Development of Workshop Modules Using Agile Scrum Models in ERP Systems at PT Surya Data Infokreasi

Risa Irmawati[1], Galih Hendra Wibowo[2], Junaedi Adi Prasetyo[3], Sepyan Purnama Kristanto[4], Lutfi Hakim[5]

[1,2,3,4,5]Software Engineering Technology Study Program, Department of Infromatics Engineering, State Polytechnic of Banyuwangi, 68461, Indonesia.

[1]risairma21@gmail.com, [2]galih@poliwangi.ac.id, [3]junaedi.prasetyo@poliwangi.ac.id, [4]sepyan@poliwangi.ac.id, [5]lutfi@poliwangi.ac.id

***Abstract*— This research aims to develop a workshop module using the Agile Scrum development model in the web-based ERP system at PT. Surya Data Infokreasi. PT ABC, a frozen goods shipping company, requires effective management of vehicle assets, including spare parts, tires, and cooling boxes. Currently, their ERP system lacks an adequate module for these needs. Agile Scrum was chosen for its suitability in developing large-scale, modular ERP systems. The workshop module includes features for service requests, spare parts requests, truck spare parts usage, and truck spare parts reports. Functionality testing was conducted using the User Acceptance Testing (UAT) method. Results show that out of 43 test cases, 36 were successful (84%), 6 failed (14%), and 1 passed with exceptions (2%). The system's success in meeting user needs reached 84%, indicating that the workshop module can help PT ABC manage vehicle assets and workshop operations more effectively. The conclusion is that Agile Scrum proved effective in developing a system that meets specific user needs. The implementation of this module is expected to increase the efficiency of workshop management at PT ABC and provide accurate data for decision-making regarding vehicle assets.**

Keywords*— *Agile Scrum, ERP System, Laravel, UAT, Workshop Modul's.

I. INTRODUCTION

Organizations or businesses that can generate good quality information most likely will have to change the processes of the business so that they can match the steps that must be taken or what goes on to reach the objectives set by the company. That means they may have to change the processes per the business aims. This process change is a requirement of the emerging technology in this information age[1]. The development of information technology impacts business growth because the advancements in information technology will facilitate the business process flow in companies. The role of information technology is very evident with the implementation of information systems used by companies in their business operations [2]. Enterprise Resource Planning (ERP) systems are among the most widely adopted systems by organizations. These software solutions aim to integrate and unify various departments and processes within a company into a single cohesive system. Typically, an ERP system encompasses multiple functionalities such as finance, accounting, human resources (HR), purchasing, manufacturing, and sales. These functionalities are often referred to as modules, and within each module, there are specific features tailored to that particular function [3].

PT Surya Data Infokreasi is an IT consulting company specializing in the garment industry. With its expertise and experience in custom software for each customer, PT Surya Data Infokreasi developed an ERP (Enterprise Resource Planning) system named XMOR. ERP systems are widely used by companies and organizations to enhance the efficiency of resource planning and management and to monitor business operations more effectively. These multi-functional platforms integrate critical areas like product planning, component procurement, inventory control, product distribution, and order tracking, providing a holistic view of operations and boosting overall efficiency [4]. ERP is an information system for manufacturing or service companies that plays a crucial role in integrating and automating business processes related to the operational, production, or distribution aspects of the company. In today's digital landscape, companies have a plethora of ERP-based applications to select from, including OpenERP or Odoo, Oracle, SAP, IFS, Peoplesoft, and numerous others [5]. However, PT. Surya Data Infokreasi endeavors to innovate by creating a web-based ERP system environment named XMOR.

PT Surya Data Infokreasi's XMOR application provides a comprehensive suite of basic modules for various business functions. These modules include accounting management, purchasing management, manufacturing management, human resource management, sales management, expedition management, warehouse management, and inventory management. However, the current modules do not yet meet the needs of one PT Surya Data Inokreasi's clients, PT ABC a frozen goods delivery service company.

As a company specializing in frozen goods delivery services, PT ABC possesses numerous vehicle assets used for operational activities. Given this situation, PT ABC requires effective management of its vehicle assets, including the management of spare parts, tires, and cooling boxes. The Operational Team at PT ABC has reported that the high frequency of replacing spare parts, tires, and cooling boxes during vehicle maintenance often results in confusing

warehouse stock reports. Currently, service records are not maintained, making it impossible for PT ABC to compare the frequency of vehicle deliveries with the service performed, which is crucial for determining the expenses and revenues of each vehicle over a specific period. To address these issues, it is necessary to add a workshop module to control workshop management at PT ABC. Research conducted [6], has demonstrated that the Agile Scrum software development model is suiTable for large and modular projects like ERP systems because it uses an iterative method and can save costs. Besides that, Scrum is a collaborative work approach that focuses on team collaboration and productivity, adapting within the agile methodology.

Scrum divides problems or system development into several Sprints, allowing the development to focus on achieving desired targets. Additionally, this method provides an opportunity to evaluate the system if issues arise through Sprint Reviews. Implementing this method supports problem-solving in the system development process with a team composed of new members [7]. Because Scrum is agile, this framework is adaptive and incremental towards change [8].

Therefore, this study aims to implement the Agile Scrum development model in the development of the workshop module in the XMOR application of PT. Surya Data Infokreasi. The development will also utilize Laravel technology with a PostgreSQL database. Laravel was chosen because researchers found that its built-in libraries are comprehensive enough to handle web development projects from small to medium scale [9]. It is expected that this development will lead to significant improvements in workshop management, impacting stock reports, and helping PT. ABC compare the expenses and revenues of each vehicle unit, thereby assisting the company in determining the profitability of its vehicles.

