JURNAL TEKNOLOGI DAN OPEN SOURCE
Vol. 8, No. 2, December 2025, pp. 514~528
e-ISSN: 2622-1659, accredited Four Grade by Kemenristekdikti, Decree No: 152/E/KPT/2023
DOI: https://doi.org/10.36378/jtos.v8i2.4692

Developing a Web-Based Printing Transaction System Using The Prototype Method: A Case Study at Amanah Advertising

Annisa Dita Putri Kartika[1], Anindo Saka Fitri[2], Nambi Sembilu[3]
[1,2,3] Department of Information Systems, Faculty of Computer Science, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Indonesia

Article Info

Article history:

Received 07 18, 2025
Revised 09 20, 2025
Accepted 11 23, 2025

Keywords:

Management Information System
Web-Based Application
Prototype Method
Black-box Testing
User Acceptance Testing (UAT)

ABSTRACT

Digital transformation is increasingly vital for micro, small, and medium enterprises (MSMEs), including those in the printing sector, to enhance their operational effectiveness. Prior to this study, Amanah Advertising managed its orders, transactions, and inventory manually, which frequently resulted in data recording errors and poor coordination. To address these challenges, this research developed a web-based transaction management information system using the Prototype method. The development process involved two iterative stages of user feedback to refine system functionality. Enhancements included the addition of key features, improvements to the user interface, and the implementation of discount management functionality. The system was developed using the CodeIgniter 4 framework to ensure better process integration and operational efficiency. Black-box testing validated that all system components functioned correctly. Furthermore, a User Acceptance Testing (UAT) evaluation using a Likert scale produced a feasibility score of 85.2%, categorized as "Very Good." These findings indicate that the implemented system significantly improves operational efficiency, increases data accuracy, and successfully supports the digitalization efforts at Amanah Advertising.

Corresponding Author:

Annisa Dita Putri Kartika
Department of Information Systems, Faculty of Computer Science
Universitas Pembangunan Nasional "Veteran" Jawa Timur, Indonesia
Email: annisadpk1727@gmail.com

1. Introduction

Technology has become an integral part of modern life, especially as society increasingly relies on efficient and practical systems [1]. Technological advancements have driven changes across sectors, including business and industry, to improve performance and efficiency [2]. In a competitive environment, organizations must operate more effectively at strategic, operational, and tactical levels. Information systems play a critical role in supporting decision-making by providing accurate and relevant data [3].

In business, information systems connect people with computer-based technology using centralized databases. One key type is transaction management information systems, which help streamline business processes. Transaction management is vital as it serves as the financial hub of a business, representing the exchange between buyer and seller [4]. A robust transaction system can improve customer experience, reduce data errors, enable better decision-making, and minimize financial losses [5].

Web-based management systems are increasingly adopted by companies and MSMEs for their ability to improve sales management and transaction speed [6]. However, many businesses, especially MSMEs, still face challenges in adopting integrated systems. According to Kemenkop UKM data (2022), Indonesia has 65.4 million MSMEs, with 83.8% initiating digitalization efforts. Yet, 70.2% encounter obstacles such as limited funding, raw material access, and technology adoption [7]. Many MSMEs have not fully implemented digital systems in their operations [8].

The printing industry faces similar challenges. Businesses like PT MYID Print Indonesia still use spreadsheets, leading to inefficiencies and data loss risks as operations grow more complex [9]. CV. Cahaya Printing, though aided by Microsoft Excel, still manages reports manually, increasing error potential [10]. Radjawali Digital Printing continues to rely on handwritten records, which are time-consuming and vulnerable to damage or loss [11].

Amanah Advertising faces comparable issues. Based on interviews with the owner, problems include unrecorded material usage, delayed customer payments, and lack of structured production tracking [12]. Poor coordination between production and admin teams further hampers operations [13]. A tracking system would enable better monitoring and organization [14].

To solve these issues, a web-based transaction management information system is needed to automate orders, payments, material usage, and reporting. The system would document transactions in real-time, improve efficiency, and facilitate better coordination. Previous studies show that digital systems improve operational performance in printing businesses. For example, Wonder Studio reported increased efficiency and user satisfaction after digital adoption, with a satisfaction rate of 79.9% [15].

This study adopts the Prototype method to design a system aligned with user needs through iterative feedback [16]. The system was developed using the CodeIgniter framework, chosen for its lightweight performance and fast execution, making it suitable for resource-efficient development [17]. To ensure functional accuracy, the system underwent Black-box Testing, which evaluates system behavior based on requirements without inspecting internal code [18][19]. This method is ideal for non-programmers and practical for this research context [20].

The main focus of this study is to build a web-based transaction and inventory management system for Amanah Advertising. The system aims to overcome manual recording issues, reduce transaction errors, and improve overall efficiency in the digital era.

2. Research Method

This study employs a qualitative case study approach centered on Amanah Advertising. The system development process adopts the Prototype model, which emphasizes iterative development and continuous user feedback. This model is particularly suitable for projects that require a strong alignment between user expectations and system functionality.

In the Prototype approach within the Software Development Life Cycle (SDLC), an initial version of the system (a prototype) is built to demonstrate basic functionality and gather user input. Based on the feedback received, the prototype is refined through multiple iterations until it evolves into a final product that effectively meets user needs [21]. As a type of evolutionary process model, the Prototype model allows for ongoing modifications, enabling the software to grow in complexity and accuracy with each iteration [22].

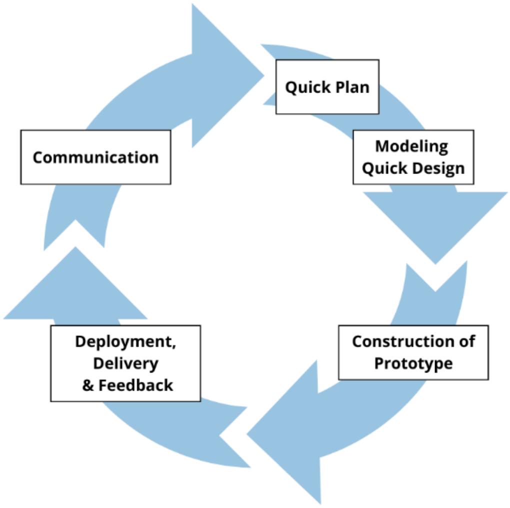

Figure 1. Illustrates the stages involved in the prototyping method.

The Prototype methodology used in this research consists of several structured stages that enable the system to evolve gradually through user input and repeated refinement:

2.1. Communication

In this initial stage, system requirements were gathered through interviews and direct observations at Amanah Advertising. The researcher examined the existing transaction processes, including order management, raw material tracking, and customer interactions. These insights were used to formulate a clear understanding of user needs and system objectives[16].

2.2. Quick Plan

Based on the findings from the communication phase, a high-level plan was created. As part of this early planning process, the developer designed preliminary elements such as wireframes, interface sketches, and flowcharts to present a clear overview of the system. These components were rapidly developed based on the previously gathered requirements analysis [23]. Among these, a cross-functional flowchart was specifically chosen because it effectively visualizes the roles and responsibilities of different actors involved in each process, making it ideal for systems with multiple user roles. This flowchart illustrated the proposed system workflow and served as a foundation for the design phase, ensuring alignment with the user's expectations.

