Research Article

Building a Web-Based Application for Transaction Recording and Inventory Management at *Unipro* Store

[1,*] Peter, [2] Bagus Mulyawan

[1,2] Department of Information Systems, Faculty of Information Technology, Universitas Tarumanagara, Jakarta, Indonesia

* Corresponding Author: peter.825210022@stu.untar.ac.id

Abstract: Unipro Store, a retail shop specializing in cellphone accessories, faces significant challenges in recording transactions and managing inventory due to its reliance on manual processes. All transactions, whether purchases from suppliers or sales conducted through various channels such as WhatsApp, Tokopedia, Shopee, or in-store are still documented manually using a notebook. This manual system makes it difficult for shop owners to retrieve historical transaction data and increases the risk of data loss, as there is no backup if the notebook is lost or damaged. To address these issues, a transaction recording and inventory management application has been designed to streamline the management of product stock, sales, and purchase data. The application supports two types of users: the owner and the admin. Its development follows the Software Development Life Cycle (SDLC) using the waterfall model. During implementation, HTML, CSS, and JavaScript were utilized alongside the Bootstrap and ASP.NET frameworks to develop the application, with Microsoft SQL Server selected as the database solution. In the testing phase, user acceptance testing (UAT) was conducted using a black box testing approach, successfully passing all test scenarios. Additionally, a System Usability Scale (SUS) questionnaire was distributed, yielding a final score of 81.67, which falls under grade A in the usability assessment.

Keywords: Transaction, Inventory, Management, Application, System Usability Scale

Citation: Peter, & Mulyawan, B. (2024). Building a web-based application for transaction recording and inventory management at Unipro Store. *Iota*, 4(4). ISSN 2774-4353. https://doi.org/10.31763/iota.v4i4.827

Academic Editor : Adi, P.D.P

Received : Oktober, 12 2024

Accepted : November, 04 2024

Published : November, 25 2024

Publisher's Note: ASCEE stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

In today's digital era, the use of information technology has become indispensable for businesses seeking to enhance their efficiency and effectiveness. Information technology enables businesses to automate operational processes, reduce the likelihood of human error, and provide real-time data insights that facilitate better and more strategic decision-making [1]. For businesses engaged in product sales, the presence of an accurate and well-organized transaction recording system is crucial. Such a system ensures that transactions are systematically recorded, providing not only operational support but also valuable insights that can be used to optimize strategies and boost competitive advantage [2].

Unipro Store, founded in 2018, is a business engaged in selling cellphone accessories, including cases, screen protectors, data cables, chargers, earphones, and various other technological accessories. The store is located in Tangerang, Banten, which is also the residence of its owner. Unipro Store sources its products from various suppliers and resells them to consumers through multiple sales channels. In addition to offering direct purchase options at the physical store, Unipro Store accepts orders via WhatsApp and markets its products on e-commerce platforms such as Tokopedia and Shopee. This multi-channel approach allows the store to reach a broader customer base and increase sales opportunities.

Despite its operational success, Unipro Store faces significant challenges due to its reliance on manual transaction recording processes. All transactions, whether purchases from suppliers or sales through the store's various channels are recorded manually in a notebook for inventory monitoring. This outdated approach presents several drawbacks. Firstly, searching for historical transaction data becomes cumbersome and time-consuming, as the owner must sift through pages of the notebook to locate specific information. Secondly, preparing financial reports is complicated, requiring considerable effort and manual calculations, which increases the likelihood of errors. Thirdly, the lack of data backups poses a high risk of data loss if the notebook is misplaced, damaged, or destroyed. Lastly, the manual recording process is inherently prone to human error, takes longer to execute, and ultimately reduces the efficiency of the store's daily operations. These inefficiencies can hinder the store's growth and its ability to compete effectively in the market.

Recognizing these challenges, a web-based transaction recording and inventory management application was developed to address the store's needs. This application is specifically designed to eliminate the limitations of manual recording methods by providing an automated, accessible, and efficient solution. As a web-based system, the application allows users to access it from anywhere with an internet connection, ensuring greater flexibility and usability. The primary objectives of the application are to minimize errors commonly associated with manual data entry, simplify inventory management processes, and significantly improve operational efficiency. By incorporating features to generate detailed and comprehensive reports, the application enables the store owner to make informed and data-driven business decisions. This functionality not only supports day-to-day operations but also provides strategic insights that can help optimize inventory levels, monitor sales trends, and plan future business growth effectively [3].

The scope of this transaction recording and inventory management application includes the development of various essential features, such as a dashboard, user management, supplier management, product management, product specifications management, purchase and sales transaction recording, and reporting functionalities. The application is designed to accommodate two types of users: owners and admins. Owners have complete access to the system, enabling them to view detailed reports, manage all data, and monitor the system comprehensively. Admins, on the other hand, are employees with restricted access, limited to specific functions relevant to their role.

2. Literature Review

2.1 State of the Art and Novelty of this Research

In designing the transaction recording and inventory management application for Unipro Store, a thorough review of previous studies was conducted to identify existing gaps and uncover opportunities for novelty. Prior research has demonstrated the importance of transaction recording systems in improving operational efficiency and minimizing errors. However, most studies only cover specific aspects and do not address the entire operational workflow of their respective research objects. For instance, some systems lack features for purchase transaction recording, supplier management, or comprehensive reporting. These gaps underscore the need for a more integrated and complete solution, which this research aims to fulfill through a tailored application for Unipro Store.

The novelty of this research was identified through a combination of methods aimed at addressing the limitations of the current manual system. Observations of Unipro Store's existing processes revealed significant challenges, including the time-consuming nature of manual data recording, difficulty in retrieving historical transaction data, and the lack of integration between different operational workflows. Furthermore, interviews with the store owner highlighted the need for features like automated reporting and real-time stock

tracking. A comparative analysis was also conducted by benchmarking against similar studies and existing systems. This analysis revealed that while previous studies have implemented various features such as transaction recording and inventory management, they have not fully addressed certain key aspects. Table 1 below outlines the weaknesses of previous studies and highlights the novelty introduced in this research.

Table 1. State of The Art

Previous Study	Weaknesses	Novelty in this Research
Ikhsan et al. [4]: Financial Transaction and Inventory Recording Application for MSL Purbalingga Distributors	The study is limited to the Android platform, making it less accessible on other devices such as desktops or web browsers. Additionally, it lacks a login feature, as the application supports only a single admin user with equal authority. There is also limited flexibility in user management and no support for multi-user roles.	The research introduces a web-based system with multi-user roles, including owners and admins, equipped with secure login features. It allows owners to manage admin accounts, assign access based on roles, and provides options to export reports in various file formats, which were absent in prior studies.
Kinselton and Tony [5]: Design of a Web-Based Sales Recording Application for My Sport Indonesia	The study focuses on recording sales transactions and does not include features for purchase transaction workflows. It lacks supplier management, leaving no functionality to track or organize supplier-related activities systematically. Although it provides features to manage incoming stock, these are solely tied to sales transactions.	The research introduces an improvement by integrating purchase transaction recording with comprehensive stock management, ensuring that all incoming stock is documented alongside its corresponding purchase transactions, rather than merely tracking stock adjustments independently.
Djoni and Tony [6]: Web-Based Inventory Management System for Sumber Abadi Building Supplies Store	This study focuses only on basic CRUD operations for inventory management and does not extend to broader business processes like tracking sales and purchase transactions. The absence of a dashboard limits the ability to monitor operations effectively.	The research expands beyond inventory management to include comprehensive tracking and documentation of all sales and purchase transactions, ensuring complete operational coverage. Additionally, the system provides a concise dashboard for better visibility into key business activities and supports advanced reporting with export features.

In addition to insights gained from previous studies, it is essential to have a comprehensive understanding of fundamental theories relevant to system development. These theories provide the foundational knowledge needed to ensure that the system is designed and implemented effectively, aligning with both technical standards and user requirements. Below are the key theoretical concepts that underpin the development of the transaction recording and inventory management application for Unipro Store.

2.2 Some important terms in this research

Furthermore, Some important terms that also play a major role in the completion of this research include: Application, Web, Dashboard, Role-Based Access Control (RBAC), Unified Modeling Language (UML), ASP.NET, Microsoft SQL Server, User Acceptance Testing (UAT), Black box testing, and System Usability Scale (SUS) and the following terms are explained in detail.

Application is a software program specifically designed to address particular user needs, encompassing functions such as data processing, analysis, and information management [7]. Application can be deployed across various platforms, including desktop, web, and mobile systems. They play a crucial role in enhancing efficiency, automating processes, and boosting productivity. Additionally, applications facilitate improved data management and offer analytical tools that support more informed and effective decision-making [8].