II. METHOD

A. System Overview

The workshop module is an addition to the existing modules in the XMOR system. In developing the workshop module, it is important to consider its seamless integration with the existing modules, such as the inventory and expedition modules. This integration ensures that the workshop operations are synchronized with inventory management for accurate tracking of spare parts and tools. Additionally, aligning with the expedition module facilitates efficient scheduling and maintenance of vehicles, thereby enhancing the overall operational efficiency of the company. The integration will all departments are up-to-date with the latest information, which is crucial for decision-making and process optimization enable real-time data sharing between modules, ensuring that Fig. 1, illustrates an overview of the RP system being develop. Firstly, regarding the inventory module, the workshop module will directly interact with the available inventory data. This means that users of the workshop module will have access to current inventory information, such as available spare parts or materials. For instance, when a mechanic needs an oil filter spare part, the workshop manager will request the spare part from the warehouse. The warehouse will then check the stock, and if the part is available, the request will be confirmed or approved.

Secondly, the workshop module will be connected to the expedition module. This is because there needs to be coordination between the workshop activities and the expedition activities related to asset maintenance (trucks). The data for trucks in the expedition module will be linked with the data in the workshop module. For example, when scheduling a delivery, users must select a truck that is not currently undergoing repairs. Additionally, the linkage can be used to compare the frequency of truck deliveries with the frequency of truck maintenance or repairs.

By considering these connections, the development of the workshop module must ensure proper integration with the existing modules in the XMOR system. This ensures that the workshop module does not operate in isolation but becomes an integral part of the overall system, supporting the company's operations, as shown in Figure 1.

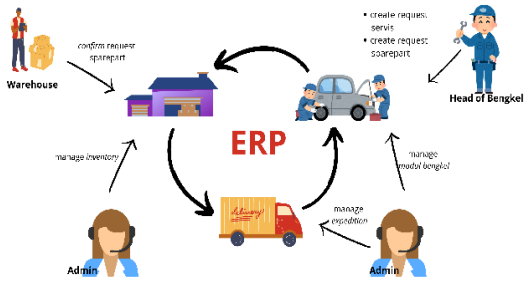

Figure 1. System Overview

B. Research Stages

In developing the workshop module for the XMOR application, researchers naturally refer to software development models to structure and streamline the work process. The research stages are divided into several steps in this process. These stages include problem identification, literature review, data collection, and the application of the Agile Scrum development model, which allows researchers to systematically follow the established development process. The Agile Scrum development model was chosen because it relies on an incremental approach to software specification, development, and delivery to customers. This method is most suiTable for the development of applications where system specifications change rapidly during the development system process [10]. This ongoing delivery provides opportunities for regular user feedback, which is critical in ensuring that the final product aligns closely with user needs and expectations. Moreover, the incremental nature of Agile Scrum helps manage the complexity of the project by focusing on small, manageable tasks that can be completed within short time frames.

1) Data Collection

The initial step in the data collection process involves interviews. This data collection technique entails direct questioning of stakeholders at PT ABC regarding their current or ongoing business processes. These interviews facilitate a

deeper understanding of the company's needs and challenges. Throughout the interview process, stakeholders involved in operating the system are identified. The actors and their interests within the system are detailed in the Table I:

TABLE I
ACTOR DEFINITION

Aktor	Description
Admin	Super admin who has access to all system modules.
Head of Workshop	The person who submits service requests and requests for spare parts or components.
Warehouse	The individual who manages inventory in the warehouse and has the authority to confirm requests for spare parts or components.

2) Implementation of Scrum

Product Backlog

The first stage in Agile Scrum is the product backlog, which consists of a list of features or requirements desired by the users. The backlog serves as the central repository for all knowledge related to project requirements and is the sole authorized source for defining the work to be undertaken. It contains a prioritized list of features or technical tasks that the team deems necessary and sufficient at any given time to complete the project. As the project progresses and the team acquires more insights, changes to the backlog are expected. The backlog typically includes bugs, technical tasks, and efforts aimed at acquiring new knowledge [11]. The product backlog is compiled by the product owner and will be worked on by the development team. This stage ensures that all desired features are accounted for and prioritized accordingly. The product backlog is attached to Table II.

TABLE II
PODUCT BACKLOG SAMPLE

Menu	Description
Master Truck	As an admin, I can view, add, and edit truck data to facilitate the management of truck master data.
Master Driver	As an admin, I can view, add, and edit driver data to facilitate the management of driver master data.
….	

SPRINT PLANNING

PROJECT NAME	PROJECT MANAGER	START DATE	END DATE	OVERALL PROGRESS	PROJECT DELIVERABLE
ABC EXPRESS	ARIS SUSANTO	02/01	04/01		SCOPE STATEMENT

At Risk?	TASK NAME	RESPONSIBLE	DOD	START DATE	END DATE	DURATION in days	PRIORITY	COMMENTS
	SPRINT 1	Risa	Menyelesaikan master data	01/16	02/15	31		
Yes	Feature 1	Risa		01/16	02/02	18	high	
No	Feature 2	Risa		01/31	02/05	6	Medium	
	Feature 3	Febe		02/05	02/15	11	Low	
	SPRINT 2	Risa	Finished transaction	01/22	02/25	35		
	Feature 1	Febe		01/22	01/29	8		
	.....	.....	...	....	....	....		.....

Figure 2. Sprint Planning Sample

After creating the product backlog, the team proceeds with sprint planning by dividing the work from the product backlog into several sprints. Sprint planning serves to define the goals to be achieved in each sprint [8]. Fig. 2 provides a sample of the sprint planning process to be conducted. In this sample, the first sprint specifies which features will be addressed, who is responsible for each, the definition of done, estimated completion times, and the priority of each task.

a. Use Case

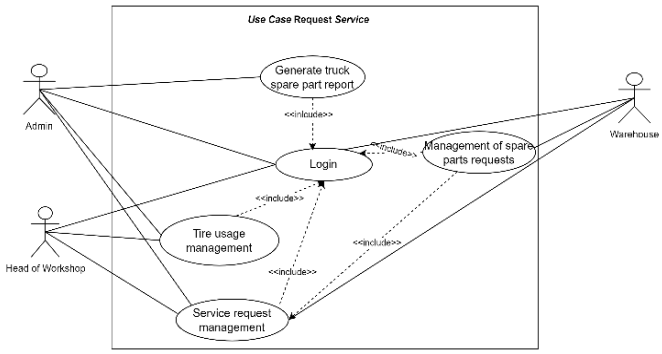

Figure 3. Use Case Request Service

In Fig. 3 is a use case diagram as a basis to build system. Use Case serving as a foundational blueprint for building the system. This diagram is meticulously designed to assist software developers in understanding the complex interactions between users and the system, providing a clear visual representation of how different components and users interact within the context of the system's functionality [12].