2.3. Modeling Quick Design

This phase involved developing a more detailed system design using various modeling tools. Modeling quick design is a stage that focuses on creating representations of system components that will later be presented to the user [24]. In this stage, the system design was further refined using modeling tools such as Unified Modeling Language (UML) diagrams, including Use Case Diagrams and Activity Diagrams [25], as well as Class and Sequence Diagrams to visualize system components and their interactions. Additionally, the Entity-Relationship Diagram (ERD), Conceptual Data Model, and Physical Data Model were constructed to define the database structure and relationships.

2.4. Construction of Prototype

The initial design served as the foundation for building the prototype, showcasing the software aspects that would be visible to end users, such as the user interface layout and display formats [26]. An illustrative prototype was then developed using Figma, focusing on the system's user interface and navigation. This prototype allowed stakeholders to visualize the system's appearance and usability, serving as a communication bridge between developers and users.

2.5. Deployment, Delivery, and Feedback

The prototype was presented to stakeholders for demonstration and evaluation. Feedback was gathered concerning interface usability, navigation flow, and feature functionality. Based on this input, improvements were made iteratively until the prototype satisfied user expectations.

2.6. System Coding

Upon approval, the prototype was translated into a functional system using the CodeIgniter 4 framework. CodeIgniter was selected due to its lightweight structure, high performance, and support for the MVC architecture, which facilitates modular code management and efficient development.

2.7. System Testing and User Evaluation

To validate functionality, the system underwent Black Box Testing, which checks whether outputs align with expected results based on given inputs—without inspecting internal code. This type of testing is effective for verifying end-user functionality from an external perspective.

Following functional testing, a User Acceptance Testing (UAT) phase was conducted to evaluate whether the system met real-world user needs. UAT was integrated into the testing process and served as the final stage before system deployment. User Acceptance Testing (UAT) was conducted in three main stages [27]. In the preparation stage, test scenarios were developed based on essential system features such as login, order processing, and report generation. A UAT questionnaire was also prepared, which included system usage instructions, task descriptions, and a 5-point Likert scale designed to measure user satisfaction in terms of functionality, performance, interface experience, and productivity. The Likert scale used in the UAT questionnaire assigned the following weights to each response option [28], as shown in Table 1.

Table 1. Likert Scale Scoring Criteria

Score	Description
1	Strongly Disagree
2	Disagree
3	Neutral
4	Agree

During the execution stage, end users performed specific tasks based on the scenarios provided. After completing each task, they filled out the UAT form according to their experience, allowing the development team to assess how well the system met actual business requirements. In the evaluation stage, the collected data were analyzed to determine the system's acceptance level. The UAT score was calculated by first computing the mean, obtained by dividing the total weighted score by the number of respondents. The UAT score was calculated by first determining the mean, which was obtained by dividing the total weighted score by the number of respondents, as shown in Equation (1). Then, the mean score was converted into a percentage by dividing it by the maximum possible score and multiplying the result by 100%, as shown in Equation (2). These calculations were used to assess the results of the User Acceptance Test for each evaluation criterion.

$$Mean = \frac{Total\ Weighted\ Score}{Number\ of\ Respondents} \tag{1}$$

$$Persentage = \frac{Mean\ Score}{Maximum\ Score} \times 100\% \tag{2}$$

The final percentage score was interpreted using predefined criteria, as shown in Table 2. These criteria provide a qualitative description of the system's acceptability based on the percentage score obtained from the User Acceptance Test.

Table 2. UAT Score Interpretation Criteria

Percentage Range	Description
0% – 20%	Very Poor
21% – 40%	Poor
41% – 60%	Fair
61% – 80%	Good
81% – 100%	Very Good

The evaluation results were then used as a reference to proceed to the system coding phase [26]. The iteration process occurred when the developer made adjustments and improvements to the prototype based on the feedback received. The iteration process was considered complete once the final UAT score interpretation reached above 81%, which falls into the "Very Good" category.

3. Result and Discussion

This chapter presents the final outcomes of the system development process, which was conducted based on the stages described in the previous chapter. The developed system aims to support operational activities at Amanah Advertising by addressing the needs identified during the system analysis phase.

The system was developed using the prototyping method, which consists of five main stages: Communication, Quick Planning, Modeling (Quick Design), Construction of Prototype, and Deployment, Delivery, and Feedback. The development process began with the creation of an initial prototype based on early user requirements and analysis. After the prototype was demonstrated to users, feedback was collected and used for refinement through several iterations. This iterative approach allowed the system to be gradually improved to better meet user needs and expectations. The continuous user involvement at each stage ensured that the final system was both functional and user-oriented.

3.1. Initial Development

The initial development phase of the system was carried out using the prototyping methodology, which comprises five iterative stages: Communication, Quick Planning, Modeling (Quick Design), Construction of Prototype, and Deployment, Delivery and Feedback. This method was chosen due to its user-centered nature, allowing stakeholders to be actively involved from the early design stage to ensure the system aligns with their actual operational needs.

3.1.1 Communication

In the Communication stage, researchers conducted in-depth interviews with stakeholders at Amanah Advertising to capture the actual business processes and identify existing problems. The analysis revealed that the transaction and order processes were still conducted manually, resulting in inefficiencies such as slow order tracking, unclear responsibilities between roles, and limited managerial access to reports. Based on this, the system's functional and non-functional requirements were formulated and categorized based on user roles namely Admin, Production Team, and Manager.

The functional requirements were classified by user roles. Admins are responsible for managing transactions, tracking orders, handling product and stock data, and maintaining reseller and supplier records. Production Teams handle order execution, including confirming designs, tracking production status, logging material usage, and reporting failures. Managers oversee operational data, manage users, adjust product pricing, and export reports for strategic decisions. In addition, the system must meet several non-functional requirements such as secure authentication, a user-friendly interface, high availability, and web-based accessibility to support seamless and reliable operations.

3.1.2 Quick Plan

Proceeding to the Quick Planning phase, a new system flow was designed to support a structured order and inventory management system. The proposed process aims to automate the transaction recording, production coordination, and reporting features. This proposed workflow is visualized in Figure 2., which outlines how each actor interacts with the system from order input to product pickup, including payment confirmation.