Web is a dynamic and ever-evolving platform that not only enables the delivery of content but also fosters user interaction, innovation, and community building.

By connecting users and granting access to an extensive array of information and resources, the web functions as a tool that enhances collective intellectual capabilities and supports the resolution of more complex problems [9]. Its structure comprises three primary components: the URL (Uniform Resource Locator), which identifies resources on the web; HTTP (Hypertext Transfer Protocol), which facilitates communication; and HTML (HyperText Markup Language), which is used to create web pages [10].

The dashboard is a visual interface designed to display essential data in an organized and concise manner, enabling users to monitor, analyze, and make informed decisions based on the presented information. In the context of a product sales business, a dashboard typically includes key components such as charts, tables, and key performance indicators (KPIs) to provide an overview of operational performance. Common metrics displayed may include inventory levels, sales trends, order statuses, and financial summaries [11].

Role-Based Access Control (RBAC) is a security paradigm that restricts system access to authorized users based on their roles within an organization. In RBAC, permissions are associated with roles, and users are assigned to these roles, thereby acquiring the corresponding permissions. This model simplifies the management of user permissions by grouping them into roles that align with organizational responsibilities [12].

Unified Modeling Language (UML) is a standardized modeling language utilized to describe, design, and document object-oriented software systems. While UML provides a comprehensive notation for representing system designs, it does not prescribe specific steps or processes for achieving those designs. Its primary role is to facilitate effective communication in system modeling, enabling various stakeholders in the development process to comprehend the design without adhering to a uniform methodology [13].

ASP.NET is a free, open-source framework developed by Microsoft, designed for building advanced websites and web applications using HTML, CSS, and JavaScript. It offers three primary frameworks for web application development: Web Forms, ASP.NET MVC, and ASP.NET Web Pages. Each framework adopts a distinct development approach, allowing developers to choose based on their expertise, experience, the application's requirements, and their preferred development style [14].

Microsoft SQL Server, developed by Microsoft, is a leading Relational Database Management System (RDBMS) that provides a range of powerful components, including the Database Engine, Analysis Services, Reporting Services, and Integration Services. These components are designed to manage structured, semi-structured, and unstructured data efficiently. With its robust integration with the Windows operating system, SQL Server is a preferred choice for many developers and business users seeking reliable and scalable database solutions [15].

User Acceptance Testing (UAT) is a process where end users evaluate a system to determine if it is ready for deployment in a real-world work environment. This testing phase is critical for securing user approval before full implementation. UAT ensures that the system meets business requirements and functions as intended under actual conditions, helping to verify its readiness for operational use [16].

Black box testing is a software testing methodology that focuses on evaluating the functionality of an application without requiring knowledge of its internal implementation or code structure. The goal is to verify that the application

produces correct outputs based on given inputs. This method ensures that the application's user interface and external functionalities work correctly, providing a reliable experience for end users [17].

System Usability Scale (SUS) is a widely used questionnaire for assessing the usability of products, including hardware, applications, and websites. The SUS consists of 10 statements, where respondents rate their level of agreement on a 5-point scale: 1 (strongly disagree), 2 (disagree), 3 (neutral), 4 (agree), and 5 (strongly agree). Odd-numbered statements convey positive sentiments, while even-numbered statements express negative ones. This structure provides a balanced evaluation of a product's usability [18].

3. Method

The research method employed in developing the web-based application for transaction recording and inventory management at Unipro Store is the Software Development Life Cycle (SDLC) waterfall model. The sequential stages of the SDLC waterfall model are illustrated in Figure 1.

The following is a detailed explanation of each stage of the SDLC waterfall model applied in system development [20]:

3.1 Requirement Analysis

This initial stage involves gathering and documenting all requirements submitted by Unipro Store users. The author also conducts interviews during this phase. The primary goal is to obtain a clear and detailed understanding of the system's requirements, which will serve as the foundation for the subsequent development stages.

3.2 System Design

After identifying the system requirements, this stage focuses on designing the system's architecture and layout. The system design process utilizes the Unified Modeling Language (UML) method, which includes creating Use Case Diagrams, Activity Diagrams, and Sequence Diagrams. Additionally, this phase involves database design and user interface (UI) design to ensure a coherent and user-friendly system structure.

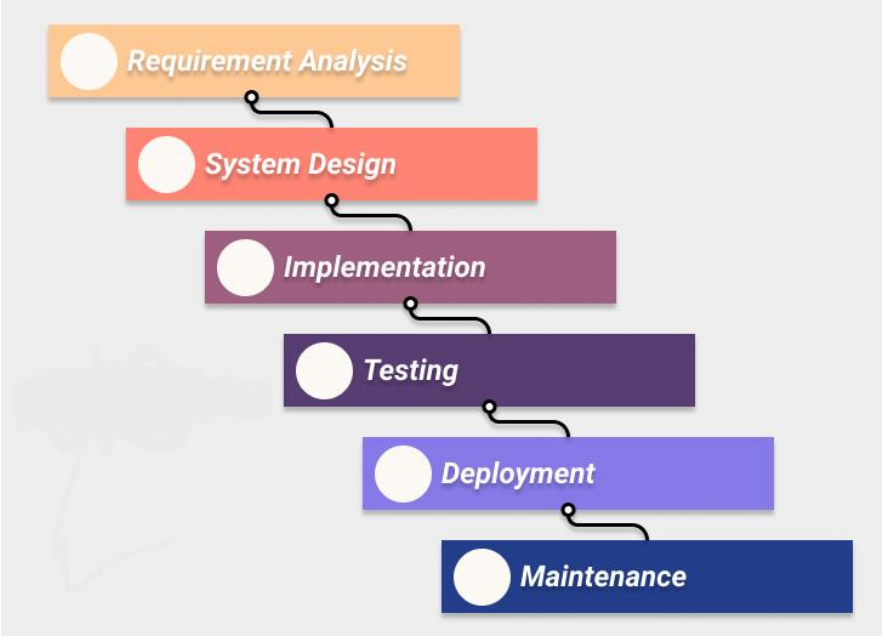

Figure 1. SDLC Waterfall Model [19]

3.3 Implementation

At this stage, the author begins coding the system based on the predetermined design. Each system component is developed following the previously agreed specifications. This phase translates the designs into a functional application.

The implementation phase involves the development of various components of the application, which include user interface design, database integration, and business logic implementation. The application was built using the ASP.NET framework, leveraging C# for server-side logic, and HTML, CSS, and JavaScript for the front-end interface. The database was implemented using Microsoft SQL Server, which integrates seamlessly with the application's back-end. The following are examples of key processes implemented during this phase:

3.3.1 User Authentication and Role Management

The system incorporates a login feature with role-based access control (RBAC). Users are classified into two roles: owner and admin. The following pseudocode illustrates the user authentication process:

```
BEGIN
  Prompt user to enter username and password
  Retrieve user details from User table
  IF username or password is invalid THEN
    Display error message and prompt user to re-enter
    RETURN to login page
  END IF
  Check user role:
    IF role = "Owner" THEN
      Redirect to owner dashboard
    ELSE IF role = "Admin" THEN
      Redirect to admin dashboard
    END IF
END
```

3.3.2 Transaction Recording

The system allows for the recording of sales and purchase transactions. Each transaction is validated to ensure the completeness of the data and the availability of stock for sales. Below is the pseudocode for adding a sales transaction:

```
BEGIN
  Prompt user to input transaction details (sales media, products
purchased, discount)
  Validate transaction details:
    IF any required field is missing THEN
      Display error message
      RETURN to input form
    END IF
  Initialize total price to 0
  Calculate total price:
    FOR each product in the transaction
      Retrieve product price
      Multiply price by quantity
      Add to total price
    END FOR
  Apply discount:
    IF discount > 0 THEN
      Calculate discounted amount = total price * (discount / 100)
      Subtract discounted amount from total price
    END IF
```

```
  Save transaction to database:
    INSERT transaction details into SalesTransaction table
      (transaction ID, date, sales media, total product, total price,
discount, discounted price, user)
     RETURN generated transaction ID
    FOR each product in the transaction
      INSERT product details into SalesTransactionDetail table
        (transaction ID, product ID, quantity, discount, subtotal)
      Update stock in Product table
    END FOR
  Display success message with transaction ID
END
```

3.3.3 Inventory Management

The system includes a stock notification feature to alert users when product stock levels fall below a defined threshold. The logic for this feature is implemented as follows:

```
BEGIN
  Initialize low-stock notification list
  Initialize out-of-stock notification list
  FOR each product in Product table
    IF stock quantity = 0 THEN
      Add product to out-of-stock notification list
    ELSE IF stock quantity < minimum threshold THEN
      Add product to low-stock notification list
    END IF
  END FOR
  Display out-of-stock notification list to user
  Display low-stock notification list to user
END
```