Use case diagrams play a crucial role in the early stages of system development, as they outline the essential actions that users can perform and how these actions are linked to specific system processes. This Use Case diagram is created for the development of the XMOR workshop module, involving three primary actors: admin, warehouse and workshop manager. This use case was developed by the team based on the requirements identified during the data collection phase. Detailed explanation of the service request use cases is provided in Tables III, IV, V, VI, and VII, offering a comprehensive view of each interaction and functionality within the system built in this research.

TABLE III
REQUEST SERVICE MANAGEMENT

Use Case	**Service Request Management**
Actor	Admin & Head of Workhosp
Description	The admin and workshop manager can manage service request data, such as creating service requests, updating service statuses, handling service documents, and managing spare part requests.
Pre-Condition	Once logged into the system as an admin or workshop manager:
Normal Course	1. The actor navigates to the service request menu. 2. The actor can view service request data. 3. The actor can create service requests. 4. The actor can create spare part requests. 5. The actor can update the status of service requests.

Alternative Course	-
Post-Condition	Data will be stored in the system according to the information input by the actor.

TABLE IV
TIRE USAGE MANAGEMENT USE CASE

Use Case	**Tire usage management**
Actor	Admin and workshop manager
Description	The admin and workshop manager can manage tire usage such as adding, editing, and deleting tire usage data.
Pre-Condition	The actor has logged into the system as an admin or workshop manager:
Normal Course	1. The actor navigates to the tire usage menu. 2. The actor can view tire usage data. 3. The actor can modify tire usage data.
Alternative Course	-
Post-Condition	Data will be stored in the system according to the information input by the actor.

TABLE V
MANAGEMENT SPARE PART REQUEST

Use Case	**Management of spare part request**
Actor	Admin and workshop manager
Description	The warehouse can confirm requests for spare parts made by the admin or workshop manager.
Pre-Condition	The actor has logged into the system as an admin.
Normal Course	1. The actor navigates to the spare parts request menu. 2. The actor can view spare parts request data. 3. The actor can confirm spare parts requests.
Alternative Course	-
Post-Condition	The system will store data based on what the user enters.

TABLE VI
GENERATE TRUCK SPARE PART REPORT USE CASE

Use Case	**Generate truck spare part report**
Actor	Admin
Description	The admin can generate a vehicle expense report in the system by inputting the transaction period.
Pre-Condition	The actor has logged into the system as an admin.
Normal Course	1. The actor navigates to the generate vehicle expense report menu. 2. The actor can input the transaction dates. 3. The system displays the vehicle expense report for the inputted period. 4. The actor can download the report.
Alternative Course	-
Post-Condition	The actor can download the report in Excel format.

b. Activity Diagram

An activity diagram illustrates the flow of functionality within an information system. It comprehensively defines where the workflow starts, where it stops, what activities occur during the workflow, and the sequence of those activities. Activity diagrams also provide an approach for modeling parallel processes. For those familiar with traditional structural analysis and design, this diagram integrates the foundational ideas of data flow diagrams and system flow diagrams [13].

Fig. 4 depicts the Activity Diagram for User Login, illustrating the step-by-step process a user follows to access the system. Initially, the user, who could be an admin, warehouse staff, or workshop manager, opens the login page. Upon doing so, the system promptly displays the login form, prompting the user to enter their email and password. Once the credentials are submitted, the system verifies them. If the login is successful, the system redirects the user to the appropriate admin dashboard. However, if the login attempt fails, whether due to incorrect credentials or other issues, the system redirects the user back to the login page, enabling them to retry the login process. This entire sequence is crucial for maintaining the security of the system, ensuring that only users with the correct credentials can gain access, thereby protecting the system from unauthorized entry and preserving the confidentiality and integrity of the system's data and functions.

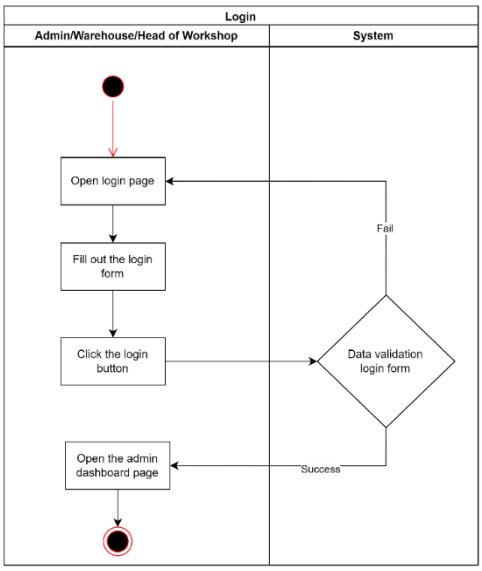

Figure 4. Activity Diagram Login

Fig.5 depicts the Activity Diagram for the Service Request process, providing a detailed overview of the steps involved. Initially, the user, who could be either an admin or a workshop manager, logs into the system. After successfully logging in, the user can go to the service request menu, which is promptly displayed by the system. To make request service, user can clicks "add data" button, prompting the system to show the form for adding a new service request. The user proceeds to fill out this form with the necessary information.

If the service request involves the need for spare parts, the system will display an additional form for adding a spare part request. Otherwise, the system will save the service request data directly. Once the user completes the form, they click the "save" button, and the system saves the newly added service request data, returning the user to the service request menu index page which display the data recently added and previously existing data.