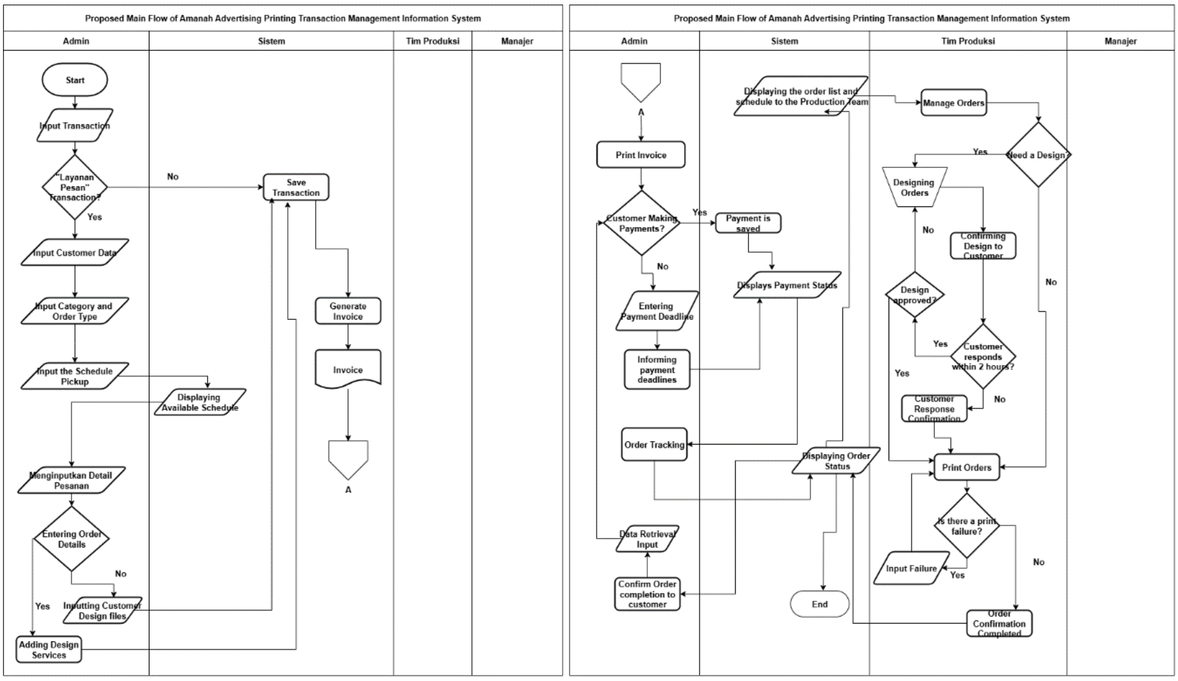

Figure 2. System Workflow of Order Processing and Inventory Management

3.1.3 Modelling Quick Plan

The Modeling stage focused on designing the system architecture and data structure using UML-based modeling. The use case diagram Figure 3 was developed to visualize functional interactions between users and the system.

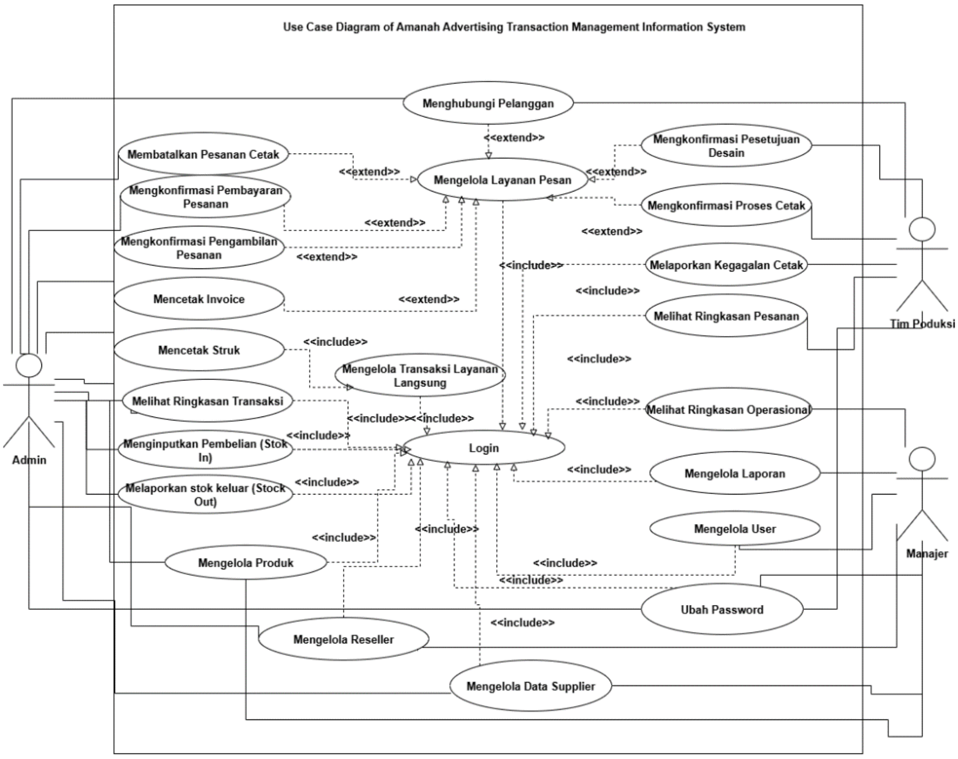

Figure 3. System Use Case Diagram

Furthermore, the class diagram (Figure 4) was used to model the internal structure and relationships between entities such as users, products, orders, and stock history.

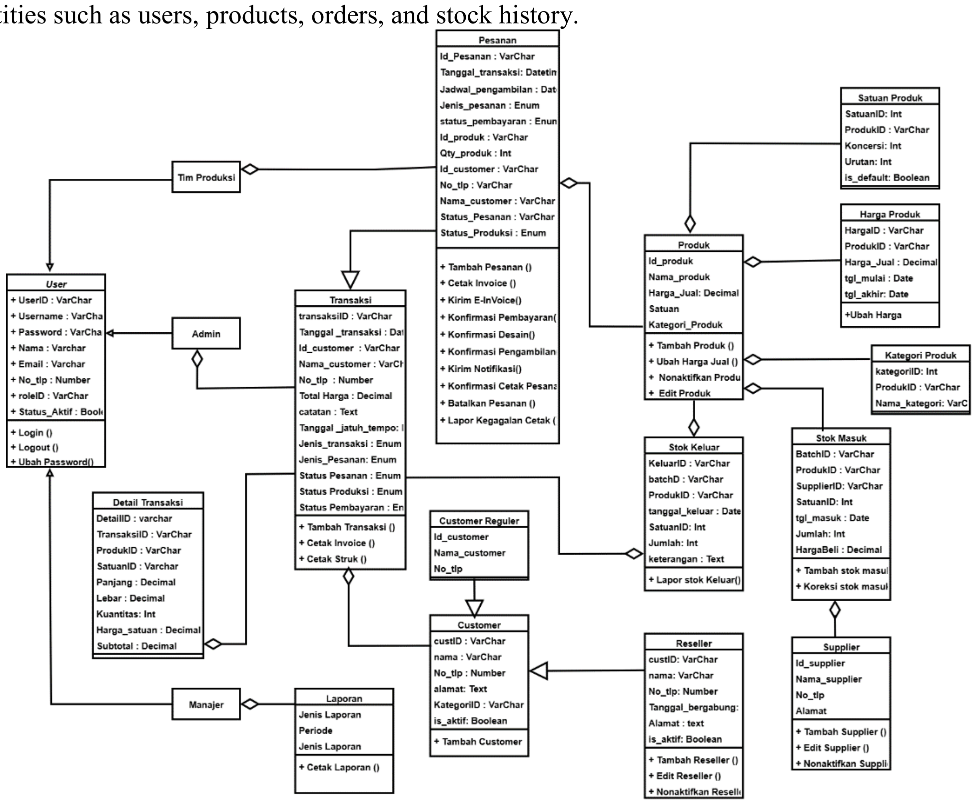

Figure 4. Class Diagram of the Order and Inventory Management System

To capture the system's behavior, Several activity diagrams and sequence diagrams were produced to illustrate the system's overall workflows. These diagrams reflect the complete process flow from user interaction to system responses. Additionally, the data model was structured into two levels: the Conceptual Data Model (CDM) and the Physical Data Model (PDM), ensuring consistency between the system's logic and its database structure.

One of the key processes modeled is the "Managing *Layanan Pesan* Transactions", as shown in Figure 5. This activity diagram outlines the admin's interaction with the system, starting from login, accessing the order menu, inputting transaction data, and tracking orders. The system responds by displaying forms, generating invoices, and updating order statuses based on admin actions. This ensures efficient and structured order management.