3.3.4 Reporting

The application generates reports for both sales and purchase transactions. These reports can be downloaded in multiple formats (PDF and Excel). The logic for generating a purchase report is as follows:

```
BEGIN
  Prompt user to select report parameters (e.g., date range,
subcategory, brand, phone, supplier)
  Validate selected parameters:
    IF any required parameter is invalid THEN
      Display error message
      RETURN to report parameter selection form
    END IF
  Retrieve filtered transaction data from SalesTransaction and
SalesItem tables
    Apply filters:
      IF date range is specified THEN
        Filter data by date range
      END IF
      IF subcategory is specified THEN
        Filter data by subcategory
      END IF
      IF brand is specified THEN
        Filter data by brand
      END IF
      IF phone is specified THEN
```

```
      Filter data by phone model
    END IF
    IF supplier is specified THEN
      Filter data by supplier
    END IF
  Format filtered data according to selected output format (PDF or
Excel)
  Generate report and save file
  Provide download link to user
  Display success message
END
```

3.3.5 Testing

This stage ensures that all system functionalities operate according to the specified requirements and are free of errors. Testing will be conducted using User Acceptance Testing (UAT) with a black box testing approach, involving the owner and employees of Unipro Store. Furthermore, the System Usability Scale (SUS) questionnaire will be distributed during this phase to evaluate the usability of the application.

3.3.6 Deployment

Once the system has passed all testing phases and is deemed qualified, it will be deployed to the production environment for use by end users. This stage ensures the application is fully operational and ready for real-world use.

3.3.7 Maintenance

The maintenance phase involves monitoring and regular upkeep of the system to ensure it continues to perform optimally. If bugs are identified or new requirements arise, the author will implement necessary fixes and updates to support system development.

4. Result and Discussion

This research has developed a web-based application for transaction recording and inventory management that can be implemented at the Unipro Store. The following sections outline the results of the requirement analysis, system design, implementation, and testing of the developed application.

4.1 Requirement Analysis Results

During the requirement analysis stage, the author conducted a comprehensive information-gathering process to identify the needs and challenges faced by Unipro Store. Data collection involved direct interviews with the store owner, who possesses the most in-depth knowledge of the business's daily operations. These interviews were conducted online via the Google Meet platform. The discussions focused on understanding the flow of business processes, the platforms used for selling products to consumers, and the current manual methods of recording transactions and managing product stock, which rely on physical notebooks.

From the interview findings, the author identified the key requirements for the application development. The application is designed to incorporate two distinct user roles: an owner role for store owners with full authority and access to advanced features, and an admin role for store employees with limited access to specific functionalities. Features such as user management and report generation are exclusively accessible to the owner. Other features, including supplier management, category management, product management, sales transaction management, and purchase transaction management, are accessible to both user roles in the web-based application.

4.2 System Design Results

During the system design stage, the author employed the Unified Modeling Language (UML) to create visual representations that effectively define and clarify the system's requirements and functionality. Various UML diagrams were developed to support this process, including use case diagrams, activity diagrams, sequence diagrams, and class diagrams. In parallel with the UML diagrams, a comprehensive database design was developed. This process was carried out in three distinct phases, namely: conceptual database design, logical database design, and physical database design.

4.2.1 Use Case Diagram

A use case diagram is utilized to capture the functionality and requirements of the system, showcasing the interactions between the application and its users. The application developed in this research involves two primary actors: the owner and the admin. The main distinction between these roles lies in their access to specific features. The owner has exclusive access to the user management menu, which allows them to manage user account data, and the sales and purchase report menu, which enables viewing and downloading reports. Apart from these exclusive features, all other functionalities are accessible to both actors. Both the owner and admin can log in and out of the application. After logging in, they can access various features, including the dashboard menu, which provides a visual summary of essential information, the supplier menu for managing supplier data, the product menu for handling product data, the product specification menu for managing detailed product specifications, the purchase menu for recording purchase transactions, including all product details, and the sales menu for documenting sales transactions from various platforms, complete with product details. The use case diagram for this web-based application, designed for transaction recording and inventory management at Unipro Store, is depicted in Figure 2.

4.2.2 Activity Diagram

An activity diagram is used to illustrate the flow or processes within the system. In the design of the *Unipro* Store transaction recording and inventory management application, a total of 67 activity diagrams have been created to represent various operational processes. For instance, the activity diagram depicting the process of adding sales transactions is presented in Figure 3, while the activity diagram illustrating the sales report download process is shown in Figure 4.

4.2.3 Sequence Diagram

The sequence diagram is used to model the behavior of a system from an analytical perspective, focusing specifically on the actor's point of view. In designing the Unipro Store transaction recording and inventory management application, a total of 67 sequence diagrams have been developed to represent various operational processes. For example, the sequence diagram illustrating the process of adding purchase transactions is presented in Figure 5, while the sequence diagram for the purchase report download process is shown in Figure 6.

4.2.4 Class Diagram

The class diagram describes the system's elements and the relationships between them, which remain constant over time. A total of 18 classes have been created in the class diagram. The class diagram designed for the web-based application for transaction recording and inventory management at Unipro Store is presented in Figure 7.