Should the user need to modify an existing service request, they can click the "edit data" button. Once this button is clicked, the system promptly displays a form pre-filled with the current details of the service request. This form allows the user to review the existing information and make any necessary

changes, such as updating the service description, altering deadlines, or adding additional notes or requirements. The user can make the necessary changes, and once the data is appropriate, click the "save" button to confirm the updates. The system then processes the new information, replacing the old data with the revised details in the database. This ensures that the most current information is always reflected in the system, preventing any potential misunderstandings or errors that could arise from outdated data. This process enables the admin and workshop manager to efficiently manage service requests within the system.

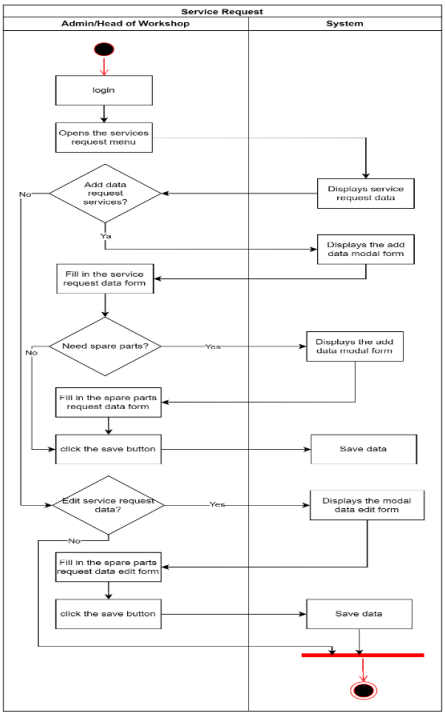

Figure 1. Activity Diagram Management Request Service

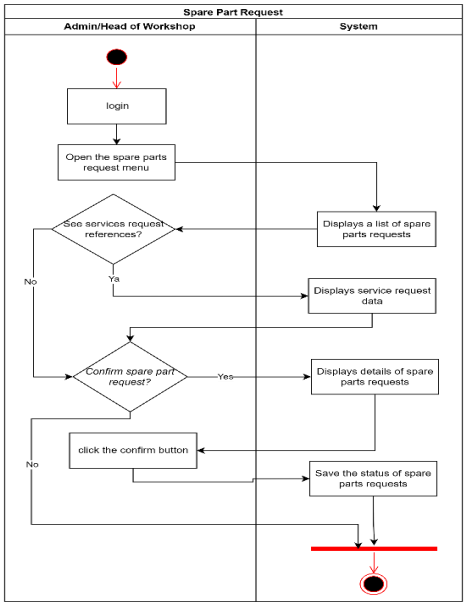

Figure 2. Activity Diagram Spare Part Request

Fig. 6 illustrates the Activity Diagram for the Spare Part Request process, outlining the sequence of actions taken by users. The process begins when the user, who could be either an admin or warehouse staff, logs into the system. Once successfully logged in, the user navigates to the Spare Part Request menu, where they can manage and review all current requests in a centralized interface. At this juncture, the user has the option to view related service request references, which is a critical step for cross-referencing and ensuring that the spare parts being requested are aligned with the specific needs of ongoing service tasks. If the user opts to view these references, the system promptly displays the relevant service request data, providing a comprehensive overview that aids in decision-making.

After thoroughly reviewing the necessary details and ensuring that everything aligns with the requirements, the user proceeds to the next step, where they decide whether to confirm the spare part request. If the decision is to confirm, the system then displays the detailed information of the spare part request one final time, giving the user an opportunity to verify all aspects before finalizing. Once satisfied, the user clicks the "confirm" button, signaling the system to save the updated status of the request. This action ensures that the request is accurately recorded in the system's database, making it accessible for future reference and further processing. By following this structured and methodical approach, the system enables the Admin and Head of Workshop to manage spare parts requests efficiently, reducing the likelihood of errors, maintaining up-to-date records, and ensuring that all necessary parts are tracked and processed in a timely manner. This process not only streamlines operations but also enhances the reliability and effectiveness of the overall spare parts management within the organization.

Figure 3. Activity Diagram Tire Usage

Fig. 7 illustrates the Activity Diagram for tire usage, detailing the steps involved in managing tire usage transactions within the system. The process begins with the user, who could be either an admin or workshop manager, logging into the system. After successfully logging in, the user navigates to the tire usage menu, where the system displays a list of existing tire usage transactions.

To add a new tire usage transaction, the user clicks the "Add Data" button, prompting the system to display a form for inputting the relevant details. The user then fills out this form with the necessary information and clicks "Save." The system processes this input and saves the new tire usage transaction data, ensuring it is recorded in the system.

If an existing tire usage transaction needs to be modified, the user can click the "Edit Data" button. The system will then display the form for editing the selected transaction. After making the necessary changes, the user clicks "Save," and the system updates the transaction data accordingly.

Finally, if the user needs to remove a tire usage transaction from the system, they can click the "Delete Data" button. The system will then delete the selected transaction, ensuring that the tire usage records remain accurate and up to date. This process enables the admin and workshop manager to efficiently manage tire usage transactions, maintaining accurate records and ensuring the effective use of resources.

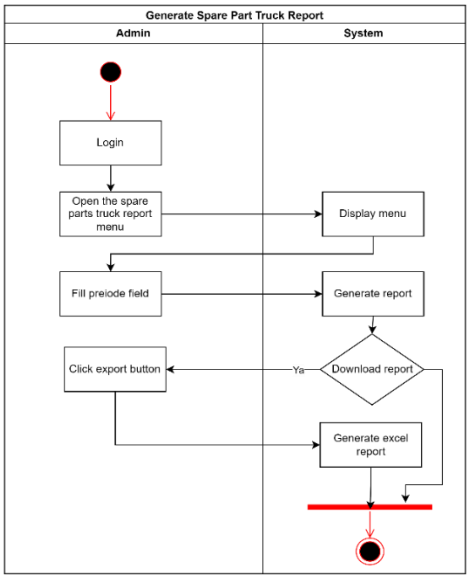

Figure 4. Acitvity Diagram Generate Spare Part Truck Report

Fig. 8 illustrates the Activity Diagram for generating a truck spare part report, outlining the steps taken by the admin to obtain and download this report. The process begins with the admin logging into the system. Once logged in, the admin navigates to the "Generate Truck Spare Part Report" menu, which the system then displays. The admin is then prompted to input the desired period for the report. After entering this information, the system generates and displays the truck spare part report corresponding to the specified period. If the admin wishes to save the report, they click the "Download" button. The system then processes the request, generates the spare part stock report, and initiates the download, providing the admin with a detailed report for their records. This streamlined process allows the admin to efficiently generate and access critical information regarding truck spare part inventories.