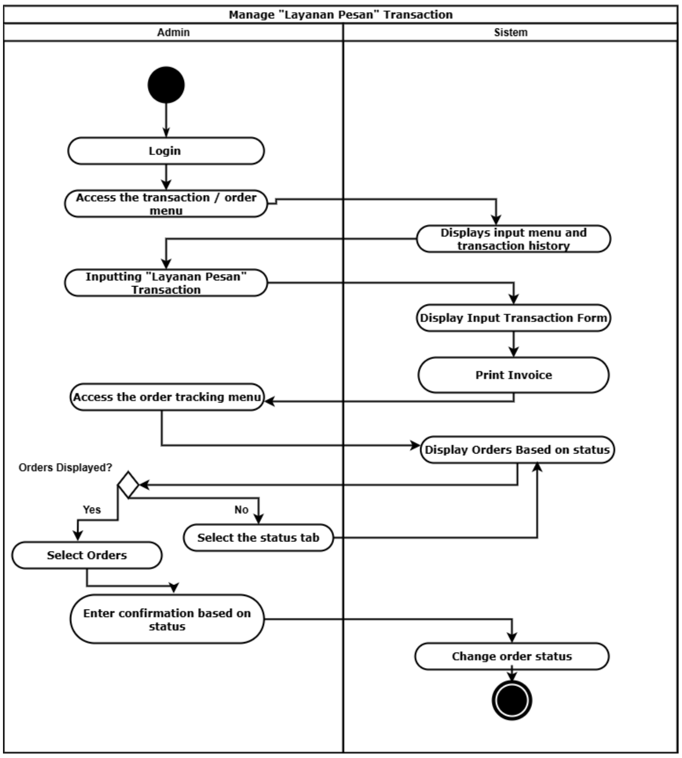

Figure 5. Admin Activity Diagram for Managing Layanan Pesan Transactions

Another important system interaction is illustrated in the sequence diagram for "Reporting Print Failures" as shown in Figure 6. This diagram depicts how the Production Team accesses the failure report menu, views the list of past failures, and submits a new failure entry. The system processes the input, stores the data, and returns a success message, which is then displayed to the user as a toast notification.

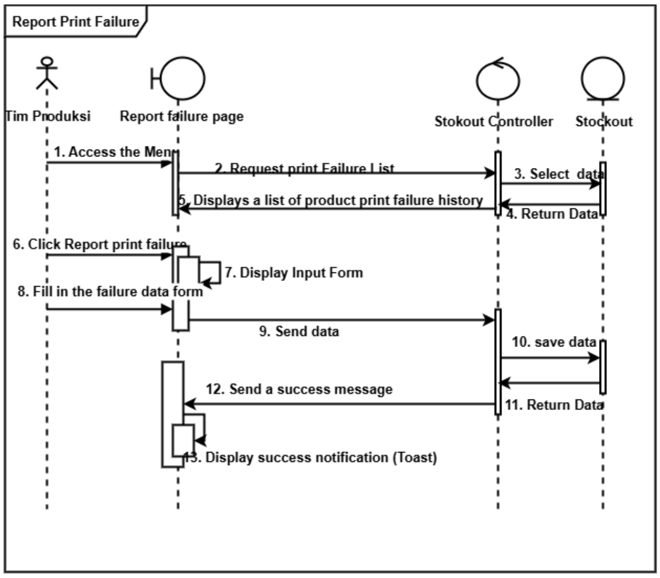

Figure 6. Sequence Diagram Depicting Print Failure Reporting Process

To represent the system's data structure and relationships between key entities, a Conceptual Data Model (CDM) was designed, as shown in Figure 7. This model outlines the main entities such as Users, Orders, Products, and Transactions along with their relationships, ensuring a clear foundation for database development and consistent data handling across all modules of the system.

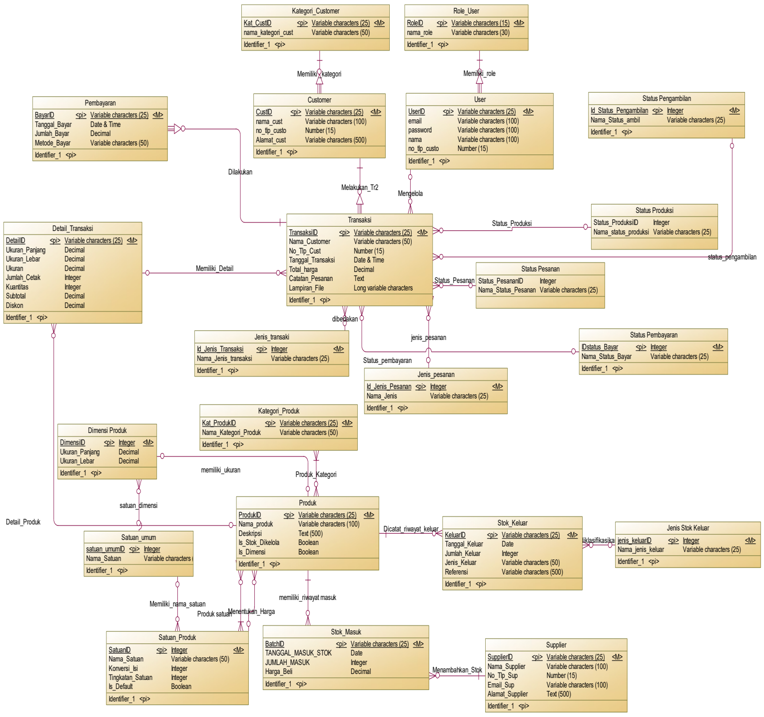

Figure 7. CDM Illustrating Key Entities and Their Relationships

3.1.4 Construction Of Prototype

During the Construction of Prototype phase, a high-fidelity interactive prototype was developed using *Figma*. The prototype was designed based on user access roles and included major system pages such as the login interface, transaction form, product management, and reporting dashboards. This initial version focused on visual representation of functionalities without full backend implementation. Examples of the interface can be seen in *Figure 8*.

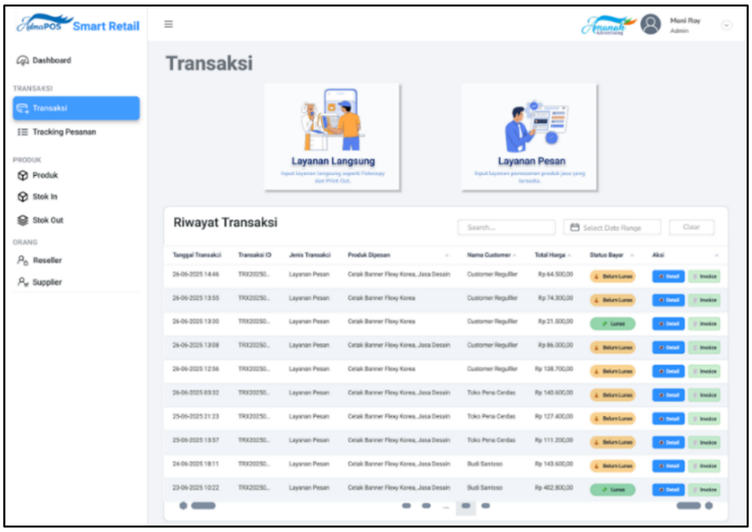

Figure 8. High-Fidelity Interactive Prototype Interfaces Designed in Figma

In addition to the main transaction page, an interface was also developed for the "Layanan Pesannya" (Custom Order Service) transaction input feature, which is one of the specialized services in the system. This page is designed to allow users to easily select the type of service, enter customer request details, and automatically calculate the total cost. The layout prioritizes ease of navigation and data entry efficiency based on the user's access role. The interface is presented in Figure 9.