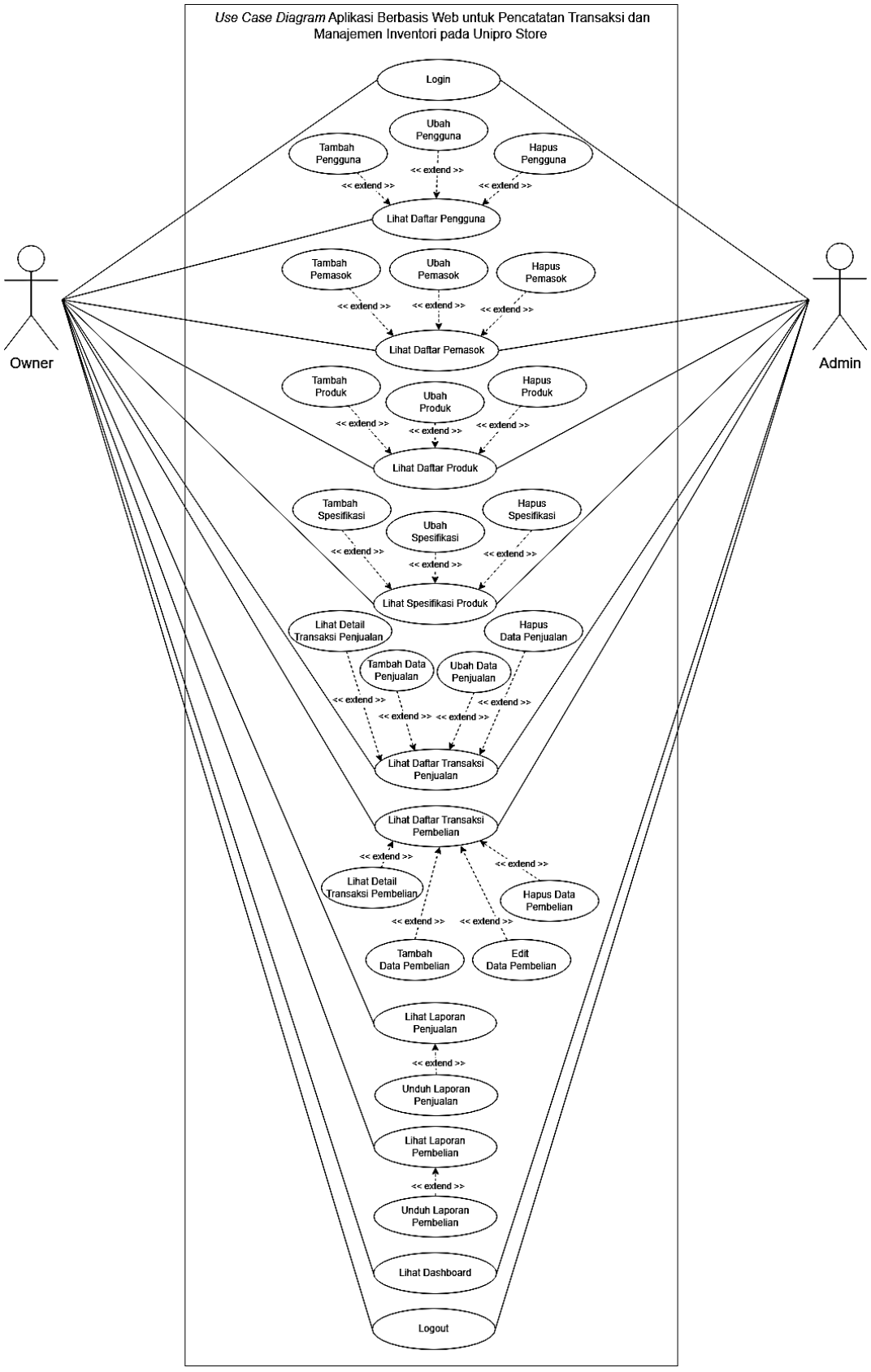

Figure 1. Use Case Diagram

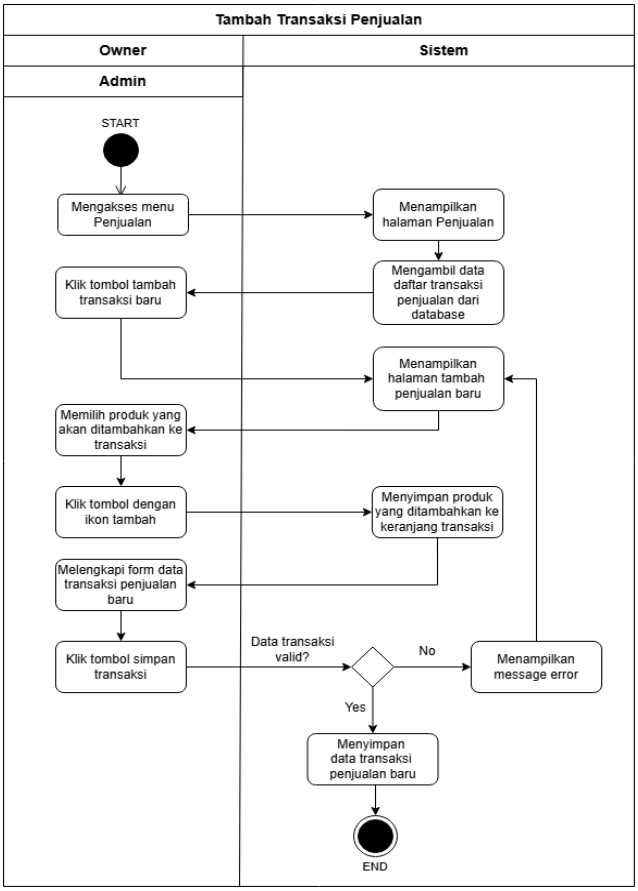

Figure 2. Activity Diagram Add Sales Transaction

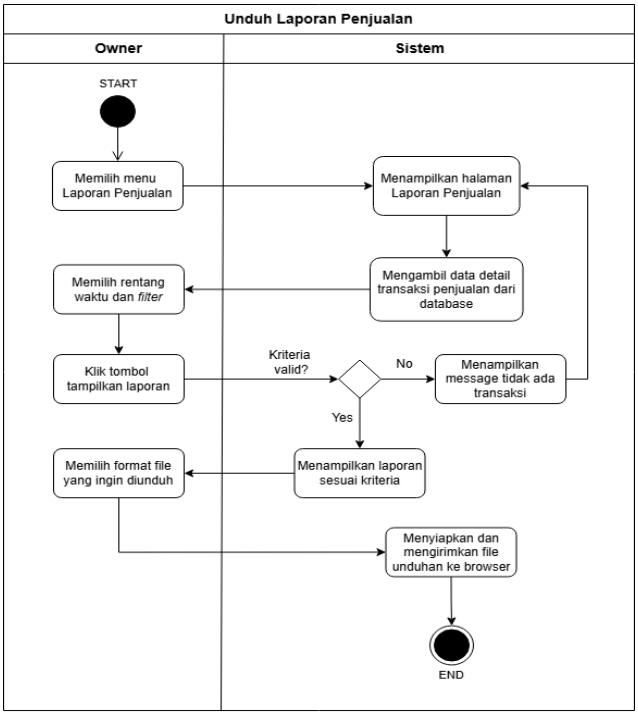

Figure 3. Activity Diagram Download Sales Report

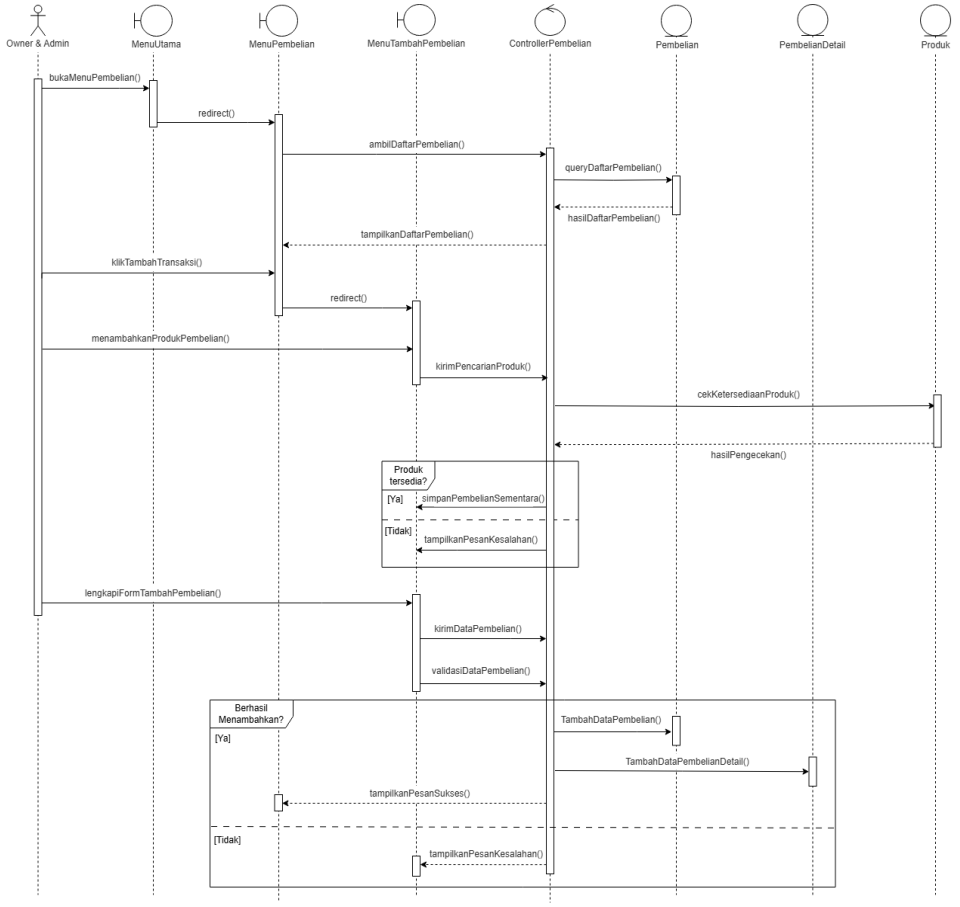

Figure 4. Sequence Diagram Add Purchase Transaction

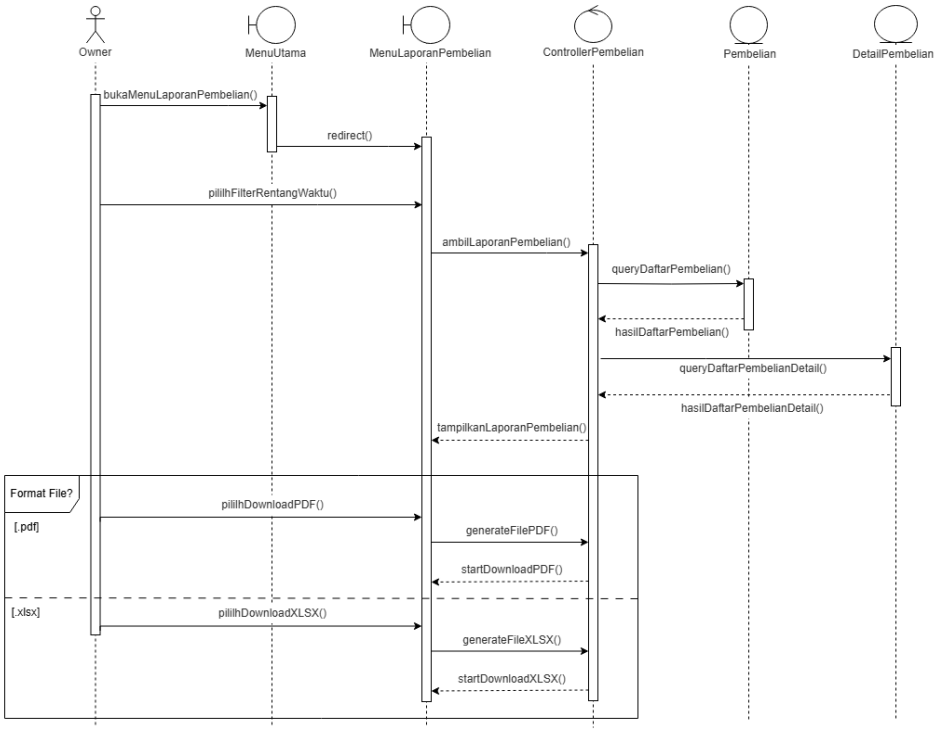

Figure 5. Sequence Diagram Download Purchase Report

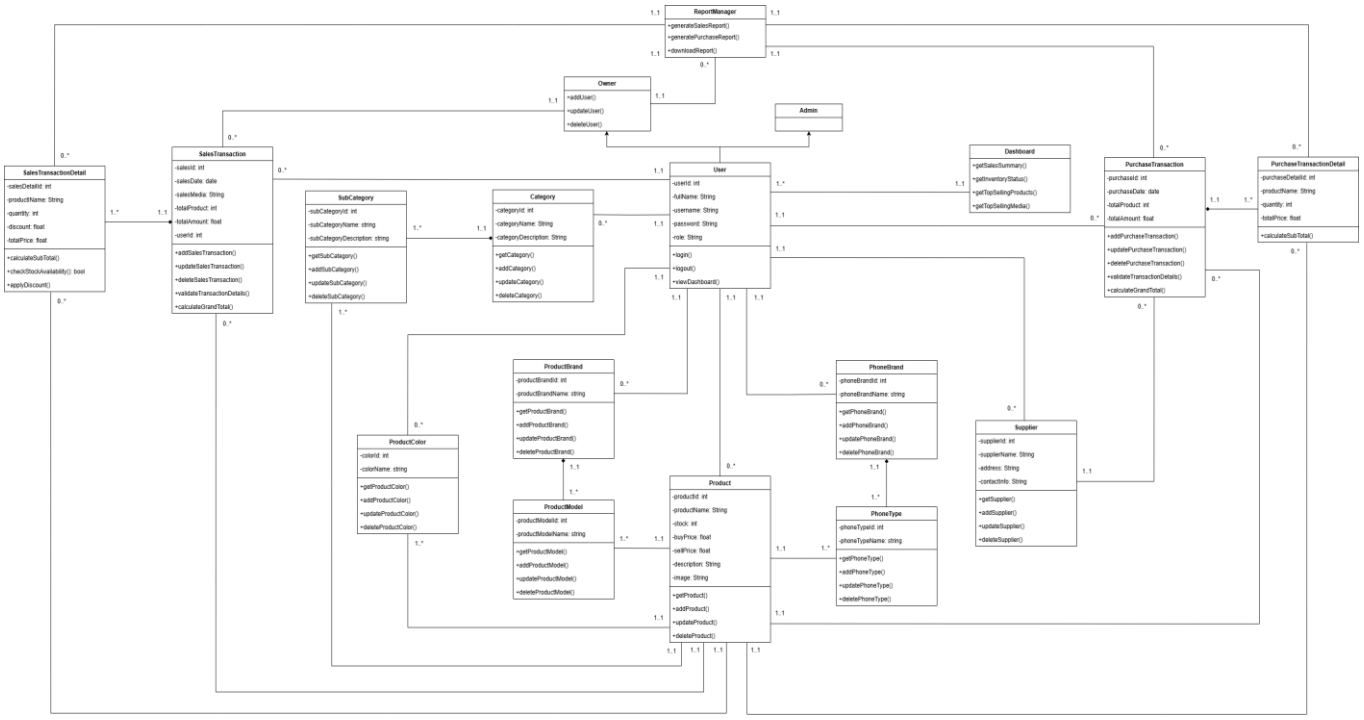

Figure 6. Class Diagram

4.2.5 Database Design

The database design of the web-based application for transaction recording and inventory management at Unipro Store is conducted in three phases: conceptual database design, logical database design, and physical database design. The first phase involves creating the conceptual database