Sprint Backlog

The sprint backlog is the phase where the team addresses the requirements listed in the product backlog. During the sprint backlog, the team engages in coding, daily scrums, sprint reviews, sprint retrospectives, and system testing to fulfill these requirements.

The development process utilizes the Laravel framework as the software development framework. Additionally, DataTables are employed to create dynamic and flexible Table displays. DataTables is a highly versatile JavaScript library that enhances the accessibility of data within HTML Tables. Furthermore, AJAX and jQuery are utilized for sending data to and receiving data from the server.

Software testing has thus become a field of its own and can be quite a complex process with many steps such as unit-, functional-, system-, integration- and user acceptance testing [14]. The software testing is carried out using User Acceptance Testing (UAT) to ensure the system meets the predetermined requirements (product backlog). This testing focuses on the functional aspects of the system, verifying whether the system operates according to its intended functionalities and produces the expected results [15]. The testing format is illustrated in Table VII as follows:

TABLE VII
UAT TESTING FORMAT

Modul	Feature	Scenario	Expectation	Status
Workshop	Master Mechanic	Input a non-image file in the photo field.	Display a warning: "Input must be an image."	Not Pass
Workshop	Master Mechanic	Input a numeric value in the Place of Birth field.	Display an error message: "Input cannot be a number."	Not Pass
Workshop	Master Mechanic	Click the add data button in the document Table.	Successfully opened the add document modal.	Pass
Workshop	Master Mechanic	Input a file that is not in the document format.	Input failed and displayed an error message.	Pass
Workshop	Master Mechanic	Click the save button in the add and edit mechanic data modal without filling in the mandatory fields.	Failed to save changes.	Pass

III. RESULTS AND DISCUSSION

A. Result

1) Product Backlog

In Scrum methodology, the product backlog is a list of user desires (features) that will be realized and fulfilled in a project. When creating a product backlog, the selection of features is

prioritized by the product owner. This prioritization is crucial as it ensures that the most valuable and impactful features are addressed first, aligning with the project's goals and user needs. The prioritization process often involves evaluating the business value, user impact, and technical feasibility of each feature. The available features can be seen in the following Table VIII:

TABLE VIII
PRODUCT BACKLOG RESULT

No	Menu	Description	Priority
1.	Designing the system architecture (M-001)	Create UML diagrams as a reference for system development.	High
2.	Master mechanic (M-002)	As an admin, I can view, add, and edit driver data to facilitate the management of mechanic master data.	High
3.	Spare part request(T-003)	• As an admin and workshop manager, I can create spare part requests so that the warehouse personnel know which spare parts are needed. • As warehouse personnel, I can confirm spare part requests so that spare parts can be issued from stock.	High
4.	Tire usage (T-004)	As an admin and workshop manager, I can manage tire usage to track tire movement for each truck.	Middle
5.	Service reaquest (T-004)	As an admin and workshop manager, I can manage service requests to record service details for each truck unit.	High
6.	*Report spare part truct* (T-005)	As an **admin** and **operations** personnel, I can download the truck spare part report to track spare part usage per truck.	High

2) Sprint Planning

This phase involves determining the sprints based on the product backlog outlined in Table VII. Total of four sprints will be created, taking into account the backlog features, tasks, and estimated time (days) in accordance with Scrum rules, ultimately forming a sprint backlog. Sprint planning will be conducted at the beginning of each sprint to plan the tasks that will be undertaken during that sprint. The outcome of sprint planning is referred to as the sprint backlog. The following sprint planning can see at the https://shorturl.at/tnD6t.

3) Sprint Review

Sprint Review is a meeting held at the end of each sprint in the Scrum methodology. The main objective of the Sprint Review is to examine the work results completed during the sprint and obtain feedback from stakeholders. Are the outcomes of the Sprint Review for a total of three sprints.

a. Sprint 1

In Fig. 9 , the master mechanic index page is displayed. This page is designed to serve as the central hub of information that integrates the master mechanics' data involved in the system. The available information includes detailed profiles of each mechanic, encompassing their qualifications, experience, and specialties in specific fields. Furthermore, the master mechanic index page include features such as search filters, and sorting. It can allow users to easily find mechanics based on specific criteria, like years of join to company, name, deparment, etc.

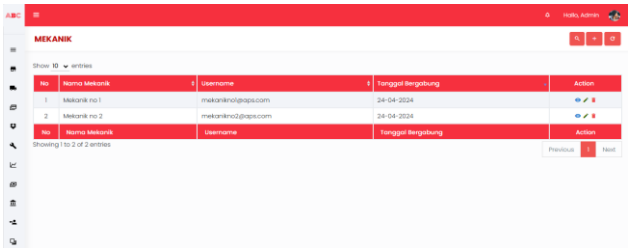

Figure 5. Index Master Mechanic Page

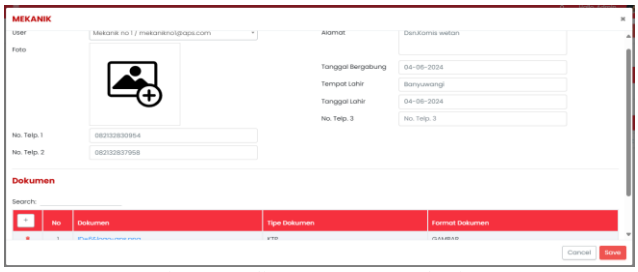

Figure 6. Edit Data Master Mechanic Page

Fig. 10, the master mechanic data editing page developed by the team is shown. This page is designed to facilitate the management and updating of detailed information for each mechanic within the system. Users can easily access and modify information such as names, contacts, qualifications, and work histories of mechanics on this page. Additional features like uploading certification documents and performance notes are also provided to ensure that stored data remains current and accurate.