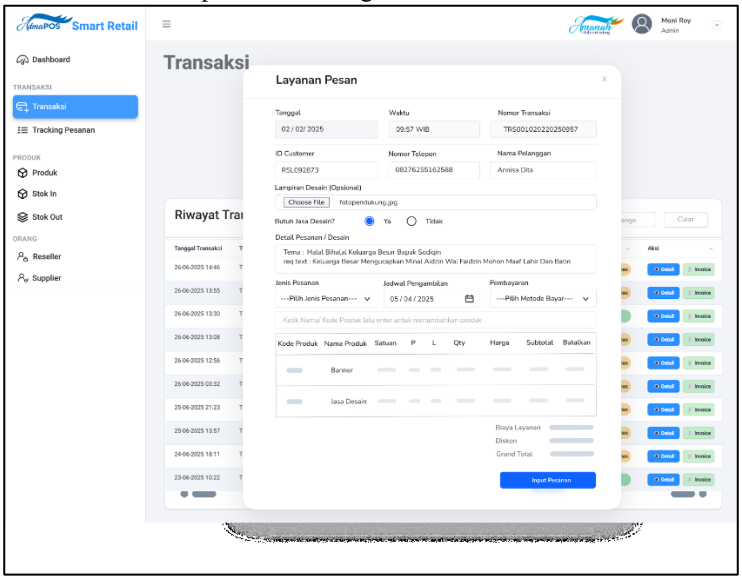

Figure 9. Transaction Input Interface for "Layanan Pesannya"

3.1.5 Deployment, Delivery and Feedback

In the final stage, the prototype was presented to users for evaluation through direct testing. Users provided valuable feedback aimed at improving interface clarity, system flow, and overall functionality. Several enhancements were suggested, including displaying the user responsible for order entry (Admin or Production Team) and showing the pickup schedule to improve tracking transparency.

The production team requested the ability to input orders without requiring payment details, better reflecting real workflow conditions. In addition, users proposed adding a confirmation step for failed print jobs after recording material usage, as well as including a summary of failures for monitoring purposes.

Managers highlighted the need for more control features, such as setting product discounts, managing tiered pricing for resellers, exporting receivable reports, and logging stock discrepancies. These inputs formed the foundation for the next iteration, focused on refining key features, improving usability, and ensuring the system meets the operational needs of each user role.

3.2. Iteration and Refinement

This section describes the iterative process carried out after the initial prototype was evaluated by users. Feedback collected during testing sessions was used to improve features, enhance usability, and ensure the system aligned more closely with operational needs. Each iteration focused on addressing specific issues, adding new features, and optimizing user experience.

3.2.1. Iteration 1

During the evaluation of the initial prototype, users provided valuable feedback highlighting several functional gaps and improvement areas. One of the most emphasized concerns came from the managerial role, which requested the ability to set flexible discount schemes not only for products but also customized per reseller. This capability was considered essential to support dynamic pricing strategies and promotional needs in the printing business. The production team also expressed the need for greater autonomy, such as the ability to input orders independently without payment information and to download design attachments for offline processing. Additionally, feedback from both admins and production staff indicated that the order tracking module lacked critical information such as who input the order and when the customer was scheduled to pick it up.

In the first iteration, the development began again from the communication stage, as the user roles remained unchanged but feedback indicated specific areas for improvement. One of the most emphasized aspects by users, especially managers, was the need for flexible discount management both at the product level and tailored for each reseller. This feedback drove major adjustments to the system's models and interface.

To accommodate these requirements, the system design was updated starting from the Use Case Diagram, which included a new use case: "Manage Discounts" for managers. The updated Use Case Diagram, shown in Figure 10. This addition supports the configuration of promotional pricing for both individual products and resellers. As illustrated in the diagram, this enhancement improves system flexibility and enables more dynamic business strategies.

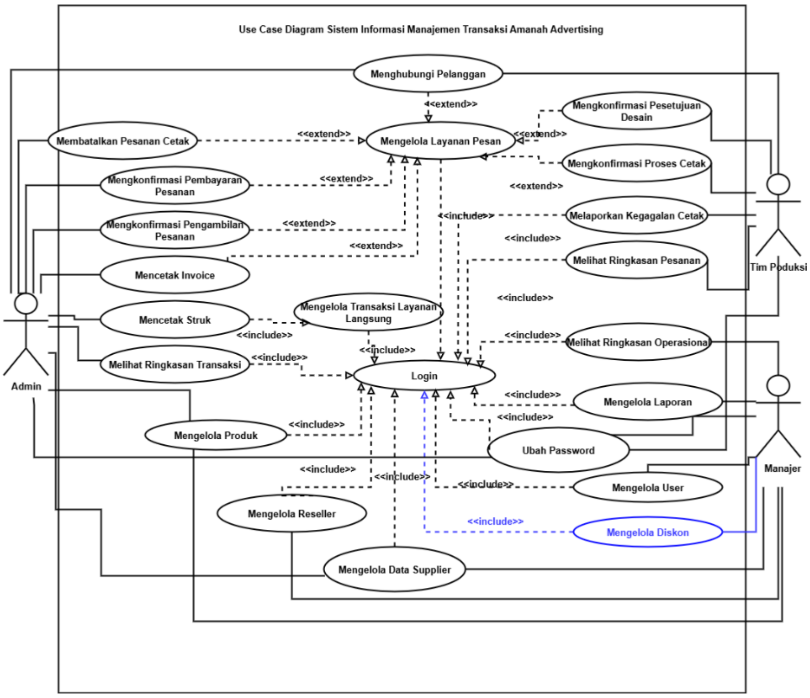

Figure 10. Updated Use Case Diagram Including "Manage Discounts"

This addition reflects their ability to configure promotional prices based on product types or reseller levels. Correspondingly, the Class Diagram was expanded by adding the Diskon, Diskon_Produk, and Diskon_Reseller classes, establishing relationships with both the Produk and Customer classes. These structural changes are essential for implementing multi-tiered discount functionality.

Adjustments were also made to the Activity Diagram, Sequence Diagram, as well as the Conceptual and Physical Data Models (CDM and PDM) in response to user feedback. These revisions ensured that new requirements—such as discount management and additional transaction flows—were accurately represented and aligned with the updated system functionality.

At the interface level, managers gained access to a new UI for managing product discounts and a dedicated interface for configuring reseller-based pricing (*Figure 11*).

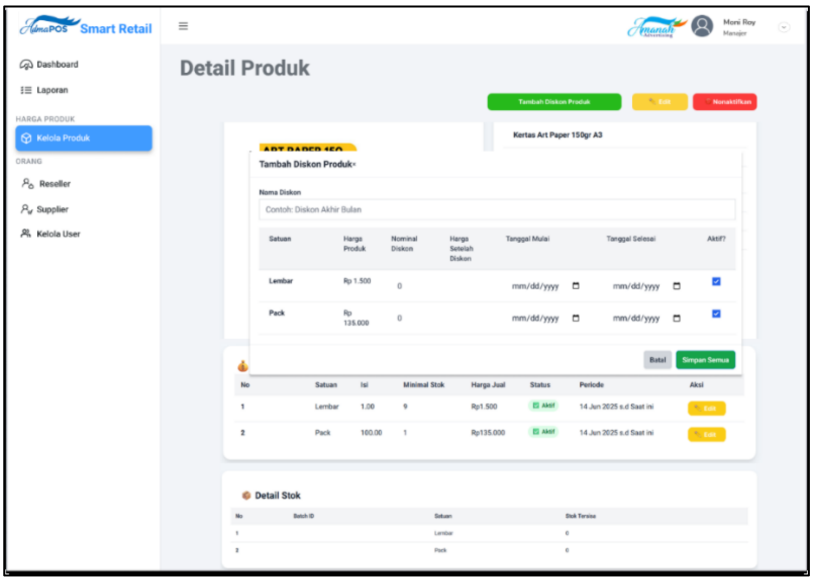

Figure 11. User Interface for Managing Product and Reseller Discounts by Manager

These changes were supported by new database tables that capture discount configurations with effective dates and activation status.

Other improvements implemented during Iteration 1 included: displaying the order creator and pickup schedule in the order tracking module; enabling the production team to input orders without requiring payment; and adding download capability for design attachments. To support autonomous workflows, a simplified order input form was developed specifically for the production team, allowing them to input orders directly without accessing payment fields. Additional features were introduced, such as the ability to mark orders as "not picked up," search through order records, confirm failed print jobs, and report stock discrepancies. These enhancements led to necessary changes across the process flow diagrams, UI components, and database structure.

After the first iteration prototype was delivered and demonstrated to users, further feedback was gathered to enhance the system. This feedback indicated that the manager’s role should be extended to allow direct handling of order issues, such as cancellations, without needing admin intervention. Additionally, detailed information in the transaction view was requested to support more efficient monitoring. These suggestions formed the basis for further development in Iteration 2.

3.2.1. Iteration 2

In the second iteration, development focused on refining features based on user feedback collected during the prototype demonstration. Two key suggestions were prioritized: enabling managers to cancel orders directly and improving the level of detail displayed in the transaction history view. These enhancements aimed to support error correction in order input and streamline monitoring activities by the managerial role.

To address these needs, updates were made starting from the modeling stage. The *Use Case Diagram* was revised to extend cancellation access to managers, complementing the previous admin-only functionality. As shown in Figure 12, the revised Use Case Diagram expands the system's flexibility by allowing managers to directly cancel customer ordersa task previously limited to admin users. This update responds to user feedback requesting broader managerial control over transaction flows, enabling faster issue resolution and reducing dependency on the admin role. The new use case “Cancel Order” is now assigned to both the Manager and Admin, reflecting this expanded access and enhancing operational efficiency.

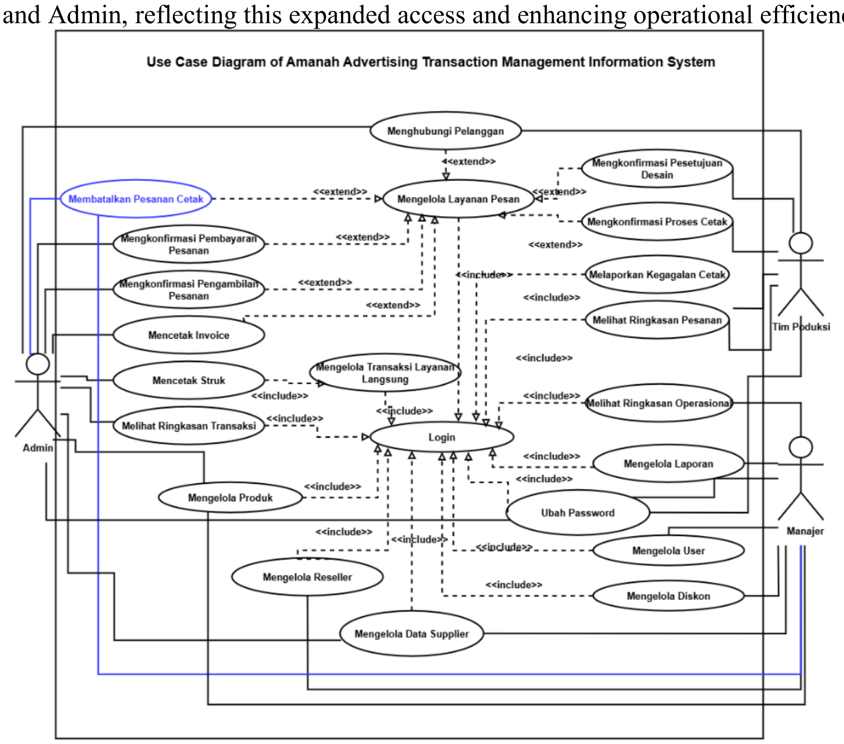

Figure 12. Revised Use Case Diagram

The *Activity Diagram* for canceled order was adjusted to illustrate the cancellation flow initiated by managers, including verification steps and direct execution. As shown in Figure 13, the updated Activity Diagram outlines the revised cancellation flow that allows managers to independently verify and execute order cancellations. This adjustment ensures that the cancellation process follows a structured and accountable path, reducing delays caused by inter-role dependency.

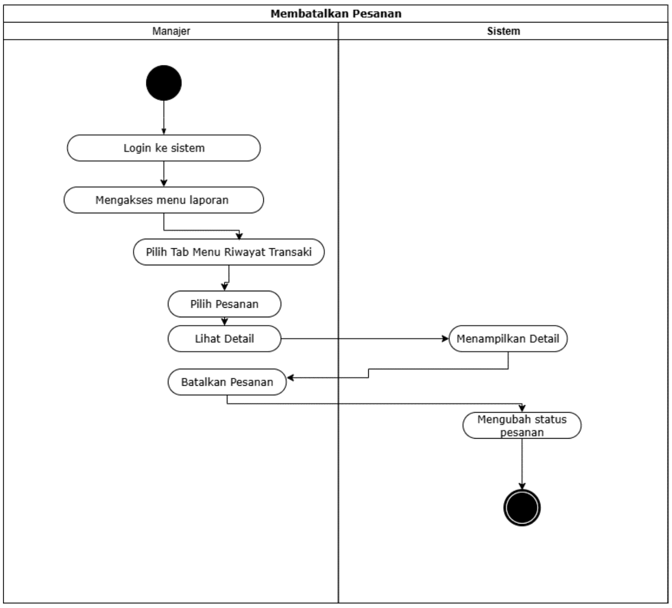

Figure 13. Updated Activity Diagram for Order Cancellation Process

Meanwhile, Figure 14 presents the corresponding Sequence Diagram, which illustrates the step-by-step interaction between the manager, system, and database components during the cancellation process. This sequence visualizes how the system validates managerial authority, updates the transaction status, and records cancellation history supporting accurate tracking and accountability.

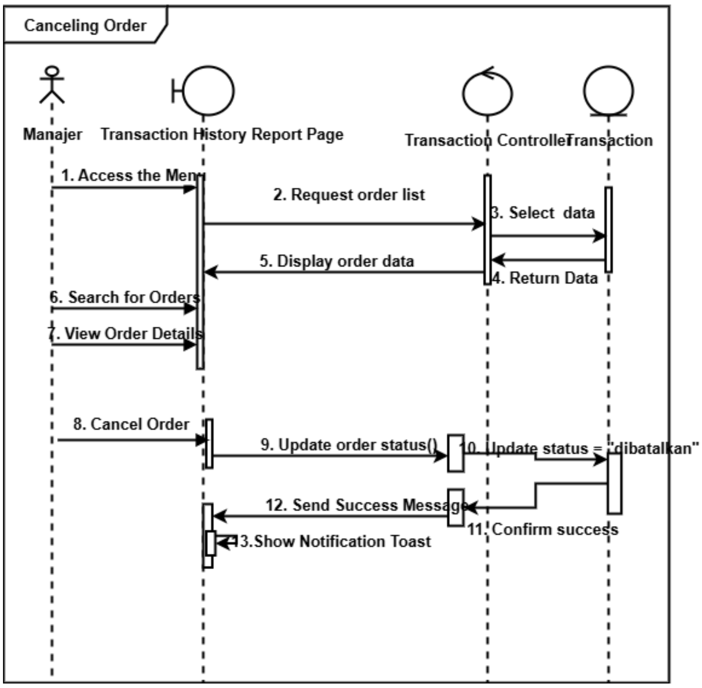

Figure 14. Sequence Diagram of the Order Cancellation Interaction

While these changes impacted system permissions and interface behavior, they did not require modifications to the data structure. Thus, the Conceptual and Physical Data Models (CDM and PDM) remained unchanged from the previous iteration.

At the interface level, the transaction history view was enhanced with a new “View Details” action button for managers. This allowed quick access to comprehensive transaction data such as ordered items, payment status, and the user responsible for input. On the transaction detail page, a “Cancel Order” button was added, restricted to users with a managerial role. These adjustments increased managerial control while maintaining the overall system flow from the first iteration, as illustrated in Figure 15.

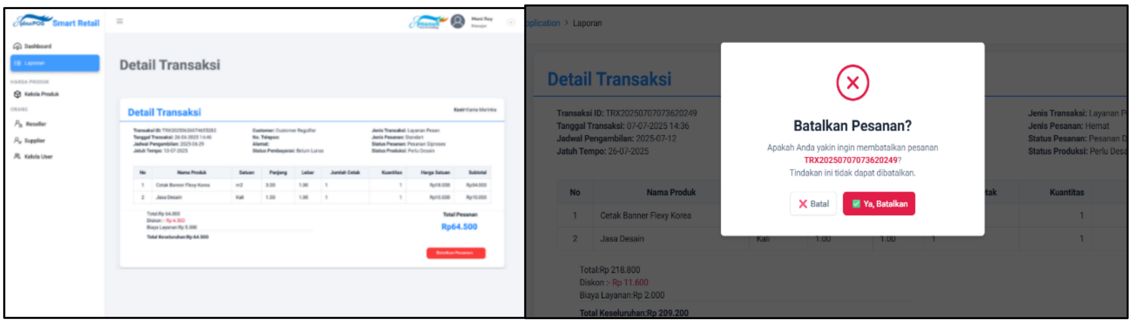

Figure 15. Enhanced Transaction History Interface

After the updated prototype was demonstrated to users, the feedback was positive. Users confirmed that the added functionalities met their expectations, and no further changes were requested. As a result, the system was considered ready for full implementation and integration into the final development and testing phase.

3.3. System Coding

Once the prototype was confirmed to meet the requirements, the next stage involved transforming the design into a fully functional system. The coding process was carried out using previously planned technologies and frameworks, focusing on implementing core features based on user needs. At this stage, the previously simulated interface and flows were developed into interactive features.

All elements in the user interface were linked to backend logic, allowing users to perform actions according to their roles and access rights. Key functionalities such as password updates, data export for suppliers, resellers, and users, as well as the generation of transactional and financial reports, were implemented in real-time. This marked the system's transition from a static prototype to a dynamic platform capable of handling real business processes.

3.4. System Testing and Evaluation

System testing was conducted to ensure all features functioned as intended. The black-box testing method was used, focusing on testing functionality without evaluating the internal code. Tests covered user authentication, role-based data management (Admin, Production Team, Manager), transaction flows, and reporting processes. Results indicated that all features responded as expected, with no significant issues found in system performance or user access. Thus, the system was deemed ready for implementation.

To validate system effectiveness and user satisfaction, a final evaluation was conducted using the User Acceptance Testing (UAT) method. This involved direct testing by seven users representing each role: 1 Director, 1 Manager, 3 Admins, and 2 Production Team members. Each participant completed a Likert-scale evaluation form based on functional and non-functional requirements. The UAT form consisted of 25 questions grouped into four variables: system functionality, performance, user interface, and efficiency. Each item was rated on a five-point scale. Scores were then calculated using a weighted average formula to determine the system's acceptance level. The detailed evaluation results are summarized in Table 3, which presents the mean scores and acceptance percentages for each question. These results provide quantitative insight into user satisfaction across the evaluated system variables.

Table 3. Summary of UAT Scores and Acceptance Percentages per Evaluation Variable

Code	Statement	Mean	Percentage
A1	I can add and update data without issues.	4.14	82.8%
A2	Reports generated by the system are accurate.	4.29	85.8%
A3	The system produces complete and reliable information.	4.00	80.0%
A4	I can easily search and find specific data.	4.29	85.8%
A5	The search or filter feature works well.	4.14	82.8%
A6	The system correctly processes all inputs.	4.14	82.8%
B1	The system is always available when needed.	4.14	82.8%
B2	The system's response speed is excellent.	4.57	91.4%
B3	Data is always saved correctly.	4.29	85.8%

Code	Statement	Mean	Percentage
B4	The system features are sufficient for my work.	4.14	82.8%
C1	The system runs without disruption.	4.14	82.8%
C2	Information is easy to read and understand.	4.29	85.8%
C3	The color scheme is appropriate.	4.43	88.6%
C4	Displayed information meets expectations.	4.00	80.0%
C5	The interface is easy to understand.	4.14	82.8%
C6	I feel comfortable using the system.	4.14	82.8%
C7	The system improves my efficiency.	4.43	88.6%
C8	Easy navigation and responsive input.	4.43	88.6%
C9	Menu and button layout is appropriate.	4.29	85.8%
D1	The system saves time in completing tasks.	4.14	82.8%
D2	I am satisfied with the system's automation.	4.29	85.8%
D3	The system reduces manual workload.	4.57	91.4%
D4	Helps minimize common work errors.	4.43	88.6%
D5	Supports team collaboration.	4.14	82.8%
D6	Improves decision-making accuracy and speed.	4.43	88.6%
	Overall Average	**-**	**85.2%**

Evaluation results showed mean scores ranging from 4.00 to 4.57 and acceptance percentages between 80% and 91.4%. The overall average score was 85.2%, which falls into the "Very Good" category. This indicates a high level of satisfaction with the system.

The highest-rated items were system responsiveness (B2) and reduced manual workload (D3), each scoring 4.57 (91.4%). The lowest-rated items, though still acceptable, were system completeness (A3) and information accuracy (C4), each scoring 4.00 (80%).

With no questions falling into the "Fair" or "Poor" categories, the evaluation confirms that the developed system meets users' expectations in terms of functionality, performance, usability, and productivity. Therefore, the Transaction Management Information System is considered suitable for full implementation at Amanah Advertising.

4. Conclusion

Based on the research and development conducted, it can be concluded that the design and development of the web-based Transaction Management Information System for Amanah Advertising using the Prototype method successfully addressed user needs and improved operational efficiency in the workplace. The system was developed incrementally through two iterations based on user feedback. The first iteration focused on the addition of core features such as discount settings, stock management, and failure reporting, which led to changes in the interface and database structure. The second iteration emphasized user access and information presentation, including features like order cancellation by the manager and detailed transaction displays. Once no further feedback was provided, the system proceeded to the coding stage using the CodeIgniter 4 framework.