design, which focuses on identifying the required entities and their relationships. This is followed by the logical database design, which specifies the attributes of each entity along with their keys. The final phase is the physical database design, where the target database management system (DBMS) is determined, along with the type and length of each attribute [21]. The conceptual database design for the web-based application is presented in Figure 8, while the logical database design is depicted in Figure 9.

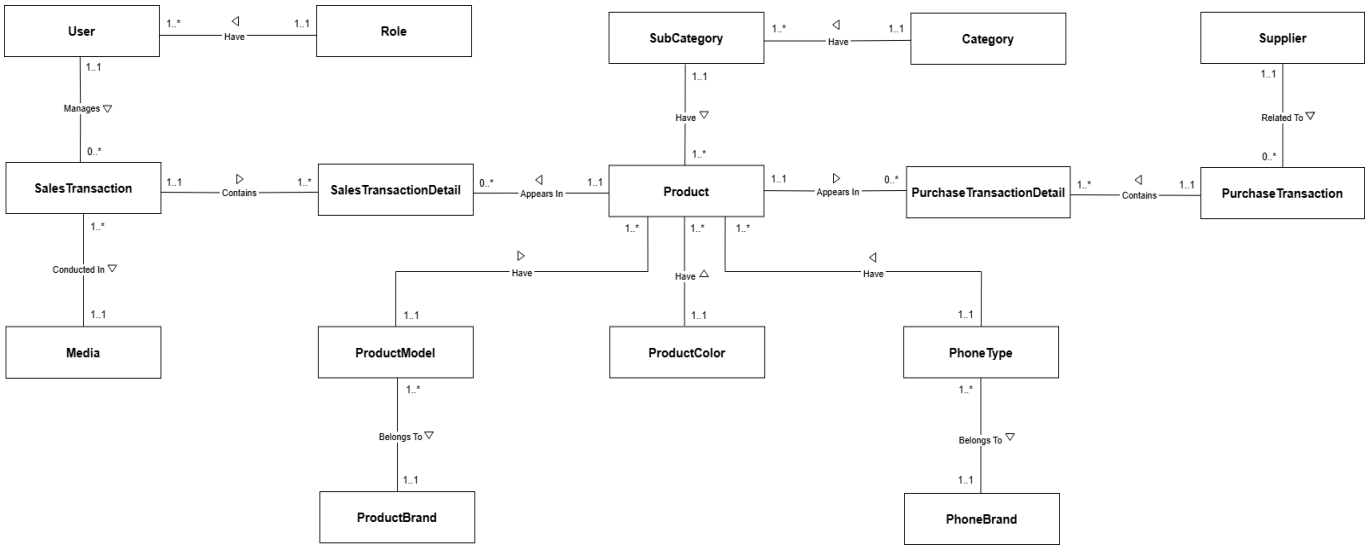

Figure 7. Conceptual Database Design

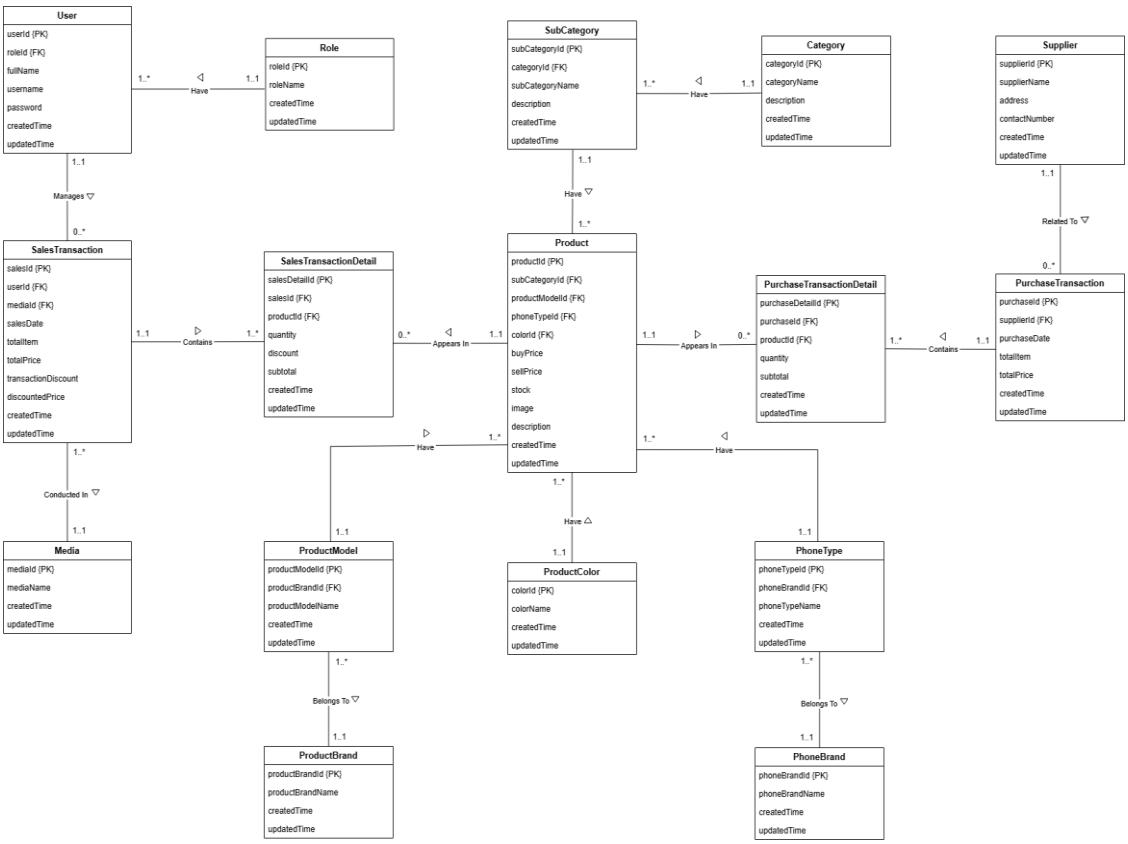

Figure 8. Logical Database Design

4.3 Implementation Results

The following are the results of the implementation of the previous system design into a web-based application.

