP 2017, Cologne, Germany, May 22-26, 2017, Proceedings* 18, 3–19.
- Boghossian, J., & David, R. J. (2021). Under the umbrella: Goal-derived category construction and product category nesting. *Administrative Science Quarterly*, 66(4), 1084–1129.
- Cano, E. L., García-Camús, J. M., Garzías, J., Moguerza, J. M., & Sánchez, N. N. (2021). A Scrum-based framework for new product development in the non-software industry. *Journal of Engineering and Technology Management*, 61, 101634.
- Cooper, R. G., & Sommer, A. F. (2018). Agile–Stage-Gate for Manufacturers: Changing the Way New Products Are Developed Integrating Agile project management methods into a Stage-Gate system offers both opportunities and challenges. *Research-Technology Management*, 61(2), 17–26.
- Furtak, A. (2021). Strategic Plan for the Travel Mobile App: The NoQ. ISCTE-Instituto Universitario de Lisboa (Portugal).
- Grapenthin, S., Poggel, S., Book, M., & Gruhn, V. (2015). Improving task breakdown comprehensiveness in agile projects with an Interaction Room. *Information and Software Technology*, 67, 254–264.
- Henriques, V., & Tanner, M. (2020). Assessing the Association between Agile Maturity Model Levels and Perceived Project Success. In *SITE 2020: Informing Science+ IT Education Conferences: Online*, 117–156.
- Kalenda, M., Hyna, P., & Rossi, B. (2018). Scaling agile in large organizations: Practices, challenges, and success factors. *Journal of Software: Evolution and Process*, 30(10), e1954.
- Miranda Filho, J., de Carvalho Filho, J. M., Paiva, A. P., de Souza, P. V. G., & Tomasin, S. (2016). A PCA-based approach for substation clustering for voltage sag studies in the Brazilian new energy context. *Electric Power Systems Research*, 136, 31–42.
- Osunde, A. O. (2022). Medium and low voltage distribution grid network composition analysis and review for the North American and European networks. *Technische Hochschule Ingolstadt*.
- Pandraj, T. K. S., Samal, S., Saravanakumar, R., Yaseen, S. M., Nandal, R., & Dhaliya, D. (2022). Advanced metering infrastructure for low voltage distribution system in smart grid based monitoring applications. *Sustainable Computing: Informatics and Systems*, 35, 100691.
- Raisin, S. N., Jamaludin, J., Rahalim, F. M., Mohamad, F. A. J., & Naeem, B. (2020). Cyber-physical system (CPS) application-a review. *REKA ELKOMIKA: Jurnal Pengabdian Kepada Masyarakat*, 1(2), 52–65.
- Riecken, H. (2022). AI in performance management: a game-changing development? *University of Twente*.
- Rupp, C., & Singh, M. (2020). *Scaling Scrum Across Modern Enterprises: Implement Scrum and Lean-Agile techniques across complex products, portfolios, and programs in large organizations*.
- Shafiee, S., Wautelet, Y., Hvam, L., Sandrin, E., & Forza, C. (2020). Scrum versus Rational Unified Process in facing the main challenges of product configuration systems development. *Journal of Systems and Software*, 170, 110732.
- Sjödin, D., Parida, V., Kohtamäki, M., & Wincent, J. (2020). An agile co-creation process for digital servitization: A micro-service innovation approach. *Journal of Business Research*, 112, 478–491.

Copyright holder:

Hestri Apriliani Widowati, Andhik Budi Cahyono (2024)

First publication right:

Journal of Social Science

This article is licensed under:

