Applied Information System and Management (AISM)
Volume 8, (1) 2025, p. 53–64
P-ISSN: 2621-2536; E-ISSN: 2621-2544; DOI: 10.15408/aism.v8i1.42602

Analysis and Design of Inventory Management Information Systems at PT. XYZ

Nia Kumaladewi[1*] , Ryan Faatih Firdaus[2]

***Abstract*—This study aims to analyze and design an Inventory Management Information System (IMIS) for PT. XYZ, a company that currently relies on manual inventory processes, leading to inefficiencies and operational delays. To address these challenges, the IMIS was developed using the Rapid Application Development (RAD) methodology, which followed three key stages: requirements planning, design workshop, and implementation. In the requirements planning stage, comprehensive user needs and system specifications were gathered. During the user design phase, iterative prototyping and continuous feedback sessions helped refine the system's interface and functionalities. The rapid construction phase saw the development of core features—such as real-time inventory tracking, automated restocking alerts, and detailed transaction histories—using PHP, HTML, JavaScript, and MySQL. Finally, in the cutover phase, pilot testing demonstrated that the system reduced material search times from 15–30 minutes to nearly instantaneous retrieval, significantly improving inventory accuracy and overall operational efficiency. Further research is needed to evaluate its long-term performance and scalability. This study not only highlights the practical advantages of the RAD-based approach but also offers a promising model for organizations seeking to modernize their inventory management practices.**

***Index Terms*— Inventory management, UML, RAD.**

I. Introduction

The advancement of information and communication technology has significantly impacted various business operations. Many companies adopt technological solutions to enhance efficiency, productivity, and competitiveness. Information technology helps businesses access information, explore new markets, and create products that drive growth [1]. It also enables automation of manual processes and supports in-depth data analysis for faster, more accurate decision-making, ultimately improving operational efficiency and reducing costs.

A key focus area is inventory management, where effective management can provide a competitive advantage. Proper inventory control is essential for business success [2]. It now emphasizes logistics and international sourcing, with a need for performance measurement and strategies suited to different businesses [3]. Effective systems help companies monitor stock in real-time, reducing risks like stockouts or overstocking that could disrupt operations.

PT XYZ, which specializes in ERP solutions for palm oil plantations, manages its Research and Development (R&D) division inventory manually. Materials and products are stored in labeled lockers without systematic record-keeping, and checks are only done upon new orders. This approach results in challenges like difficulty in locating items and tracking quantities, leading to inefficiencies [4]. Employees waste 10–15 minutes searching for materials which could be allocated to more productive tasks.

These inefficiencies slow down PT XYZ's production process and impact product development. A lack of real-time information on stock levels can delay project timelines and hinder customer service. Therefore, an integrated inventory management information system is needed to streamline operations, from tracking raw materials to managing finished products. Such systems significantly affect post-implementation success through improved service quality and user satisfaction [5].

Implementing this system can save costs, enhance customer service, and reduce storage expenses by optimizing inventory levels [6]. Real-time data enables better decision-making, production planning, and resource allocation [7]. For PT XYZ, an effective system would allow for better trend analysis and more strategic inventory control, supporting growth and operational efficiency.

Effective inventory management is key to optimizing a company's financial and operational aspects. It involves a systematic approach to controlling and organizing inventory to minimize costs and ensure smooth production processes. Proper material planning methods are used to maintain a continuous flow of materials, one of which is the Re-order

Received: 26 November 2024; Revised: 21 March 2025; Accepted: 7 April 2025
*Corresponding author

[1]Nia Kumaladewi, UIN Syarif Hidayatullah Jakarta, Indonesia (e-mail: nia.kumaladewi@uinjkt.ac.id).
[2]Ryan Faatih Firdaus, UIN Syarif Hidayatullah Jakarta, Indonesia (e-mail: ryanfaatih.firdaus21@mhs.uinjkt.ac.id).

Point (ROP) method. The ROP system is a primary method for managing inventory by determining the optimal replenishment timing to improve service and inventory levels [8]. Centralized demand information strengthens this approach by enabling efficient reordering and enhancing customer service across the serial inventory system [9]. ROP can be effectively determined based on the maximum demand observed in recent periods, adjusted for lead time, ensuring adequate stock availability [10]. Furthermore, advanced methods like artificial neural networks have been applied to optimize ROP, leading to improved safety stock management, productivity, and reduced inventory costs [11]. The integration of demand data, optimal ROP, and computational models enables inventory managers to effectively balance service quality with cost efficiency.

In order to meet these demands, the research will examine PT XYZ's inventory management needs and use the Rapid Application Development (RAD) approach to create an appropriate solution. In order to guarantee that the finished product satisfies user demands, RAD facilitates rapid prototyping and iterative feedback. This strategy can offer an inventory system that is easy to use and flexible enough to accommodate the company's operating objectives and future expansion.

By using IoT (Internet of Things) technology for real-time stock tracking or including machine learning-based demand forecast features for procurement optimization, further research can broaden the system. Additionally, to improve accessibility and scalability across the company's operating sites, the creation of a cloud-based system may be taken into consideration. To gauge the system's long-term efficacy and efficiency, more assessment via direct trials in PT XYZ's operational environment is also crucial.

II. Related Work

Previous studies have focused on designing inventory management systems using the Rapid Application Development (RAD) methodology. One study by [12] focused on implementing RAD to create a web-based inventory system that improved the efficiency of inventory management. The system facilitated faster data entry, accurate stock tracking, and real-time access to reports. The RAD process, with its iterative and user-centered approach, enabled the creation of a system that effectively reduced errors and enhanced operational efficiency.

Similarly, [13] developed a school inventory management system using RAD, utilizing ReactJS, Redux, and Firebase. The RAD approach allowed for user involvement throughout the design process, ensuring the system met user needs. The study’s functional testing results were 100% valid, indicating that the system’s features performed as expected. However, non-functional tests highlighted compatibility issues with some web browsers, suggesting an area for further improvement.

In the study done by [14], they also employed RAD methodology in their study of village asset management. Their system aims to address the inefficiencies of managing village assets manually, providing a structured way to track asset status and generate reports. The RAD-based system enabled village officials to manage assets effectively and provided real-time updates that supported decision-making processes.

These studies collectively demonstrate the effectiveness of the RAD methodology in developing inventory management systems across different contexts, from businesses to schools and village administrations. For PT XYZ, these insights are particularly useful as they highlight how a similar approach can streamline inventory processes, reduce search times and minimize operational delays. Moreover, the challenges identified, such as browser compatibility and scalability, serve as important reminders that further refinement and targeted customization may be needed. While the research comes from varied environments, the lessons learned are highly applicable to PT XYZ. The emphasis on rapid development, iterative user feedback, and real-time data tracking provides a solid foundation for improving inventory management in a corporate setting.

III. Research Method

Rapid Application Development (RAD) is a linear sequential system development method that emphasizes a relatively short development cycle, allowing time savings and a faster system development process [15]. As described by [16], the RAD methodology involves several stages that facilitate fast and iterative development, ensuring user involvement. The stages include Requirements Planning, Design Workshop, and Implementation as shown in Fig. 1.

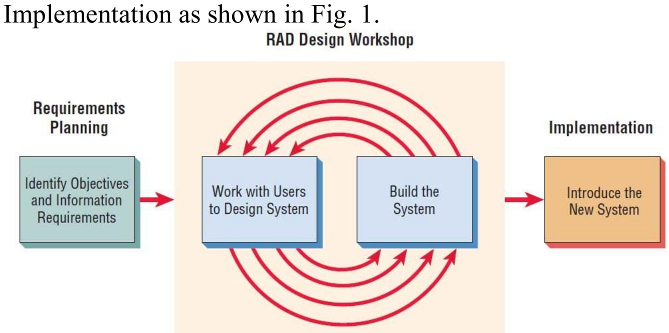

Fig. 1. RAD phases.

A. Requirements Planning Phase

In this phase, the requirements analysis involves gathering data from stakeholders to identify the system's requirements, constraints, and objectives that will be developed [17]. It requires active participation from various stakeholders, not just in reviewing proposals but in actively engaging in discussions about system objectives and user needs. This phase involves users from different levels within the organization to ensure that the system design is well-informed and comprehensive. The process begins with structured interviews with related stakeholders, where the R&D staffs highlight inefficiencies like spending 10–15 minutes daily searching for materials in unlabeled lockers, while R&D teams emphasize the need for real-time stock visibility to avoid project delays. Through workshops involving executives, key requirements are identified, such as reducing search times to under 2 minutes,

Applied Information System and Management (AISM)
Volume 8, (1) 2025, p. 53–64
P-ISSN: 2621-2536; E-ISSN: 2621-2544; DOI: 10.15408/aism.v8i1.42602

and providing alerts for low stock levels.

B. Design Workshop Phase

During the Design Workshop phase, the design process takes place, along with any necessary refinements to ensure alignment between user needs and system design. In this phase, the researcher develops the proposed system to ensure it operates smoothly and is anticipated to resolve the current issues [18]. Sequence diagrams are used to show interactions between objects and working prototypes of key interfaces, such as the inventory management screens, which R&D staff can test and provide immediate feedback on. This iterative design process is crucial to the RAD methodology, as it ensures that users play an active role in shaping the system, allowing for immediate adjustments if the design does not meet their expectations. This engagement is essential for ensuring that the final product closely aligns with user requirements.

C. Implementation Phase

The Implementation phase involves the practical deployment of the system design. During this phase, analysts collaborate closely with users during workshops to design both the business and non-technical aspects of the company. After the design is approved, the system or its components are constructed, tested, and refined before being introduced into the organization. The testing process ensures that the system functions as intended and is ready for full deployment [19]. Users continue to provide feedback during this phase, ensuring that the system meets their needs and expectations before it is fully integrated into the organizational workflow. Comprehensive training sessions are conducted for R&D staff. The company implements a parallel testing period where both existing and new systems operate simultaneously, allowing for thorough data accuracy verification. The methodology's emphasis on quick iterations may lead to oversights in system testing. Additionally, if user feedback is not properly accommodated, user dissatisfaction could arise, undermining adoption and long-term success.

IV. Result

A. Requirements Planning Phase

In the proposed system illustrated in Fig. 2, the admin oversees user data management, ensuring that access to the application's features is restricted to authorized users. General staff utilize the application to monitor material and product stock levels and review transaction histories, enabling them to track inventory status and prevent production shortages. Inventory staff, on the other hand, are responsible for restocking materials and initiating the production process, which involves updating inventory records. They also manage the addition of new products and materials, ensuring that the system maintains an accurate stock database. The application serves as a centralized platform where each role performs specific tasks, promoting efficient communication and effective inventory management within the organization. Additionally, the system design incorporates the ROP (Reorder Point) calculation. The mechanism works by notifying users when raw material stock reaches the minimum threshold set in the material_minimum column, prompting procurement actions. The formula used for ROP is as follows.

$$ROP = Demand.LeadTime + Safetystock \quad (1)$$

where demand is the average daily usage or sales rate of an item. Meanwhile, lead time is the average time it takes for an order to arrive from a supplier, and safety stock express the amount of inventory to keep on hand to avoid running out of stock.

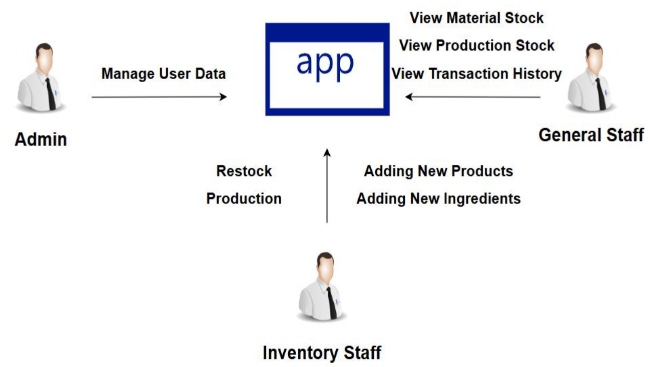

Fig. 2. Recommended system.

B. Design Workshop Phase

The main features of an inventory management system are shown in the use case diagram in Fig. 3, which is assigned to the following user roles: admin, inventory staff, and general staff. Logging into the system, controlling production, replenishing supplies, introducing new items and materials, monitoring material and product stock levels, and analyzing transaction history are just a few of the many duties that General Staff may complete. While they have comparable access, inventory staff members are mostly responsible for inputting and keeping track of stock-related data.

The admin position has administrative powers in addition to access to all staff functions. These include managing user accounts and seeing the user list, which guarantees appropriate permission assignments and regulated system access. After a successful login, the Logout function, represented by a "extend" connection from Login, indicates that logging out is an optional step. An organized role-based access system inside the inventory management platform is made possible by this diagram, which clearly delineates the functionality allocated to each user role.

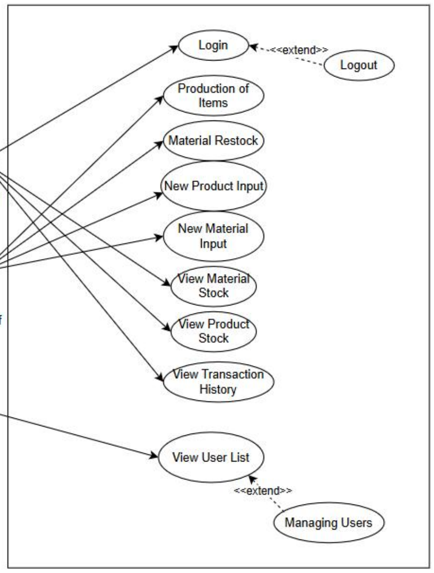

Fig. 3. Use case diagram.

The sequence diagram in Fig. 4 illustrates the login process available to all users. This diagram shows the steps involved, starting from the user entering a username and password through the interface. The interface then sends this information to the system for validation. The system checks the validity of the username and password by comparing them with the data in the database. The validation process in the database has two possible outcomes: if the username and password match, the database sends a successful validation response to the system, which then instructs the interface to open the main page; if the username or password does not match, the database sends a failed validation response to the system, which then instructs the interface to display the login form again along with an error message. Based on this validation result, the interface will display the main page if the validation is successful or show the login form again if the validation fails.

The sequence diagram in Fig. 5 illustrates the production process that can be used by the admin and inventory staff. The interaction begins when the user logs in. The system then opens the main page. Once on the main page, the user selects the production menu, which prompts the system to display the production page. The user proceeds by filling out the form on the production page, providing the necessary data, and then pressing the submit button. The system receives the form data and checks the availability of materials in the database. If the materials are sufficient, the system proceeds to save the data and sends a successful response, which is displayed as a successful message to the user. If the materials are insufficient, the system sends a failure response, and a failure message is displayed to the user, indicating the inability to proceed with the production request.

The sequence diagram in Fig. 6 illustrates the restocking process that can be used by the admin and inventory staff. The interaction begins when the user logs in. The system then opens the main page. Once on the main page, the user selects the restock menu, which prompts the system to display the restock page.

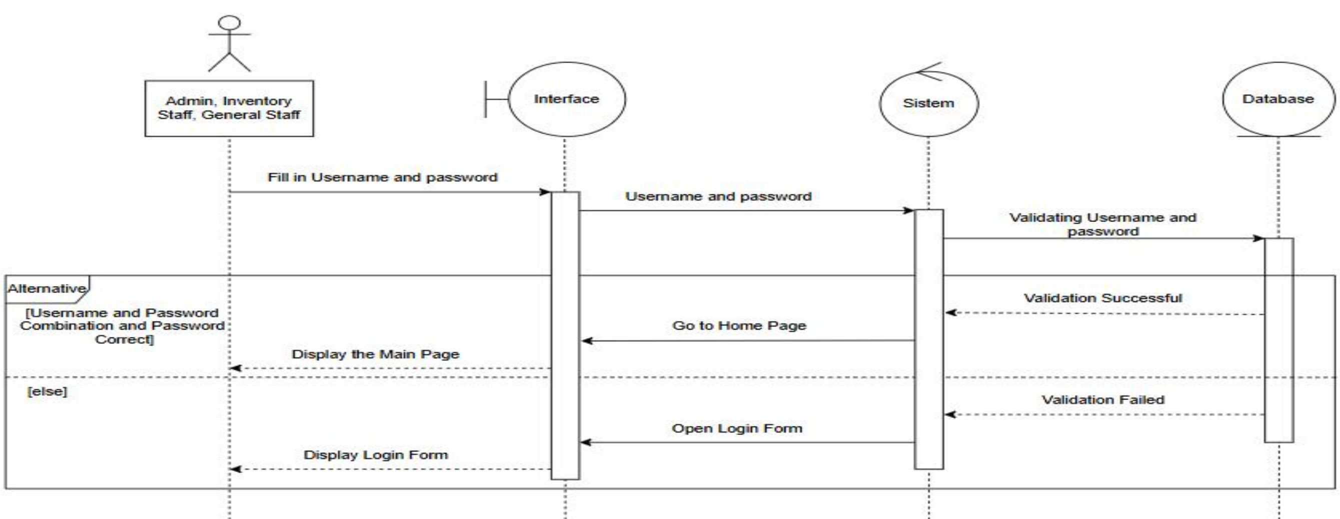

Fig. 4. Login sequence diagram.

Applied Information System and Management (AISM)
Volume 8, (1) 2025, p. 53–64
P-ISSN: 2621-2536; E-ISSN: 2621-2544; DOI: 10.15408/aism.v8i1.42602

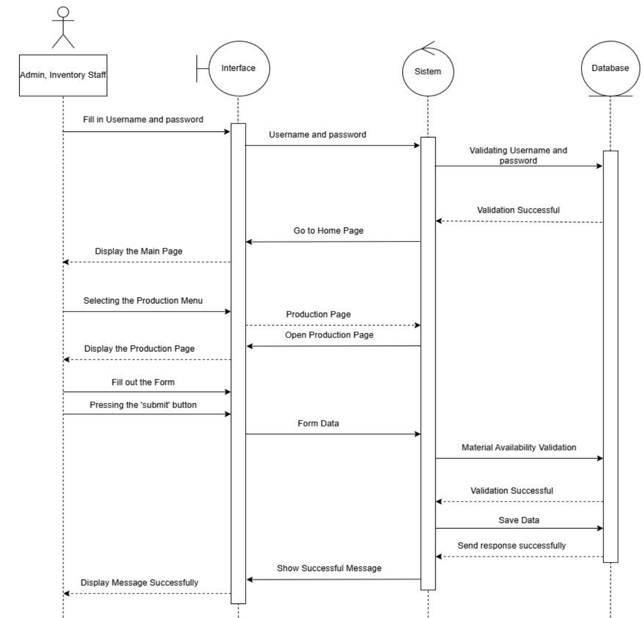

Fig. 5. Production sequence diagram.

The user proceeds by filling out the form on the restock page, providing the necessary data, and then pressing the submit button. The system receives the form data and checks if the quantity entered in the form is greater than 0. If it is, the system proceeds to save the data and sends a success response, which is displayed as a success message to the user. If not, the system sends a failure response, and a failure message is displayed to the user, indicating the inability to complete the restock request.

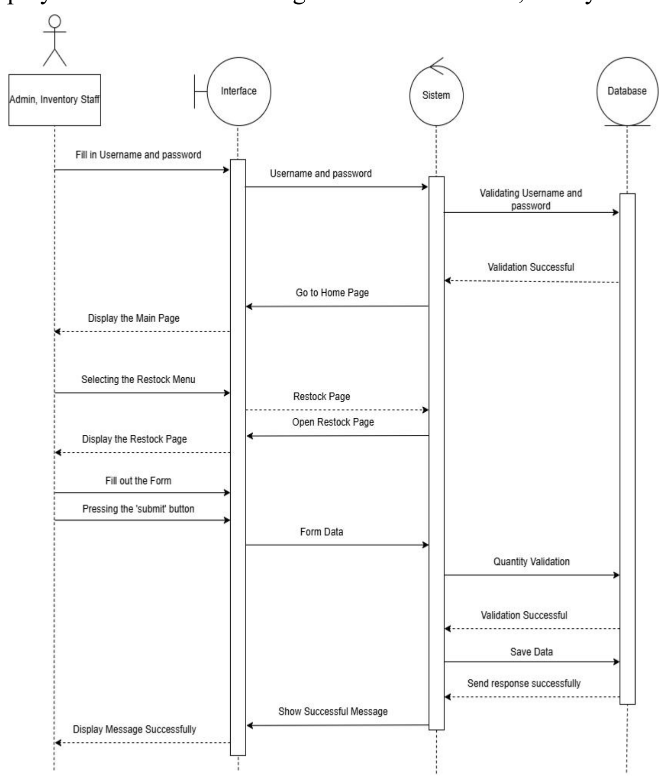

Fig. 6. Restock sequence diagram.

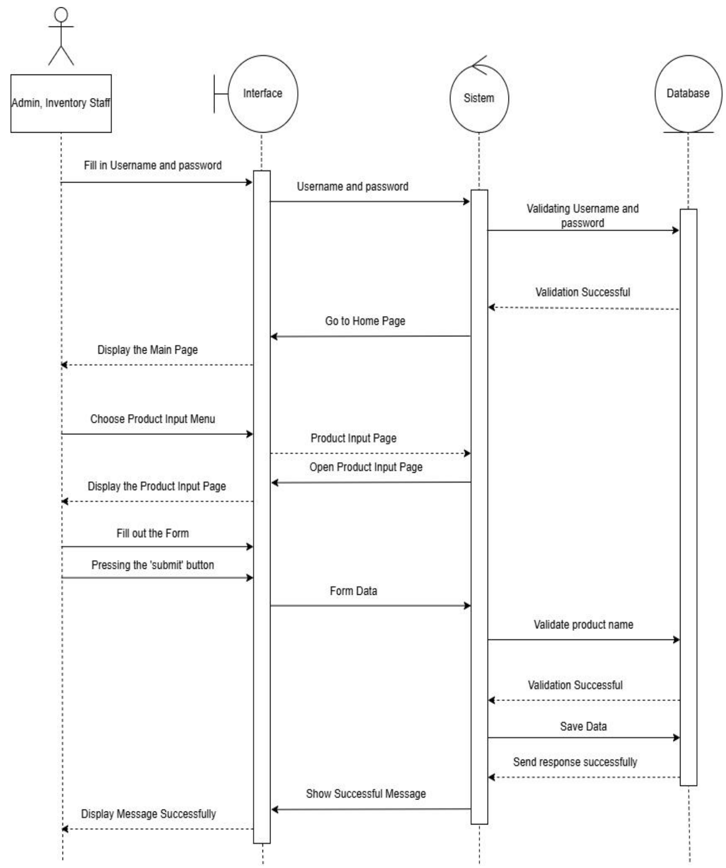

Fig. 7. Product input sequence diagram.

The sequence diagram in Fig. 8 illustrates the new material input process that can be used by the admin and inventory staff. The interaction begins when the user logs in. The system then opens the main page. Once on the main page, the user selects the material input menu, which prompts the system to display the material input page. The user proceeds by filling out the form on the material input page, providing the necessary data, and then pressing the submit button. The system receives the form data and checks whether the material name is already registered in the database. If it is not, the system proceeds to save the data and sends a success response, which is displayed as a success message to the user. If it is already registered, the system sends a failure response, and a failure message is displayed to the user, indicating the inability to proceed with the new material input request.

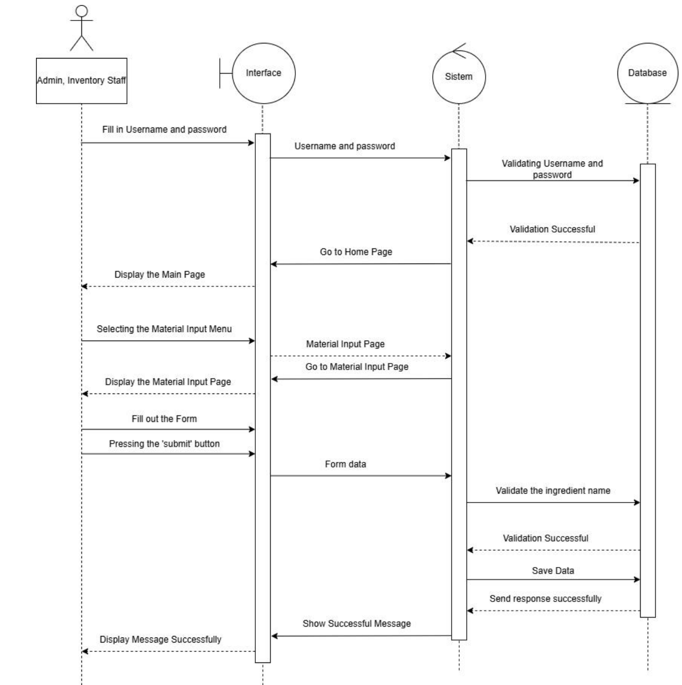

Fig. 8. Material input sequence diagram.

The sequence diagram in Fig. 9 illustrates the process of viewing material stock, which can be used by all users. The interaction begins when the user logs in. The system then opens the main page. Once on the main page, the user selects the material stock menu, which prompts the system to display the material stock page. The interface then requests material stock data from the system. The system performs a query on the database to retrieve the requested material stock data. The database returns this data to the system, which then forwards it to the interface. The interface processes the data and displays the page with the material stock information to the user, completing the process of viewing material stock.

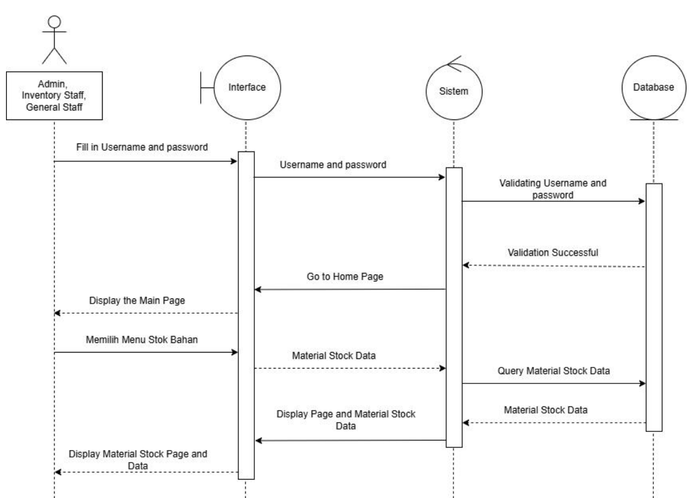

Fig. 9. Material stock sequence diagram.

The sequence diagram in Fig. 10 illustrates the process of viewing product stock, which can be used by all users. The interaction begins when the user logs in. The system then opens the main page. Once on the main page, the user selects the product stock menu, which prompts the system to display the product stock page. The interface then requests product stock data from the system. The system performs a query on the database to retrieve the requested product stock data. The database returns this data to the system, which then forwards it to the interface. The interface processes the data and displays the page with the product stock information to the user, completing the process of viewing product stock.

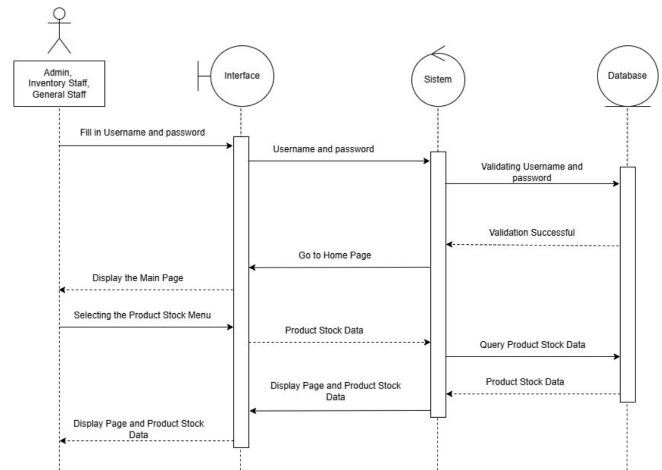

Fig. 10. Product stock sequence diagram.

The sequence diagram in Fig. 11 illustrates the process of viewing transaction history, which can be used by all users. The interaction begins when the user logs in. The system then opens the main page. Once on the main page, the user selects the transaction history menu, which prompts the system to display the transaction history page. The interface then requires transaction history data from the system. The system performs a query on the database to retrieve the requested transaction history data. The database returns this data to the system, which then forwards it to the interface. The interface processes the data and displays the page with the transaction history information to the user, completing the process of viewing transaction history.

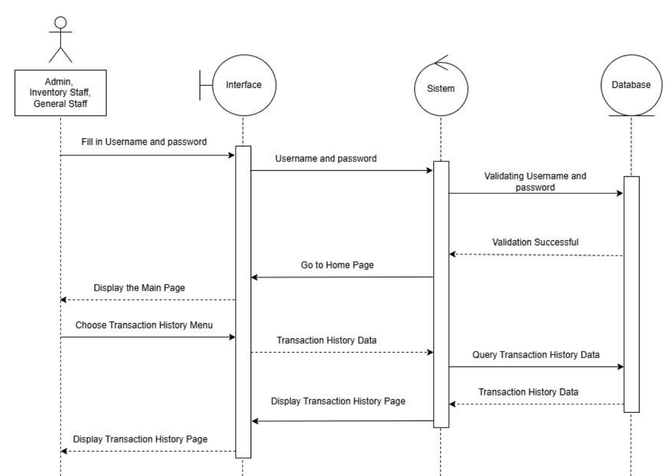

Fig. 11. Transaction history sequence diagram.

Applied Information System and Management (AISM)
Volume 8, (1) 2025, p. 53–64
P-ISSN: 2621-2536; E-ISSN: 2621-2544; DOI: 10.15408/aism.v8i1.42602

The sequence diagram in Fig. 12 illustrates the process of viewing the user list, which can be accessed by the admin. The interaction begins when the user logs in. The system then opens the main page. Once on the main page, the user selects the user management menu, prompting the system to display the user management page. The interface then requests the user list data from the system. The system performs a query on the database to retrieve the requested user list data. The database returns this data to the system, which then forwards it to the interface. The interface processes the data and displays the page with the user list information to the admin, completing the process of viewing the user list.

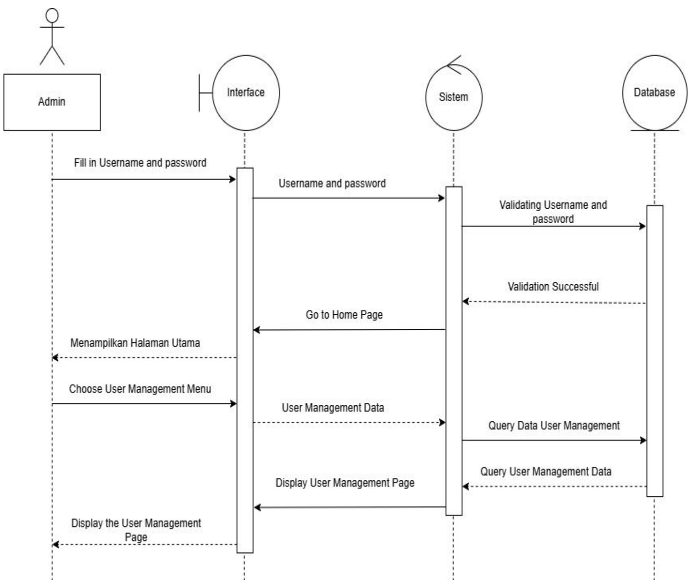

Fig. 12. User list sequence diagram.

The sequence diagram in Fig. 13 illustrates the user management process carried out by the admin within the system. The interaction begins when the user logs in, after which the system opens the main page. Once on the main page, the user selects the user management menu, prompting the system to display the user management page. The interface then requests user list data from the system, which queries the database to retrieve the requested user list data. The database returns this data to the system, which then forwards it to the interface. The interface processes the data and displays the page with the user list to the admin.

The sequence diagram in Fig. 14 illustrates the logout process, which can be used by all users. The interaction begins when the user logs in, and the system opens the main page. Once on the main page, the user selects the logout menu. The interface then requires the system to delete the user's session. The system proceeds to delete the user session and instructs the interface to display the login page to the user. Finally, the interface displays the login page to the user, completing the logout process.

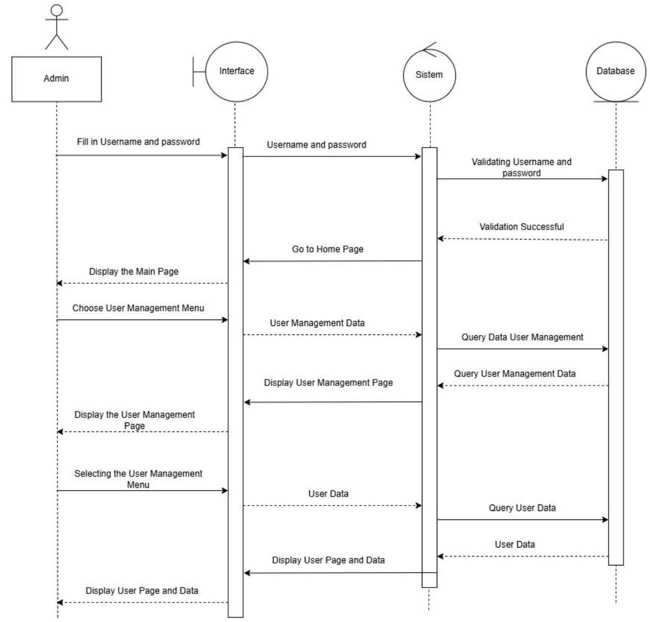

Fig. 13. User management sequence diagram.

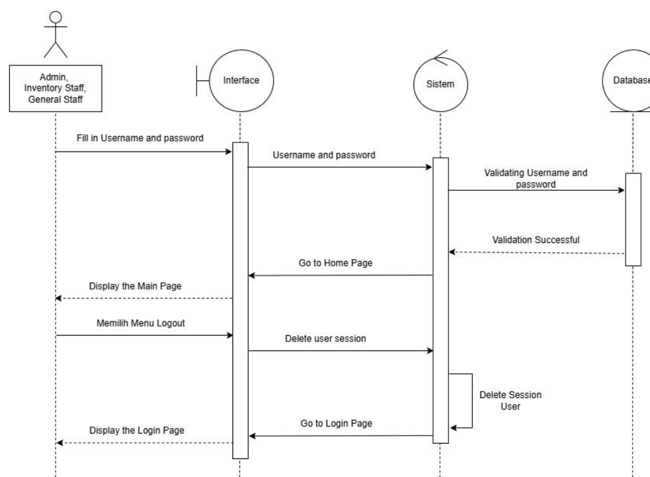

Fig. 14. Logout sequence diagram.

The class diagram in Fig. 15 represents the main entities involved in the application, detailing its attributes, relationships, and methods. The 'user_account' class manages user data, including attributes such as 'id', 'full_name', 'username', 'password', and 'role'. It has methods for login, logout, password changes, and profile updates. The 'historical' class, related to 'user_account' with a many-to-one relationship, records transaction history and includes attributes like 'id', 'account_id', 'product_id', 'material_id', 'created_at', 'quantity_in', 'quantity_out', 'activity', and 'description'. Its method only includes adding entries, making it a central record of user activities.

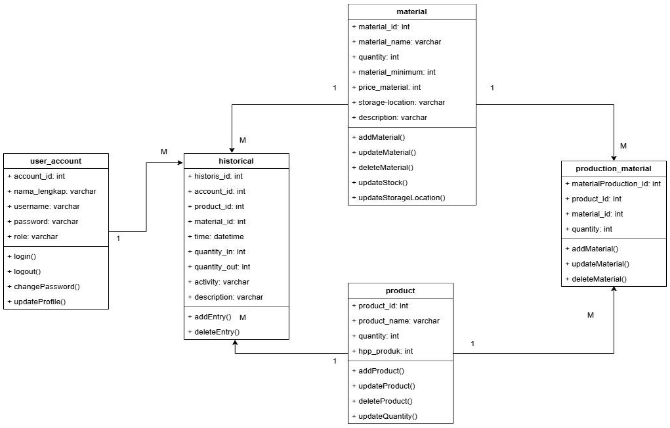

Fig. 15. Class diagram.

The 'material' class represents materials used in production, with attributes such as 'id', 'material_name', 'quantity', 'minimum_amount', 'price', 'location', and 'description'. It provides methods for adding, updating, and deleting materials, as well as methods to update stock and storage location. This calculated 'minimum_amount' will serve as a threshold, triggering a notification to alert the user to reorder the material, as well as other materials nearing this threshold. By implementing this approach, the system ensures optimal inventory levels, reducing the risk of stockouts while minimizing excess inventory. This proactive measure not only streamlines operations but also contributes to cost savings and improved resource management. Furthermore, it allows users to make timely and informed decisions, fostering a more efficient and responsive supply chain.

The 'product' class represents finished products, with attributes including 'id', 'product_name', 'quantity', and 'cost'. This class has methods to add, update, delete, and modify product quantities. The 'production_material' class manages the relationship between materials and products in production, with attributes like 'id', 'product_id', 'material_id', and 'quantity'. It provides methods for adding, updating, and deleting materials within the production process. The relationships between classes are represented by one-to-many or many-to-one connections, ensuring that data flow between user accounts, transaction history, materials, and products is consistent and well-organized. This structure is essential for managing and tracking inventory, production activities, and user interactions within the system.

C. *Implementation Phase*

During the implementation phase, program development uses PHP, HTML, CSS, JavaScript, and libraries like jQuery and Bootstrap. This stage involves integrating components that were designed and tested during the prototyping phase. PHP serves as the primary programming language for building backend logic and interacting with the MySQL database. HTML and CSS are used for structuring and styling the user interface, while JavaScript, with jQuery, adds interactive and dynamic functionality to the web pages. Bootstraps are utilized to expedite interface development by providing responsive design components. MySQL manages the database, storing and organizing the information required by the system. Throughout the implementation, each module is gradually integrated and tested in a real environment, allowing developers and users to provide quick feedback and ensuring that the system meets the desired requirements before full deployment.

1) Login Interface

Figure 16 shows the user interface design of the login page for the XYZ R&D inventory management system. On this page, there are two fields that users can fill in to log in: a username field and, directly below it, a password field, followed by a sign-in button to access the system.

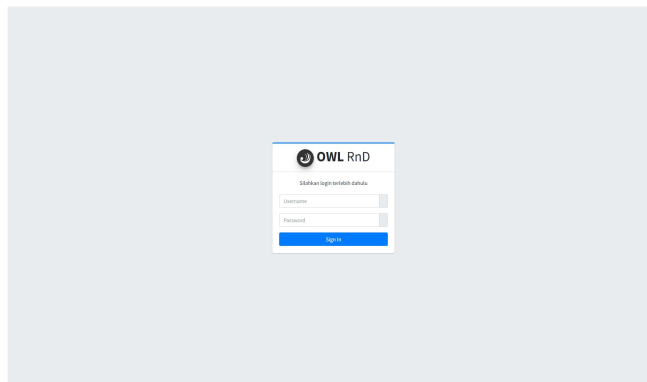

Fig. 16. Login interface.

2) Homepage Interface

Figure 17 shows the user interface design of the main page of the XYZ R&D inventory management system that has been designed. The main display features a quick action panel with different colors for access to production features, restocking materials, viewing material stock, and product stock. Navigation on the left side makes it easy for users to explore features such as material management, devices, and transaction history. At the top, users can see the active username and options for logging out. The 'Recent Transaction History' table at the bottom provides detailed information about recent activities, including item name, quantity, type of activity, description, and transaction time, allowing for real-time activity monitoring.

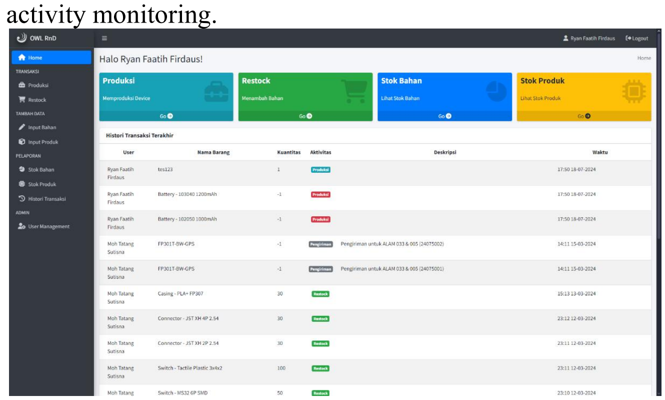

Fig. 17. Homepage interface.

3) Production Interface

Figure 18 shows the user interface design of the production page of the XYZ R&D inventory management system that has been designed. This page contains a form that users must fill

Applied Information System and Management (AISM)
Volume 8, (1) 2025, p. 53–64
P-ISSN: 2621-2536; E-ISSN: 2621-2544; DOI: 10.15408/aism.v8i1.42602

out when they want to produce a product. The form includes a dropdown column for selecting the product to be produced, a quantity column for entering the number of products to be produced, and a check button to verify the number of materials needed for the production of that product, which will be displayed in a table below. The table will show the names of the materials, the quantity of materials needed, the available stock of those materials, whether the materials are sufficient, the price of each unit of the materials, and the total cost of the materials. Below the table, there is an optional description column where users can provide context for the production process, and finally, there is a submit button at the bottom right to send the form data to the system.

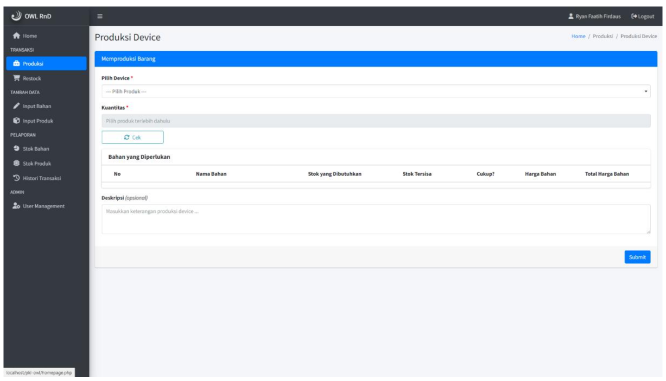

Fig. 18. Production interface.

4) *Restock Page Interface*

Figure 19 shows the user interface design of the restock page of the XYZ R&D inventory management system that has been designed. This page contains a form that users must fill out when they want to restock material. The form includes a drop-down column for selecting the material to be added, a quantity column for entering the amount of material to be added, an optional description column for providing context, and finally, there is a submit button at the bottom right to send the form data to the system.

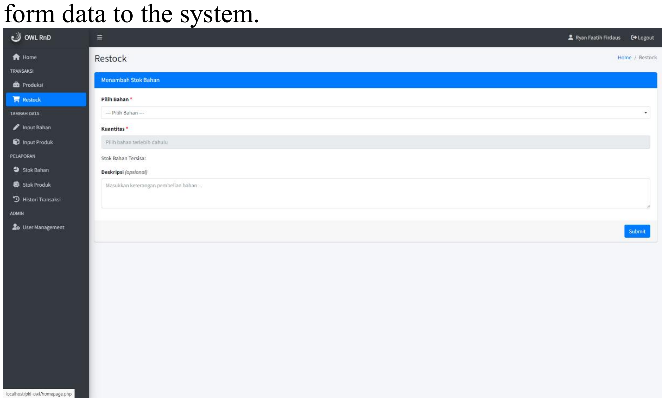

Fig. 19. Restock interface.

5) *Material Input Interface*

Figure 20 shows the user interface design of the material input page of the XYZ R&D inventory management system that has been designed. This page contains a form that users must fill out when they want to register a new material. The form includes a column for the name of the material to be added, a quantity column for entering the amount of material to be added, a column for the price of the material, storage location, and an optional description column for providing context. Finally, there is a submit button at the bottom right to send the form data to the system.

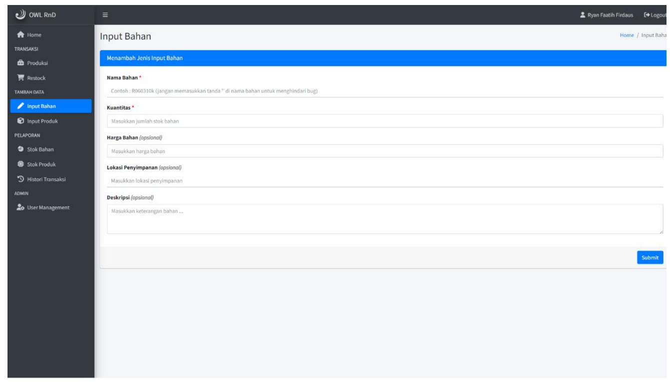

Fig. 20. Material input interface.

6) *Product Input Interface*

Figure 21 shows the user interface design of the product input page of the XYZ R&D inventory management system that has been designed. This page contains a form that users must fill out when they want to register a new product. The form includes a column for the name of the product to be added, an optional image column for the material, a table containing the materials and the quantities needed for the production of that product, and finally, there is a submit button at the bottom right to send the form data to the system.

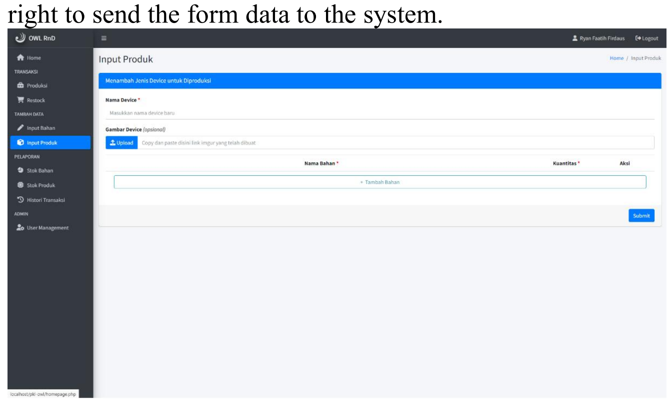

Fig. 21. Product input interface.

7) *Material Stock Interface*

Figure 22 shows the user interface design of the material stock page of the XYZ R&D inventory management system that has been designed. This page contains a table with information related to the materials owned. The table includes information on material ID, material name, material stock, last price of the material, storage location of the material, and description of the material.

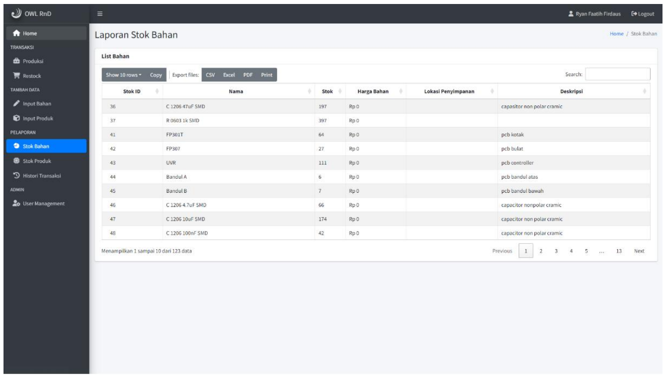

Fig. 22. Material stock interface.

8) Product Stock Interface

Figure 23 shows the user interface design of the product stock page of the XYZ R&D inventory management system that has been designed. This page contains a table with information related to the products owned. The table includes information on product ID, product name, product stock, and production cost of the product.

9) Transcation History Interface

Figure 24 shows the user interface design of the transaction history page of the XYZ R&D inventory management system that has been designed. This page contains a table with information related to the activity history in the system. The table includes information on the user performing the activity, the name of the affected item, the affected quantity, a description of the activity, and the timestamp of when the activity occurred.

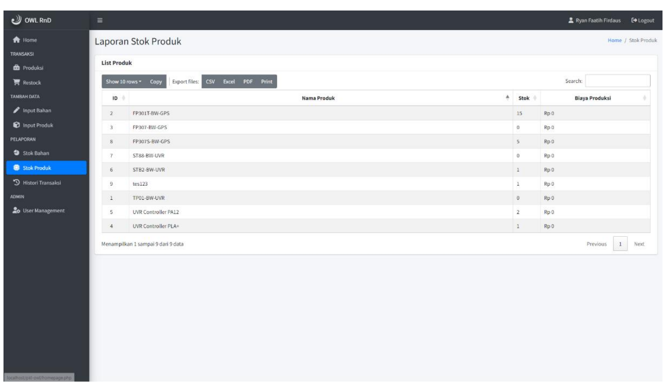

Fig. 23. Product stock interface.

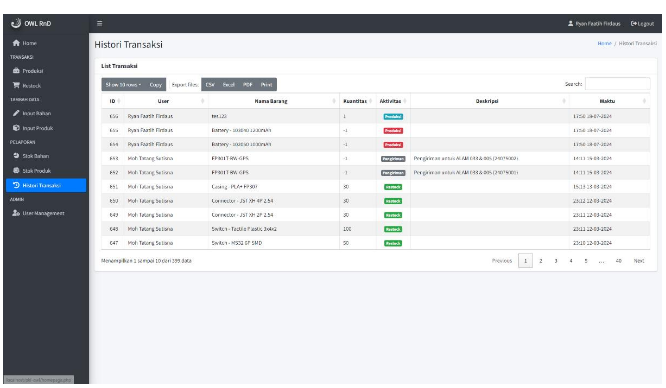

Fig. 24. Transaction history interface.

10) User Management Interface

Figure 25 shows the user interface design of the user management page of the XYZ R&D inventory management system that has been designed. This page contains a table with information related to the users registered in the system. In the top left, there is a button to add a new user. The table includes information on the user's full name, username, role, and actions to modify or delete that account.

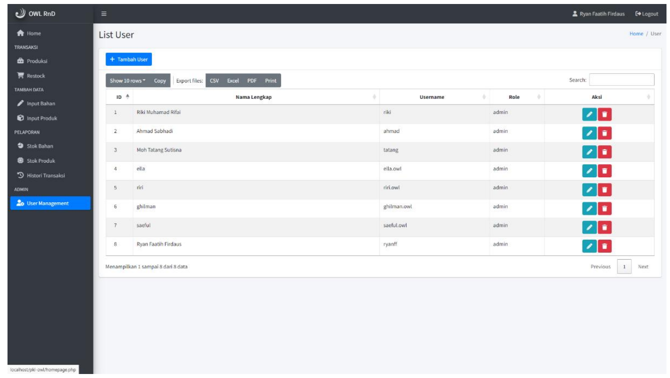

Fig. 25. User management interface.

11) Add User Interface

Figure 26 shows the user interface design of the ad account page of the XYZ R&D inventory management system that has been designed. This page contains a form that the admin must fill out when registering a new user in the application. The form includes a column for the full name of the user to be added, an option for the account role, a username column, a password column, and finally, there is a submit button at the bottom right to send the form data to the system.

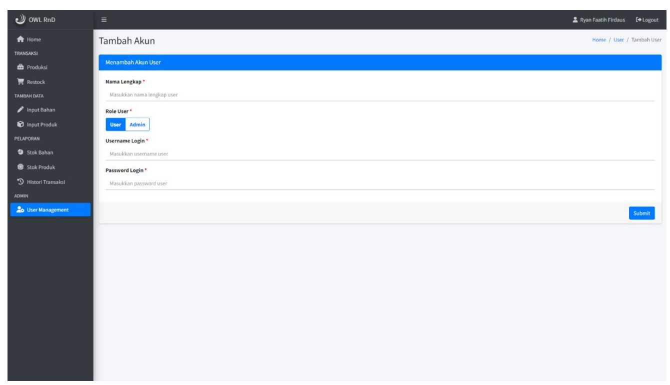

Fig. 26. User management interface.

12) Edit User Interface

Figure 27 shows the user interface design of the edit user page of the XYZ R&D inventory management system that has been designed. This page contains a form that can be modified by the admin when they want to change user data in the application. The form includes a column for the user's full name, an option for the account role, a username column, a change password button, and finally, there is a submit button at the bottom right to send the form data to the system.

Applied Information System and Management (AISM)
Volume 8, (1) 2025, p. 53–64
P-ISSN: 2621-2536; E-ISSN: 2621-2544; DOI: 10.15408/aism.v8i1.42602

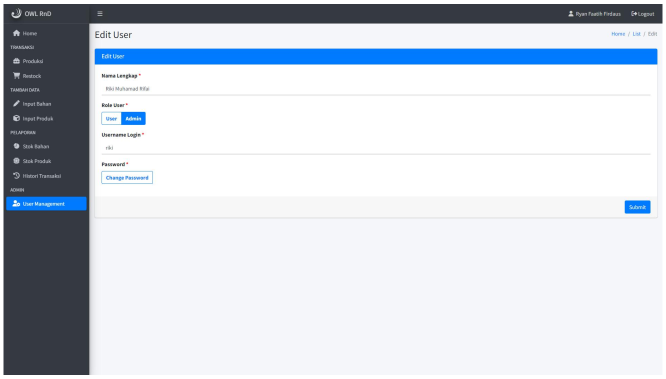

Fig. 27. User management interface.

During this phase, a primary technical obstacle also emerged with the PHP back-end: handling the complex categorization of R&D materials and products that had previously been stored in physical lockers. The development team overcame this challenge by creating a hierarchical database structure that mirrored the physical storage system, while also incorporating detailed metadata to facilitate easy digital searches. This solution reduced search times from 10–15 minutes to near-instant digital lookups.

Another significant challenge was real-time inventory tracking, particularly for implementing the Re-order Point (ROP) system functionality. To prevent delays in project timelines caused by material shortages, the team developed a JavaScript-based real-time monitoring system. This system triggers automated alerts when inventory approaches predefined reorder points and is integrated with customizable parameters tailored to different material categories.