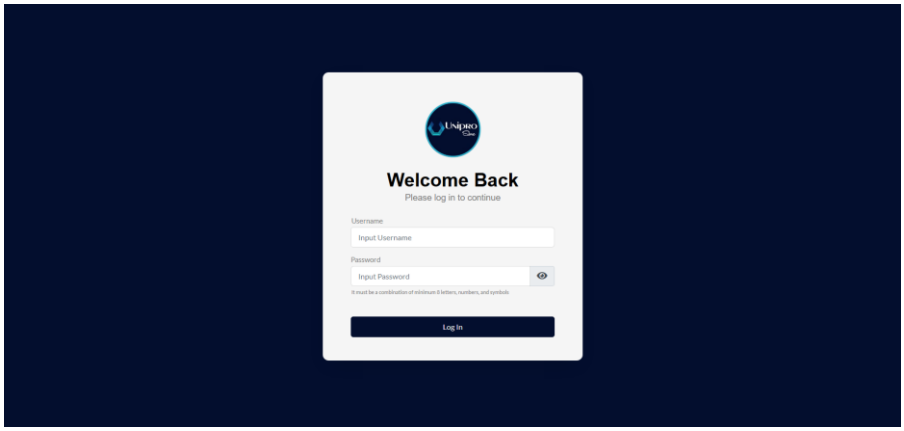

Figure 9. Login Page

Figure 10 shows the login page which is used to distinguish access rights between users with owner and admin level. On this login page, the user is asked to enter a username and password. If the login is successful, the dashboard page will appear. However, if the username or password does not match, the user fails to log in and will remain on the login page.

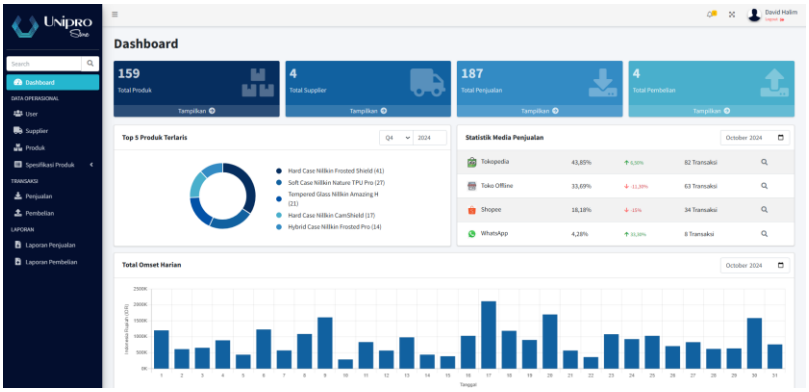

Figure 10. Dashboard Page

Figure 11 shows the dashboard page after successful login. This dashboard is useful for presenting data summaries and analysis results from sales at the Unipro Store which can help decision making as well. On the dashboard page, there are several widgets including, total products, total suppliers, total sales and purchase transactions in the month, top 5 best-selling products, sales media statistics, and total daily turnover.

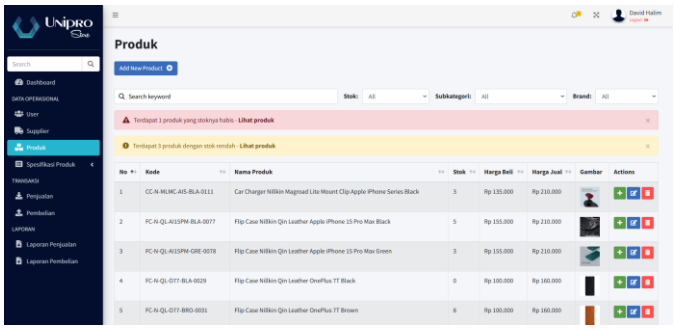

Figure 11. Product Page

Figure 12 shows the product page. On the product page, a list of all registered products is displayed and users can manage products, such as create, read, update, and delete product data.

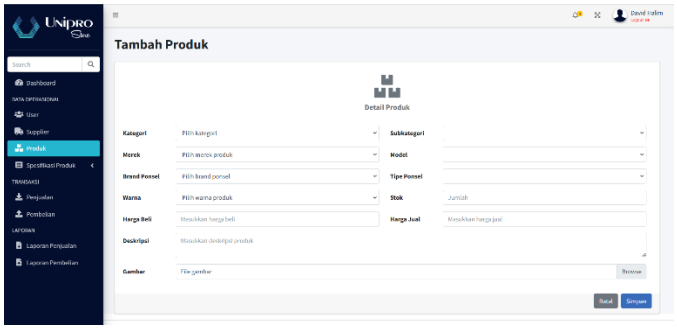

Figure 12. Add Product Page

Figure 13 shows the add product page. This page allows users to enter new product details, including fields such as category, subcategory, product brand, product model, phone brand, phone type, color, stock, buy price, sell price, and description. There is also an option to upload an image to visually represent the product. Once the required information is filled out, users can save the new product data.

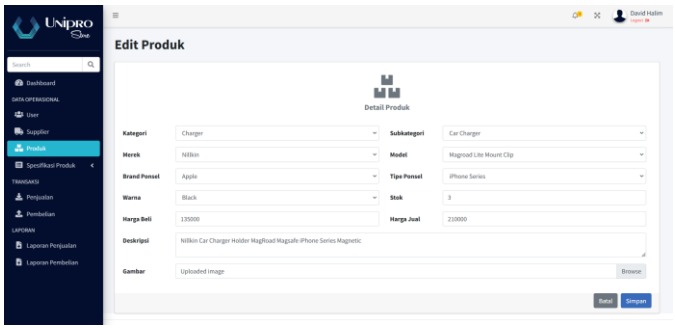

Figure 13. Edit Product Page

Figure 14 shows the edited product page. This page enables users to modify existing product information by updating various fields, such as category, subcategory, product brand, product model, phone brand, phone type, color, stock, buy price, sell price, and description. Similar to the add product page, there is an option to replace the product image if needed. Once edits are made, users can save the changes.

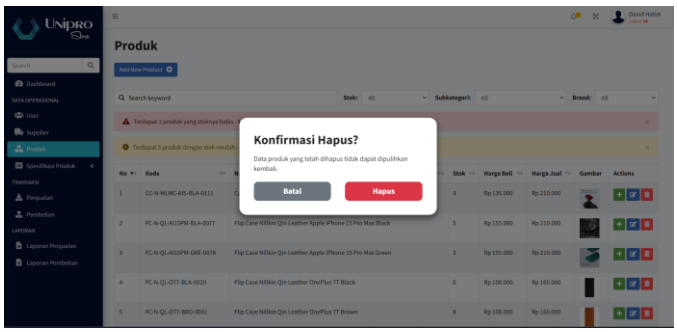

Figure 14. Delete Product Confirmation Message

Figure 15 shows the deleted product confirmation message. When a user attempts to delete a product, a confirmation dialog appears, asking the user to confirm or cancel the deletion. This dialog helps prevent accidental deletions by requiring explicit confirmation from the user.

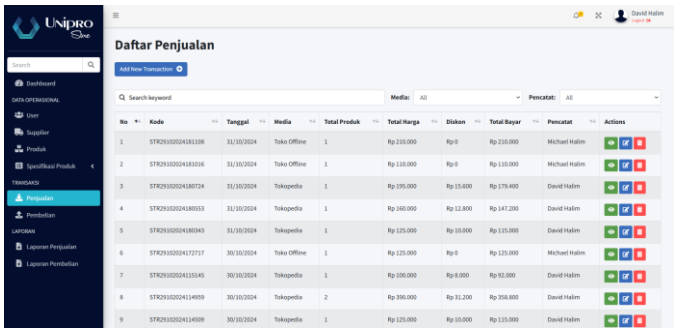

Figure 15. Sales Transaction Page

Figure 16 shows the sales transaction page. This page displays a list of all recorded sales transactions, providing users with an overview of transaction details such as transaction code, date, media, total product, total price, discount, discounted price, and name of the person who recorded the transaction. Each transaction entry includes options for viewing, editing, or deleting which are accessible through action buttons on the right side of each row.