b. Sprint 2

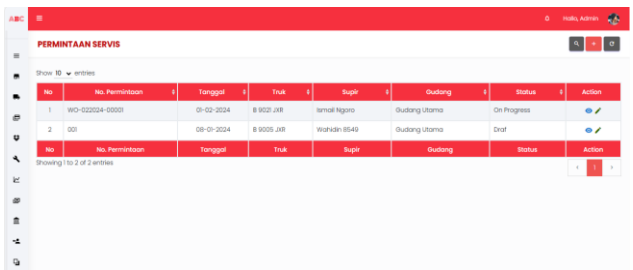

Figure 7. Index Service Request Page

In Fig. 11, the service request index page developed by the team is depicted. On this page, users can view a list of service requests, including information such as request number, submission date, current status, problem description, and assigned mechanics. Search and filtering features are also provided to facilitate users in finding and managing specific requests based on desired criteria, like service number, service date, truck, driver,warehouse , and status request services. For instance, users can search by service number to find a particular request instantly, or filter by service date to view requests scheduled within a specific timeframe. The ability to filter by truck allows users to focus on service requests related to a particular vehicle, while filtering by driver enables them to track requests associated with specific personnel. The warehouse filter is particularly useful for managing requests based on the location of inventory or resources, ensuring that service tasks are aligned with available supplies.

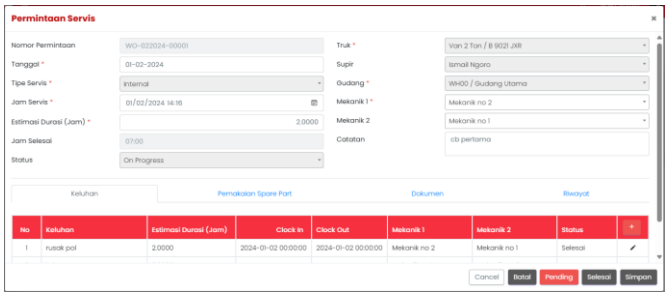

Figure 12. Edit Service Request Page

In Fig. 12, show the service request editing page. When this page is opened, the selected data ID is sent to the server to retrieve information based on the ID. The main information or header on this page includes the request number, date, service type, service time, status, the truck scheduled for service, and the mechanic handling the service. The header, there are several tabs such as the complaint tab, spare parts usage tab, documents tab, and history tab.

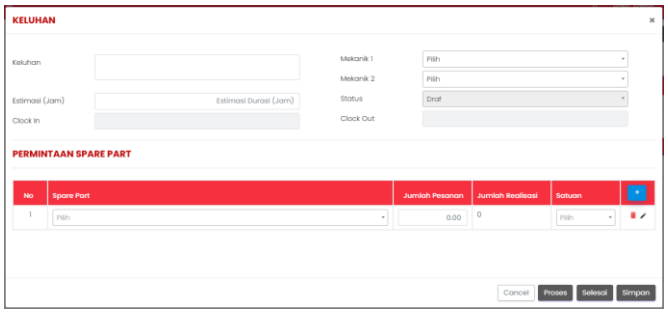

Figure 8. Form Input Issue and Spare Part Request

When the complaint tab is opened and the user clicks the 'add data' button on the complaint Table, the system will display a complaint modal like in Fig. 13, which contains information about the complaint and the spare part request that will be used to address the complaint. When the user clicks the 'add data' button on the spare part request Table, the system will automatically generate a row for the user to input the desired spare part, the quantity requested, and its unit.

Figure 9. Tab Spare Part Usage

When the user opens the spare part usage tab, the system will display a view similar to Fig. 14, which contains two additional tabs within it. The "Incoming Goods" tab stores information about spare parts used by the truck during the ongoing service process. Meanwhile, the "Outgoing Goods" tab records spare parts issued by the truck during the service process. In this tab, users can only view data and cannot perform operations to insert or update data.

c. Sprint 3

In the third sprint, we successfully developed the tire usage menu and the truck spare part report menu. The results can be seen in Fig. 15, Fig. 16, and Fig. 17:

Figure 10. Index Tire Usage Page

Fig. 15 displays the index view of the tire usage menu. In this menu, users can manage tire usage, such as replacing tires on vehicles. It provides features that facilitate users in recording and monitoring each tire replacement, ensuring that all tire usage-related information is well-organized. The menu is also equipped with a search feature, allowing users to quickly find the necessary information. Stored data includes transaction number, date, truck details, total kilometers, and status.

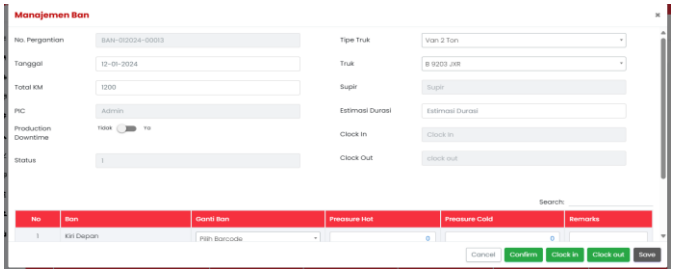

Figure 11. Edit Tata Tire Usage

In Fig 16, we see the edit page for tire usage data. This page is accessed when users click the pencil icon in Fig. 15. The system saves the ID and sends it to the server side to initiate a

search based on that ID. Users are presented with header information such as the change number, date, total kilometers, responsible person (PIC), status, truck type, the specific truck, and the estimated time required. Once the user selects the desired truck type, the system automatically generates rows in the detail Table corresponding to the number of tires for that truck type selected.

Once the data is retrieved, the user is presented with a comprehensive header section that encapsulates all critical information related to the tire usage entry. This header includes the change number, which uniquely identifies the transaction or update event, and the date of the tire usage or change event, providing a clear reference point in time. Additionally, it displays the total kilometers the tires have been in use, which is vital for tracking wear and tear and planning future maintenance. The age also includethe name of the responsible person (PIC - Person in Charge), ensuring accountability and clarity regarding who handled the tire data. The status of the tire usage record, whether pending, completed, or in progress, is also shown, allowing users to understand the current state of the entry at a glance.

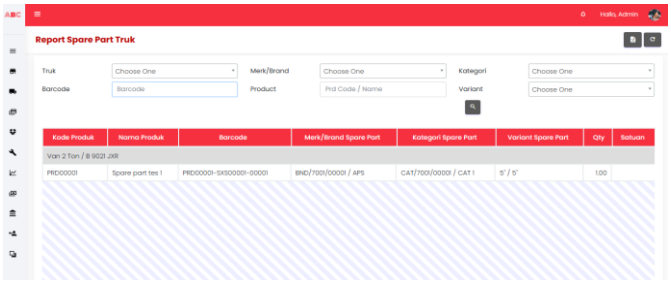

Figure 12. Report Spare Part Truck Menu

The truck spare part report menu is shown in Fig 17. This menu serves as an information hub for transaction data in the workshop module. The displayed report contains information on the usage of spare parts for each truck over a specified period. Users can filter the report based on the data in the header, such as truck, spare part barcode, brand, spare part category, and spare part variant. After obtaining the desired results, users can download the report in Excel format. With this menu, PT ABC can easily review the expenses for each truck, facilitating asset sales decision-making.

d. Sprint 4

In In the final sprint of the project, the testing team meticulously conducted a series of tests on the developed system, marking a critical phase in the software development life cycle. System testing plays an indispensable role in this process, serving as a comprehensive evaluation designed to verify that the system or application functions as intended and meets both the specific needs of the users and the established quality standards. This phase is essential in identifying and rectifying any potential issues before the system is released for broader use.

One of the key components of this stage is User Acceptance Testing (UAT). UAT is particularly vital because it shifts the focus from technical specifications to the end-user experience, ensuring that the system not only works technically but also resonates with the users' expectations and operational requirements. Through UAT, the testing team collaborates closely with actual users to validate that the system's functionalities are practical, intuitive, and aligned with real-world workflows. This form of testing acts as the final checkpoint before the system is deployed on a larger scale, providing a safety net that helps prevent costly errors or user dissatisfaction after launch.

Moreover, system testing, and UAT in particular, serve as a crucial bridge between the development team and the end-users. While the technical teams focus on coding, bug fixing, and performance optimization, UAT ensures that the final product is not only technically sound but also user-friendly and aligned with the business objectives. This alignment is critical because it ensures that the solution delivered by the development team truly addresses the needs it was designed to solve, offering tangible benefits and enhancing overall user satisfaction.

The documentation of the system testing process, which includes detailed reports and findings, is vital for maintaining transparency and for future reference. It provides a record of what was tested, the issues that were found, and how they were resolved, serving as a valuable resource for continuous improvement and future development cycles. For those interested in a deeper dive into the specifics of this testing phase, including the methodologies used and the results obtained, you can access the detailed documentation of system testing through the following link: https://shorturl.at/V30sO.

Based on the testing results, a calculation of the percentage of the total test cases or testing scenarios was obtained. The results of the system testing calculation can be seen in the following graph on Fig. 18 :

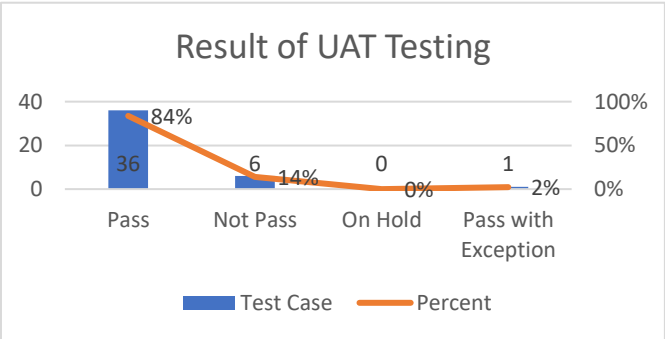

Figure 13. Result of UAT Testing

The results of using the Agile Scrum method in developing this ERP system are documented in the Product Delivery Analysis Table. The Table analyzes the outcomes of completed sprints based on the Product Backlog. Based on the results of the scrum process, which consists of 4 sprints, the percentage of product delivery during each sprint is as follows at the Fig. 19.

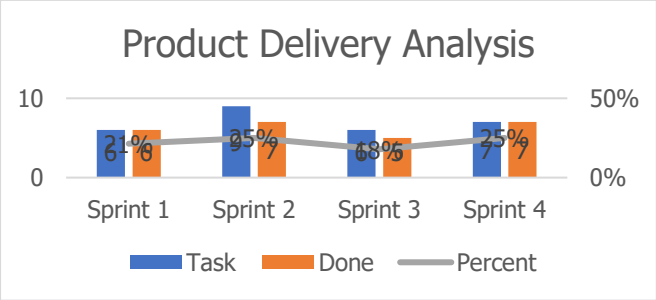

Figure 14. Product Delivery Analysis

Based on the Product Delivery Analysis chart on Fig. 19, it indicates a high success rate in completing features. There were setbacks in several sprints, notably in sprint two, where the total planned work was nine tasks but only seven were accomplished. This means that the two unfinished tasks were carried over to the next sprint, namely sprint three. Barriers can be identified by reviewing the sprint retrospective for problematic sprints. Furthermore, sprint three also experienced a setback with one task incomplete, which was then added to sprint four. Looking at the Product Delivery Analysis chart, the sprint success rate remains relatively good, with the largest delay being two tasks per sprint.