When comparing previous research and older systems, the new system offers big improvements. Traditional inventory management often relied on manual tracking or partly digital methods, which were slow and prone to mistakes. In contrast, the new system uses a solid PHP backend, an organized MySQL database, and real-time monitoring to boost efficiency, speed, and accuracy. Several studies have shown that hierarchical databases and live tracking can significantly improve operations, and our system confirms this by cutting material search times and by triggering automatic reordering when needed.

User feedback during the testing phase further reinforced these improvements. R&D staff reported significant enhancements in daily operations, particularly praising the swift inventory search function and the elimination of project delays due to proactive stock monitoring. Although initial concerns were raised regarding the complexity of data entry forms, subsequent modifications simplified the process while maintaining strict data consistency standards.

V. Conclusion

This study concludes that the information system developed for PT XYZ significantly enhances inventory management through streamlined processes and advanced technological integration. The system's real-time data tracking and user-friendly interface simplify stock monitoring, enabling employees to manage inventory with greater accuracy and efficiency. For instance, staff can now access instant stock updates while the automated Reorder Point (ROP) feature ensures timely procurement by alerting teams when stock levels approach predefined thresholds. Previously, staff spent 10–15 minutes searching for materials, but with the new system, this process has become nearly instantaneous. This not only reduces waiting time but also reduces excess inventory, lowers storage costs, and frees up valuable resources. Additionally, by providing actionable insights such as stock turnover rates, the system empowers managers to make more informed, data-driven decisions that further optimize operations.

Despite these advantages, the research has notable limitations. The study's focus on the R&D division might not fully capture the operational challenges faced by other departments like sales or logistics, which limits how broadly the findings can be applied across the company. The short development and testing period also meant that the long-term performance of the system under varying demands wasn't fully evaluated. Moreover, with rapid technological advancements, features like the ROP method may soon be overshadowed by AI-driven solutions, and user resistance or inadequate training could affect the system's overall effectiveness.

Looking ahead, further research should expand the system's implementation to other departments to assess its scalability and company-wide impact. Future development could explore integrating advanced technologies such as AI and machine learning to boost demand forecasting and inventory optimization. Additionally, investing in comprehensive training programs is crucial to ensure smooth adoption by all users. To measure the success of the system, metrics like reduced stockout frequency, improved inventory turnover ratios, and positive user satisfaction surveys will be key, along with operational benchmarks such as shorter order fulfillment times, reduced production downtime, and quantifiable cost savings from optimized inventory levels.

References

[1] M. I. Mgunda, "The impacts information technology on business," *Journal of International Conference Proceedings*, vol. 2, no. 3, pp. 149–156, Dec. 2019, doi: 10.32535/jicp.v2i3.656.

[2] M. Salman, M. P. Bhagat, and N. S. Kumar, "Role of inventory management and control in a manufacturing company," *International Journal For Science Technology And Engineering*, vol. *11, no.* 4, pp. 3836–3841, 2023, doi: 10.22214/ijraset.2023.51143.

[3] A. Dubey and R. Kumar, "Recent trends and advancements in inventory management," *ICST Transactions on Scalable Information Systems*, Dec. 2023, doi: 10.4108/eetsis.4543.

[4] E. Sahin and Y. Dallery, "Assessing the impact of inventory inaccuracies within a Newsvendor framework," *European Journal of Operational Research*, vol. 197, no. 3, pp. 1108–1118, Apr. 2008, doi: 10.1016/j.ejor.2008.03.042.

[5] P.-F. Hsu, H. R. Yen, and J.-C. Chung, “Assessing ERP post-implementation success at the individual level: Revisiting the role of service quality,” *Information & Management*, vol. 52, no. 8, pp. 925–942, Jul. 2015, doi: 10.1016/j.im.2015.06.009

[6] K. Pandey, A. Tripathi, V. Sharma, T. Mittal, and J. Abrol, “Inventory Management Systems: A Comprehensive Review and Analysis,” *Interantional Journal of Scientific Research in Engineering and Management*, vol. 07, no. 11, pp. 1–11, Nov. 2023, doi: 10.55041/ijsrem26979.

[7] M. P. V. De Oliveira and R. Handfield, “Analytical foundations for development of real-time supply chain capabilities,” *International Journal of Production Research*, vol. 57, no. 5, pp. 1571–1589, Jul. 2018, doi: 10.1080/00207543.2018.1493240.

[8] J. R. Freeland, E. N. Weiss, and R. Landel, “Managing Inventories: the reorder point system,” *SSRN Electronic Journal*, Jan. 2010, doi: 10.2139/ssrn.1584179.

[9] F. Chen, “Echelon reorder points, installation reorder points, and the value of centralized demand information,” *Management Science*, vol. 44, no. 12-part-2, pp. S221–S234, Dec. 1998, doi: 10.1287/mnsc.44.12.s221.

[10] G. Priniotakis and P. Argyropoulos, “Inventory management concepts and techniques,” *IOP Conference Series Materials Science and Engineering*, vol. 459, Art. no. 012060, Dec. 2018, doi: 10.1088/1757-899x/459/1/012060.

[11] T. Inprasit and S. Tanachutiwat, “Reordering point determination using machine learning technique for inventory management,” *2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST)*, pp. 1–4, 2018. [Online]. Available: https://doi.org/10.1109/ICEAST.2018.8434473.

[12] B. Rudianto and Y. E. Achyani, “Penerapan metode rapid application development pada sistem informasi persediaan barang berbasis web,” *Bianglala Informatika*, vol. 8, no. 2, pp. 117–122, Sep. 2020, doi: 10.31294/bi.v8i2.8930.

[13] M. D. R. Bassya, F. Amalia, and S. Adinugroho, “Pengembangan sistem manajemen barang inventaris SMKN 1 pasuruan berbasis website menggunakan metode rapid application development,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 5, no. 4, pp. 1488–1496, 2021. [Online]. Available: https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/8861

[14] E. A. Pratama, S. Krisgianti, and H. M. Paramita, “Penerapan metode rapid aplication development (RAD) pada pengembangan sistem informasi inventaris aset desa rempoah,” *Infotek Jurnal Informatika Dan Teknologi*, vol. 6, no. 1, pp. 49–59, Jan. 2023, doi: 10.29408/jit.v6i1.7313

[15] S. L. M. Sitio, *Penerapan Metode Rapid Application Development (RAD) untuk Aplikasi E-Learning Berbasis Web*. Eureka Media Aksara, 2023

[16] K. E. Kendall and J. E. Kendall, *Systems Analysis and Design*. Pearson, 2014.

[17] N. F. Irwanda, N. A. Agussalim, and N. A. R. E. Najaf, “Rancang bangun sistem informasi perpustakaan SMA negeri 1 sreseh sampang berbasis website dengan metode RAD,” *Neptunus*, vol. 2, no. 3, pp. 87–95, Jul. 2024, doi: 10.61132/neptunus.v2i3.214.

[18] N. Aini, S. A. Wicaksono, and I. Arwani, “Pembangunan sistem informasi perpustakaan berbasis web menggunakan metode rapid application development (RAD) (Studi pada: SMK negeri 11 malang),” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK)*, vol. 3, no. 9, pp. 8647–8655, Sep. 2019. [Online]. Available: https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6236/2996

[19] A. Andeka, S. A. Saputera, M. Utami, and A. Sonita, “Sistem informasi penerimaan siswa baru SMAN 05 kepahiang berbasis website menggunakan metode rapid application development,” *JUSIBI (Jurnal Sistem Informasi Dan E-Bisnis)*, vol. 4, no. 2, pp. 102–111, Jul. 2022, doi: 10.54650/jusibi.v4i2.452.