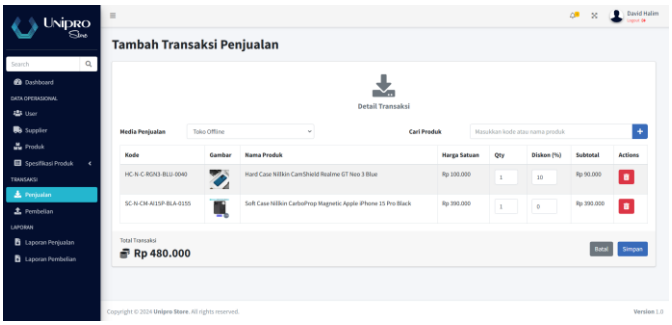

Figure 16. Add New Sales Transaction Page

Figure 17 shows the add new sales transaction page. This page enables users to create a new sales transaction by inputting relevant details. Users can add multiple products to a single transaction, with a subtotal automatically calculated for each product. The page also provides fields for entering quantity and discount amounts.

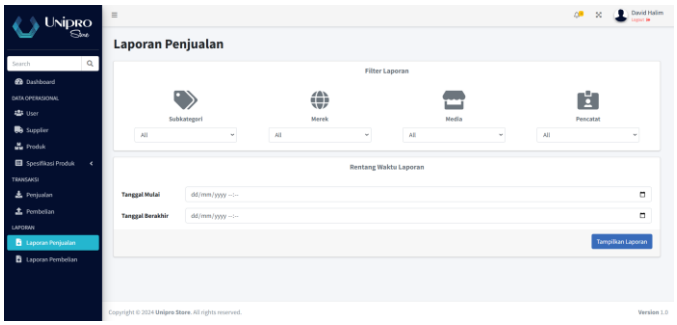

Figure 17. Sales Report Page

Figure 18 shows the sales report page, where users can generate and view reports on sales transactions based on specified criteria such as subcategory, product brand, media, and the person who recorded the transaction. Before the report results. Before the report results are displayed, users must select the report date range and press the button to display the report.

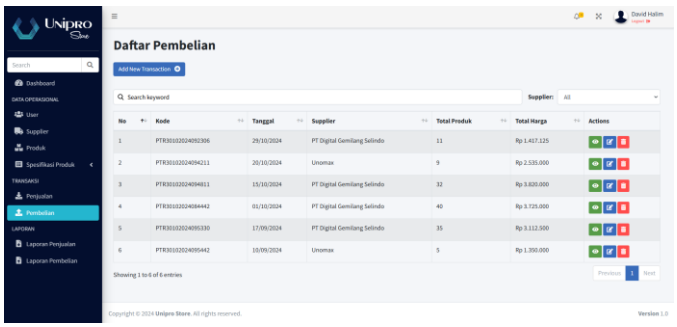

Figure 18. Purchase Transaction Page

Figure 19 shows the purchase transaction page. This page displays a list of all recorded purchase transactions, providing users with an overview of transaction details such as transaction code, date, supplier, total product, and total price. Each transaction entry includes options for viewing, editing, or deleting which are accessible through action buttons on the right side of each row.

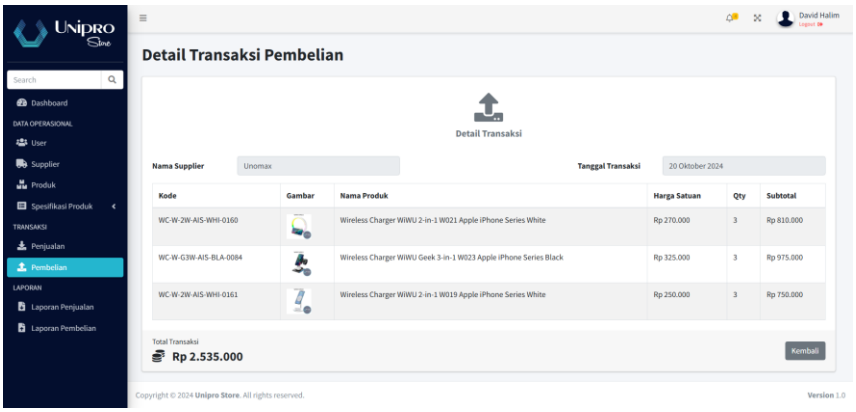

Figure 19. View Purchase Transaction Detail Page

Figure 20 shows the view purchase transaction detail page. This page provides a detailed view of a specific purchase transaction, including information such as the supplier's name and transaction date. The page lists each product purchased, along with relevant details like the product name, unit price, quantity, and subtotal for each product. The total amount at the bottom summarizes the overall cost of the transaction.

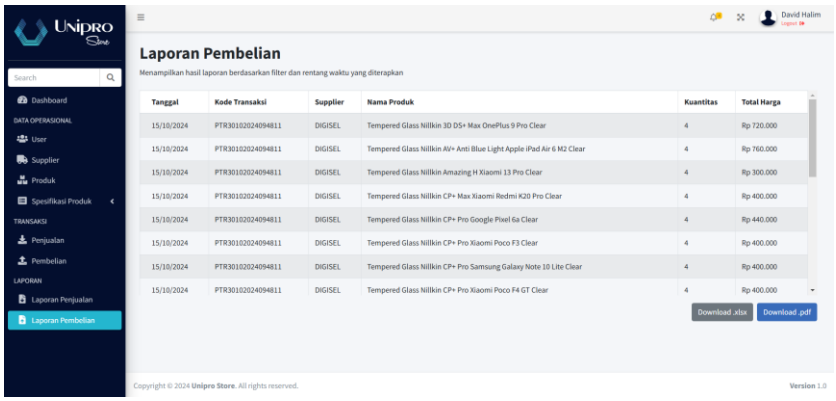

Figure 20. Purchase Report Result Page

Figure 21 shows the purchase report result page. This page provides a detailed list of purchase transactions based on previously selected criteria and date ranges. Each transaction includes information such as transaction date, transaction code, supplier, product name, quantity, and the total price. Users also can download this report in PDF or Excel format.

4.4 Testing Result

Upon completing the development process, the application undergoes User Acceptance Testing (UAT), which involves evaluation by end users. This testing is conducted using the black box testing approach to ensure the application meets the specified requirements. The results of the application testing using the black box testing approach are presented in Table 2.

Table 1. Black Box Testing

No	Functional	Test Scenario	Expected Output	Status
1	Login	Fill in the login form with valid and invalid credentials.	Successfully logs in with valid credentials. Then, displays an error message "Invalid username or password" for invalid credentials.	Pass
2	Dashboard	Open the dashboard page, and apply filters for month, quarter, and year in a widget.	Displays dashboard with widgets based on current data.	Pass
3	User Management	CRUD operations on user accounts: add, update, delete, and search users.	Adds, updates, and deletes users successfully; displays error for unmet conditions; searches display relevant user records.	Pass
4	Supplier Management	CRUD operations on suppliers: add, update, delete, and search suppliers.	Adds, updates, and deletes suppliers successfully; displays error for unmet conditions; searches display relevant supplier records.	Pass
5	Category Management	CRUD operations on product categories: add, update, delete, and search categories.	Adds, updates, and deletes categories successfully; displays error for unmet conditions; searches display relevant category records.	Pass
6	Subcategory Management	CRUD operations on product subcategories: add, update, delete, and filter by main category.	Adds, updates, and deletes subcategories successfully; displays error for unmet conditions; filter displays subcategories based on selected main category.	Pass
7	Product Brand Management	CRUD operations on product brands: add, update, delete, and search brands.	Adds, updates, and deletes product brands successfully; displays error for unmet conditions; searches display relevant brand records.	Pass
8	Product Model Management	CRUD operations on product models: add, update, delete, and filter by brand.	Adds, updates, and deletes product models successfully; displays error for unmet conditions; filters display models according to the selected brand.	Pass
9	Phone Brand Management	CRUD operations on phone brands: add, update, delete, and search phone brands.	Adds, updates, and deletes phone brands successfully; displays error for unmet conditions; searches display relevant phone brand records.	Pass