IV. CONCLUSION

This research commenced with the development of a workshop module within the ERP system at PT ABC. The development was carried out using the agile Scrum methodology as a software development approach. This methodology was chosen for its ability to accommodate changes and deliver measurable results in a short period. Technologies employed in this study included the Laravel framework for application development and PostgreSQL as the database management system. The study comprised four sprints, each lasting two weeks. During implementation, the second sprint experienced delays in task completion. In the sprint planning phase, the second sprint had nine tasks that needed completion. However, two of these tasks were not finished on time, necessitating their inclusion in the third sprint. Upon completing the system development process, the system underwent testing using the User Acceptance Testing (UAT) method. During this phase, comprehensive testing was conducted with a total of 43 test cases. Out of these, 39 test cases passed successfully, indicating that the related functions operated as expected. However, four test cases did not pass, indicating issues or deficiencies that need to be addressed before the system can be fully utilized by users.

REFERENCES

[1] E. P. Saflembolo and M. N. N. Sitokdana, "Penerapan SAP Untuk Mendukung Invoice Travel Agent Pada PT Freeport Indonesia," Jatisi, vol. 9, no. 4, pp. 3569–3578, 2022, [Online]. Available: http://jurnal.mdp.ac.id

[2] H. Shavira et al., "Analisis Bisnis Proses Dan Implementasi Enterprise Resource Planning (ERP) Pada Batik Srikandi Banyuwangi," vol. 10, no. 2, pp. 414–428, 2023, [Online]. Available: http://jurnal.mdp.ac.id

[3] T. Febrianto, D. Soediantono, S. Staf, K. Tni, and A. Laut, "Enterprise Resource Planning (ERP) and Implementation Suggestion to the Defense Industry: A Literature Review," Journal of Industrial Engineering & Management Research, vol. 3, no. 3, pp. 2722–8878, 2022, [Online]. Available: http://www.jiemar.org

[4] I. G. Susrama, Sugiarto, and W. Agustiono, "Buku Ajar Enterprise Resource Planning (ERP)," no. February, pp. 1–168, 2022, [Online]. Available: https://www.researchgate.net/publication/358387687

[5] Y. E. Demilda, A. Arvianto, and Z. F. Rosyada, "Implementasi Software Odoo dengan Menggunakan Modul Accounting, Inventory, Purchase, dan Point Of Sales Pada Toko Al Hikmah Mart (Ah Mart) Di Bogor Jawa Barat," Industrial Engineering Online Journal, vol. 11, no. 4, pp. 343–354, 2022, [Online]. Available: https://ejournal3.undip.ac.id/index.php/ieoj/article/view/35967

[6] A. Teresia and Y. L. B. Nugraheni, "Analysis of Enterprise Resource Planning (ERP) Implementation: Institutional Work and Institutional Logics Perspectives," Tjyybjb.Ac.Cn, vol. 20, no. 2, pp. 58–66, 2022, [Online]. Available: http://117.74.115.107/index.php/jemasi/article/view/537

[7] F. A. Dzaky and D. Kurniawan, "Implementasi Metode Agile Framework Scrum dalam Pengembangan Sistem Informasi Manajemen Aset Terpadu Universitas Diponegoro Modul Inventarisasi," Jurnal Masyarakat Informatika, vol. 14, no. 1, pp. 2777–0648, 2023.

[8] N. W. Hidayah, R. R. Sasmita, M. K. Mayangsari, O. G. W. Kusuma, H. Rante, and A. Fariza, "Invitin Project: Scrum Framework Implementation in a Software Development Project Management," INTEK: Jurnal Penelitian, vol. 9, no. 1, p. 58, 2022, doi: 10.31963/intek.v9i1.3332.

[9] R. Y. Endra, Y. Aprilinda, Y. Y. Dharmawan, and W. Ramadhan, "Analisis Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website," EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi, vol. 11, no. 1, p. 48, 2021, doi: 10.36448/expert.v11i1.2012.

[10] A. Rama Febrianto, A. Wulansari, and L. Latipah, "Pengembangan Sistem Pengelolaan dan Pemantauan Proyek dengan Metode Agile Pola Scrum," Jurnal Teknik Informatika dan Sistem Informasi, vol. 6, no. 2, pp. 206–221, 2020, doi: 10.28932/jutisi.v6i2.2592.

[11] B. Reddaiah, "Handling Transition Product Backlog with Scrum off Scrum," International Journal of Advanced Information Science and Technology (IJAIST) ISSN, vol. 5, no. 1, pp. 123–126, 2016, doi: 10.15693/ijaist/2016.v5i1.123-126.

[12] L. Setiyani, "Desain Sistem : Use Case Diagram Pendahuluan," Prosiding Seminar Nasional : Inovasi & Adopsi Teknologi 2021, no. September, pp. 246–260, 2021, [Online]. Available:

https://journal.uii.ac.id/AUTOMATA/article/view/19517

[13] L. P. Dewi, U. Indahyanti, and Y. H. S, "Pemodelan Proses Bisnis Menggunakan Activity Diagram Uml Dan Bpmn (Studi Kasus Frs Online)," Informatika, pp. 1–9, 2021, [Online]. Available: https://repository.petra.ac.id/15653/1/BPMN-UML.pdf

[14] A. Edin and M. Qorbanzada, "An Ambidextrous Approach to User Acceptance Testing Management and Automation : How software customers can optimize their processes to explore new technology as well as current performance to improve the effectiveness of user acceptance testing," 2022, [Online]. Available: https://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-319551

[15] M. A. Chamida, A. Susanto, and A. Latubessy, "Analisa User Acceptance Testing Terhadap Sistem Informasi Pengelolaan Bedah Rumah Di Dinas Perumahan Rakyat Dan Kawasan Permukiman Kabupaten Jepara," Indonesian Journal of Technology, Informatics and Science (IJTIS), vol. 3, no. 1, pp. 36–41, 2021, doi: 10.24176/ijtis.v3i1.7531.