System testing was conducted using the black-box testing method, in which all test scenarios were executed and met the expected outcomes without any functional errors. Further evaluation through User Acceptance Testing (UAT) involved eight users from different roles, resulting in an average feasibility score of 85.2% and a “Very Good” category. This indicates that the system has successfully met user requirements and is ready for full implementation at Amanah Advertising.

References

[1] N. Nabillah Purba, Mhd Yahya, “REVOLUSI INDUSTRI 4.0 : PERAN TEKNOLOGI DALAM EKSISTENSI PENGUASAAN BISNIS DAN IMPLEMENTASINYA,” JPSB J. Perilaku Dan Strateg. Bisnis, vol. 9, no. 2, pp. 91–98, 2021.

[2] S. I. Dawam Sahrin Najah. S, Rusdi Hidayat N, Maharani Ikaningtyas, Muhammad Ryan N.z, “PERAN TEKNOLOGI DALAM TRANSFORMASI EKONOMI DAN BISNIS DI ERA DIGITAL,” J. Ilm. Res. Student, vol. 1, no. 5, pp. 8–16, 2024.

[3] H. F. Andreas Rian Dwi S, “Sistem Informasi Penjualan Pada Cendana Digital Printing Station Menggunakan Metode FIFO,” vol. 8, no. 2, pp. 167–173, 2023.

[4] R. Sulastri et al., “PERANCANGAN SISTEM MANAJEMEN KASIR POINT OF SALE BERBASIS WEB DENGAN METODE AGILE DEVELOPMENT SCRUM (STUDI KASUS : WARUNG UKM ‘ TOKO DEWI ’) WEB-BASED POINT OF SALE CASHIER MANAGEMENT SYSTEM DESIGN USING AGILE,” vol. XIX, no. 01, pp. 11–20.

[5] J. Teknologi and S. Informasi, “Analisis Dan Perancangan Sistem Informasi Point O f Sale Pada PO Indocahaya,” vol. 5, no. August, pp. 34–45, 2024, doi: 10.35957/jtsi.

[6] U. R. Agus Mulyana, “RANCANG BANGUN SISTEM INFORMASI POINT OF SALE (POS) BERBASIS WEB (STUDI KASUS TOKO ANDORIO) AGUS,” Maj. Ilm. UNIKOM, vol. 21, no. 1, pp. 43–50, 2023.

[7] M. M. Dr. LESTARI MOERDIJAT S.S., “Segera Atasi Kendala dalam Transformasi Digital di Sektor UMKM,” https://mpr.go.id/.

[8] R. N. Hayati, H. Lalu, and Y. Ardiansyah, “lmplementasi Sistem Informasi Manajemen Dalam Pemasaran Usaha Mikro Kecil Menengah,” vol. 02, no. 02, 2024.

[9] M. A. M Salman Alfsrisi, “PERANCANGAN SISTEM INFORMASI PERCETAKAN (SICETAK) PADA PT MYID PRINT INDONESIA,” Semin. Teknol. Majalengka, vol. 8, 2024.

[10] R. R. Setiawan, A. P. Utomo, P. Setiaji, and E. Pangestu, “Pengelolaan Efisien Bahan Baku dengan Sistem Informasi Manajemen Persediaan Berbasis Web pada CV Cahaya Printing,” pp. 43–51, 2024.

[11] D. Candra, P. Yuda, A. Susilo, Y. Irawan, and E. H. Nurkifli, “RANCANG BANGUN SISTEM POINT OF SALES WEB FRAMEWORK LARAVEL PADA PERCETAKAN RADJAWALI DIGITAL,” vol. 12, no. 3, pp. 2434–2443, 2024.

[12] P. Cv and B. Cemerlang, “Perancangan Sistem Tracking Produksi Berbasis Website Menggunakan Framework Codeigniter,” vol. 1, no. 2, pp. 99–122, 2023.

[13] K. N. Tania, “PERAN STAF ADMINISTRASI DALAM PENDATAAN HASIL PRODUKSI di PT RYU EI KOGYO KOTA TANGERANG,” J. Ilm. ILMU Sekr. PERKANTORAN, vol. 9, no. 1, 2022.

[14] M. R. Stephany and A. F. Hadining, “Analisis Sistem Penjadwalan Produksi Berdasarkan Pesanan Pelanggan Dengan Metode Sequencing Pada PT XYZ,” J. Tek. Ind., vol. 8, no. 2, pp. 1–8, 2022.

[15] D. R. Ramadhan and N. Pratiwi, “Rancang Bangun Sistem Informasi Berbasis Web Pada Perusahaan Percetakan”.

[16] B. K. Farindra, R. Insani, P. Anaking, C. Diagram, and S. Diagram, “RANCANG BANGUN SISTEM INFORMASI MANAJEMEN LAYANAN PUSAT BAHASA BERBASIS WEB DI,” vol. 6, no. 3, pp. 616–631.

[17] R. R. Prasena, H. Sama, and U. I. Batam, “STUDI KOMPARASI PENGEMBANGAN WEBSITE DENGAN FRAMEWORK CODEIGNITER DAN LARAVEL,” vol. 1, pp. 613–621, 2020.

[18] M. T. Abdillah, I. Kurniastuti, and F. A. Susanto, “Implementasi Black box Testing dan Usability Testing pada Website Sekolah MI Miftahul Ulum Warugunung Surabaya,” vol. 8, pp. 234–242, 2023.

[19] S. D. Pratama and M. N. D. , Lasimin, “Pengujian Black Box Testing Pada Aplikasi Edu Digital Berbasis Website Menggunakan Metode Equivalence Dan Boundary Value Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD,” J. Teknol. Sist. Inf. dan Sist. Komput. TGD, vol. 6, pp. 560–569, 2023.

[20] I. J. Habibah Nurfauziah, “PERBANDINGAN METODE TESTING ANTARA BLACKBOX DENGAN WHITEBOX PADA SEBUAH SISTEM INFORMASI,” J. Vis., vol. 8, no. 2, pp. 105–114, 2022.

[21] R. S. U. Fitria Nur Hasanah, Rekayasa Perangkat Lunak. Sidoarjo: UMSIDA PRESS, 2020.

[22] L. Setiyani, Rekayasa Perangkat Lunak[Software Engineering], no. May. 2019.

[23] A. N. Athaya and N. L. Marpaung, “Rancang Bangun Aplikasi Bon Permintaan Dan Pengeluaran Barang Menggunakan Metode Prototype Berbasis Website,” J. Inform. J. Pengemb. IT, vol. 8, no. 2, pp. 134–141, 2023.

[24] W. Herwansyah and N. L. Marpaung, “Rancang Bangun Sistem Informasi Toko Izdiihar Laptop Berbasis Website,” pp. 484–497, 2023.

[25] F. Z. Badeni, A. Guntara, and I. Fadil, “Implementasi Sistem Informasi Manajemen Aset Kelompok Ternak Ikan Hias Sumedang Menggunakan Metode Prototype,” vol. 18, pp. 196–208, 2024.

[26] D. Ardiyansah et al., “IMPLEMENTASI METODE PROTOTYPING PADA SISTEM INFORMASI,” vol. 2, pp. 17–22, 2021.

[27] M. T. I Putu Agus Eka Pratama, S.T., PROTOTYPING SEBAGAI MODEL PENGEMBANGAN SOFTWARE. CV.RUANG TENTOR, 2023.

[28] N. Hartono and A. A. Muin, “Penggunaan User Acceptance Testing (UAT) Pada Pengujian Sistem Informasi Pengelolaan Keuangan Dan Inventaris Barang,” 2025.