No	Functional	Test Scenario	Expected Output	Status
10	Phone Type Management	CRUD operations on phone types: add, update, delete, and filter by phone brand.	Adds, updates, and deletes phone types successfully; displays error for unmet conditions; filters display types according to the selected phone brand.	Pass
11	Product Color Management	CRUD operations on product colors: add, update, delete, and search colors.	Adds, updates, and deletes product colors successfully; displays error for unmet conditions; searches display relevant color records.	Pass
12	Product Management	CRUD operations on products, including variations and filtering by stock and brand.	Adds, updates, and deletes products with variations; filters display products based on stock and brand filters.	Pass
13	Sales Transactions	CRUD operations on sales transactions, including viewing details and filtering.	Adds, updates, and deletes sales transactions; displays transaction details; filters transactions by media and user.	Pass
14	Purchase Transactions	CRUD operations on purchase transactions, including viewing details and filtering.	Adds, updates, and deletes purchase transactions; displays transaction details; filters transactions by supplier.	Pass
15	Reports	Generate and filter sales and purchase reports, and download them in .xlsx or .pdf format.	Displays reports based on filters; successfully downloads in both formats.	Pass
16	Notifications	Click the bell icon to display notifications for low/out-of-stock products.	Displays relevant stock notifications when the bell icon is clicked.	Pass
17	Logout	Click the "Logout" button and confirm.	Ends the session and redirects to the login page.	Pass

In addition, a System Usability Scale (SUS) questionnaire was distributed during the testing phase to evaluate the usability of the web-based application for transaction recording and inventory management at Unipro Store. The questionnaire was shared with respondents via Google Forms, allowing them to provide their feedback conveniently. The SUS scores were calculated by adjusting respondent ratings: for odd-numbered statements, 1 was subtracted from the rating, and for even-numbered statements, the rating was subtracted from 5. The adjusted scores were then summed to obtain a raw score, which was multiplied by 2.5 to produce a standardized SUS score out of 100. The detailed results for each statement are displayed in Figure 22, which provides a breakdown of user responses.

Figure 21. SUS Questionnaire Results for Each Statement

Table 3. SUS Questionnaire Final Results

Respondents	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total	Score
1	3	3	4	3	4	4	2	4	3	2	32	80
2	4	3	3	3	4	3	4	3	3	2	32	80
3	3	4	4	4	3	3	3	3	4	3	34	85
4	4	2	2	3	4	3	3	3	3	2	29	72,5
5	4	3	3	4	3	4	3	4	4	3	35	87,5
6	3	4	4	3	4	3	3	4	4	2	34	85
Final Score												**81,67**

As shown in Table 3, the SUS scores from six respondents ranged from 72.5 to 87.5, with an average score of 81.67. Respondents who scored higher, such as Respondent 5 with a score of 87.5, indicated a highly positive perception of the application's usability, while Respondent 4, with a score of 72.5, identified potential areas for improvement. The scores reflect the ease of use, functionality, and overall user experience provided by the application. The average SUS score of 81.67 places the application in the "A" grade category, signifying excellent usability. These results confirm that the application is user-friendly, meets expectations, and is ready to be deployed in an operational setting.

5. Conclusion

Based on the results of the web-based application for transaction recording and inventory management developed for Unipro Store, the following conclusions can be made: The application greatly improves efficiency in recording transactions and accessing historical data, while simplifying inventory management through its stock notification feature. It enables users to directly download sales and purchase reports in two different formats and offers valuable analytical insights through its dashboard. Additionally, the application successfully passed black box testing with all scenarios executed by both user types completing without issues. The System Usability Scale (SUS) questionnaire further assessed the application with a score of 81.67, placing it in the A grade of usability evaluation.

6. Suggestion

For future research, enhancements could include integrating the application with Application Programming Interfaces (APIs) of prominent e-commerce platforms such as Tokopedia and Shopee, enabling automatic transaction recording directly into the application to streamline data entry and minimize errors. Synchronizing product stock across these platforms would ensure accurate inventory tracking and operational consistency across all sales channels. Additionally, the implementation of a barcode scanning feature for product searches could be explored, allowing users to retrieve product information quickly and accurately during sales or inventory management processes. This feature would reduce manual input errors, save time, and further enhance the system's overall efficiency and user experience.

Acknowledgments: The researchers would like to extend their deepest gratitude to Unipro Store for the invaluable support and cooperation throughout this project. Unipro Store's willingness to provide insights into operational processes and share data was essential in understanding the practical requirements for the development of this application. Special thanks to the owner and staff for their patience, responsiveness, and encouragement, which greatly contributed to the success of this study. This project would not have been possible without the involvement and dedication of all parties.

Author contributions: The authors were responsible for building Conceptualization, Methodology, analysis, investigation, data curation, writing—original draft preparation, writing—review and editing, visualization, supervision of project administration, funding acquisition, and have read and agreed to the published version of the manuscript.

Funding: The study was conducted without any financial support from external sources.

Availability of data and Materials: All data are available from the authors.

Conflicts of Interest: The authors declare no conflict of interest.

Additional Information: No Additional Information from the authors.

References

1. E. Brynjolfsson and K. McElheran, “The rapid adoption of data-driven decision-making,” American Economic Review, vol. 106, no. 5, pp. 133-139, 2016.
2. K. C. Laudon and J. P. Laudon, Management Information Systems: Managing the Digital Firm, Pearson, 2019.
3. A. N. Ikhsan, A. P. Kuncoro and A. A. Akbar, “Perancangan Aplikasi Pencatatan Transaksi Keuangan dan Inventory Berbasis Android Distributor Produk MSL Purbalingga,” Jurnal Teknologi Informasi, vol. 4, no. 2, 2020.
4. N. Saputra and B. Mulyawan, "Web-Based System Inventory at PT Sapta Tunas Teknologi," International Journal Software Engineering and Computer Science (IJSECS), vol. 3, no. 3, pp. 334-340, 2023.
5. F. Kinselton and Tony, "Perancangan Aplikasi Pencatatan Penjualan Berbasis Web pada My Sport Indonesia," Jurnal Ilmu Komputer dan Sistem Informasi, vol. 12, no. 1, 2024.
6. J. G. Djoni and Tony, "Sistem Manajemen Stok Barang Berbasis Web pada Toko Bangunan Sumber Abadi," Jurnal Ilmu Komputer dan Sistem Informasi, vol. 12, no. 1, 2024.
7. E. Turban, C. Pollard and G. Wood, Information Technology for Management: On-Demand Strategies for Performance, Growth, and Sustainability, United Kingdom: Wiley, 2018.
8. R. S. Pressman and B. R. Maxim, Software Engineering: A Practitioner's Approach, United Kingdom: McGraw-Hill Education, 2019.
9. T. O'Reilly, WTF?: What's the Future and Why It's Up to Us, United Kingdom: Random House, 2017.
10. T. F. Morris, Web Development and Design Foundations with HTML5, United Kingdom: Pearson, 2019.
11. S. Wexler, J. Shaffer, Cotgreave and Andy, The Big Book of Dashboards: Visualizing Your Data Using Real-World Business Scenarios, Germany: Wiley, 2017.
12. G. Liu, R. Zhang, B. Wan, S. Ji, and Y. Tian, "Extended Role-Based Access Control with Context-Based Role Filtering," KSII Transactions on Internet and Information Systems (TIIS), vol. 14, no. 3, pp. 1263-1279, 2020.
13. J. Osis and U. Donins, Topological UML Modeling: An Improved Approach for Domain Modeling and Software Development, Netherlands: Elsevier Science, 2017.
14. Microsoft, “Ringkasan ASP.NET,” 13 Juli 2023. [Online]. Available: https://learn.microsoft.com/id-id/aspnet/overview. [Accessed 1 September 2024].
15. D. Petkovic, Microsoft SQL Server 2016: A Beginner's Guide, Sixth Edition, United States: McGraw Hill LLC, 2016.
16. R. Kohlman, User Acceptance Test (UAT) Planning Guide: Technology Testing Series - TM 020, Amazon Digital Services LLC, 2020.
17. G. Blokdyk, Black-Box Testing a Complete Guide - 2019 Edition, Emereo Pty Limited, 2018
18. B. Blažica and J. R. Lewis, "A Slovene Translation of the System Usability Scale: The SUS-SI," International Journal of Human-Computer Interaction, vol. 31, no. 2, pp. 112-117, 2015.
19. A. Akbar, “Pengertian dan Tahapan Metode SDLC Waterfall,” Sekayu Web, 21 Mei 2021. [Online]. Available: https://www.sekayuweb.com/2021/05/pengertian-dan-tahapan-metode-sdlc-waterfall.html. [Accessed 1 September 2024].
20. G. Blokdyk, Waterfall Model a Complete Guide - 2019 Edition, Emereo Pty Limited, 2019.
21. T. M. Connolly and C. E. Begg, Database Systems: A Practical Approach to Design, Implementation, and Management (Sixth Edition), United Kingdom: Pearson, 2015.