Website-Based Point Of Sales (POS) Application Design Using The Laravel Framework

Edwin Aji Sentosa[1*], Tutik Khotimah[2], Rizkysari Mei Maharani[3]

[1,2,3]Informatics Engineering Study Program, Faculty of Engineering, Muria Kudus University, Indonesia
[*] Corresponding Author:
Email: 202151211@std.umk.ac.id

Abstract.

The data processing process at Toko Karunia in Trangkil Village still relies on manual recording. This method still poses many risks such as inaccurate product recording, slow sales reports, and a higher risk of data loss, all of which have an impact on the work efficiency of the shop owner. The purpose of this research is to solve the problem by designing a web-based Point of Sale (POS) application according to the needs of the store. The application design method uses the waterfall method with the System Development Life Cycle (SDLC) approach. The application design process in this study is carried out sequentially. Starting with requirements analysis, system design, system development, system testing, and system implementation. At the application development stage, the programming language used is PHP with the Laravel framework, with the MySql database. After the application has been designed, it will be tested through the Black Box Testing method. As a result, the main features such as login, managing categories, products, members, suppliers, transaction features, and printing reports function properly. Overall, this POS application improves all aspects such as speed, accuracy, security, and ease of accessing data. This application also strengthens Toko Karunia's position to keep competing in the digital era.

Keywords: *Point of sale; PHP; Waterfall; Laravel and MySql.*

I. INTRODUCTION

Currently, the use of digital technology in the business world has become an urgent need to help people increase productivity and business competitiveness in an increasingly tight business environment.[1]The growing number of retail businesses has led to increased competition, necessitating unique strategies for a store to survive and operate effectively and meet the owner's expectations. One way to increase a store's competitiveness is by leveraging digital technology to support its operations. This technology will significantly impact data processing and increase a store's productivity.Point of Sale (POS) is an application widely used by stores to process data for their transactions. POS is an application designed to record sales transactions, commonly known as a cashier system. Businesses in the sales sector will place POS as a crucial system because it can provide transaction information and various sales reports needed by the store.[2] However, especially in rural areas, many stores still use traditional methods and have not yet utilized this technology. These relatively traditional methods often present challenges in maintaining data accuracy, slow down store operations, and increase the risk of losing important data, all of which can impact customer satisfaction and business operations.

Karunia Store, a retail store located in Trangkil Village, Pati Regency, is one such business that still uses traditional methods for data processing, which often causes various problems, such as inaccurate inventory records, difficulties in generating sales reports, and the risk of data loss due to unstructured storage. Manual data collection is also quite time-consuming, thus reducing store operational efficiency.[3]Therefore, the POS application design for Toko Karunia is expected to provide a solution to these problems. This application design is expected to facilitate faster and more accurate sales transactions, simplify inventory data processing, and simplify reporting.[4] Based on this background, this research will design and develop a website-based POS application specifically tailored to the operational needs of Toko Karunia. This application is expected to significantly improve the store's operational efficiency and effectiveness, while also preventing the risk of losing critical data.

II. METHODS

Data Collection Techniques

In this study, data collection techniques used qualitative methods through direct observation and interviews. Observations were conducted by observing the transaction process at Karunia Store. Researchers observed how the owner handled various activities, such as processing inventory data and transactions with customers, which were still carried out manually. Furthermore, a direct interview was conducted with the owner of Karunia Store, Mr. Tugiman. Through this interview, information was obtained regarding obstacles encountered during the sales process and the need for a POS system to support store operations.

Waterfall System Development Method

The research method used includes application design. In the design phase of this website-based POS system, the Waterfall method was used, which is one of the approaches in the System Development Life Cycle (SDLC). SDLC is a frequently used approach in system design due to its systematic structure.[5]Each stage is carried out in stages so that the results of one stage become the basis for the next stage. Figure 1 below shows the stages in the waterfall method:

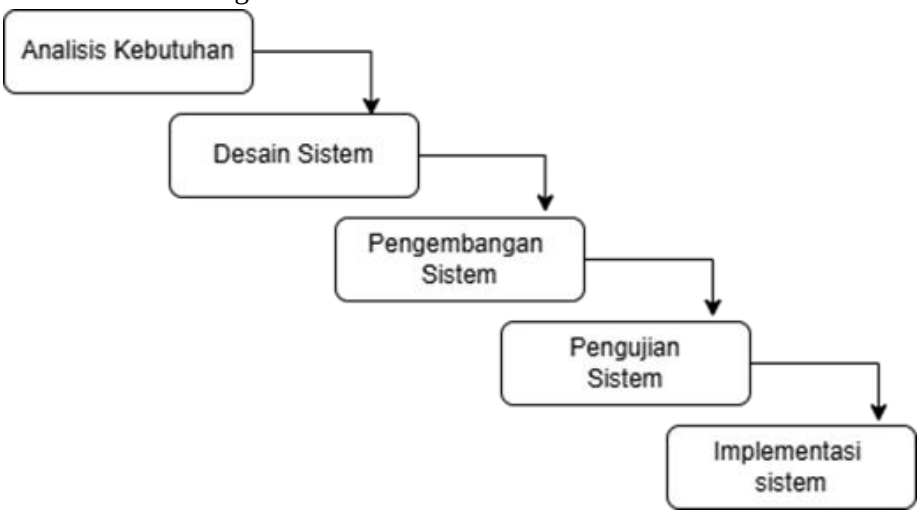

Fig 1. Waterfall Method

1. Needs Analysis

Data was collected through observations and interviews with the owner of the Karunia shop. At this stage, a system requirements analysis will be conducted through direct interviews with the owner of the Karunia shop to obtain the data needed for designing the POS system. The results of these interviews will then be used to identify various functional and non-functional requirements in the system to be developed. Functional requirements include product data processing, transactions, and sales reports. Non-functional requirements include security, ease of use, and system performance. All of these analysis results will form the basis for the next design stage.[6]

2. System Design

At this stage, the design is carried out based on the results of the previous needs analysis. The system design begins with the use of Unified Modeling Language (UML) to describe the features and interactions within the system, including use case diagrams to show user interactions with the system, activity diagrams to explain the flow of activities, and sequence diagrams to show the sequence of interactions between objects. This stage also includes the creation of a User Interface (UI) in the form of a mockup that includes the dashboard display, category pages, product pages, transaction pages, and other pages needed in the POS system.

3. System Development

The system development phase was based on a previously created design, using the PHP programming language and MySQL as the database, along with the Laravel framework. Laravel was chosen for its supporting features for application development, such as Laravel Breeze for system authentication and the Blade Template Engine for the interface. On the UI side, the system was designed using a combination of Bootstrap as the CSS framework and JavaScript for a more responsive display.[7]

4. System Testing

After the application is developed, testing is carried out using the Black Box Testing method to ensure that the application meets user needs and that all functions function properly. Black Box Testing is performed by examining input and output without examining all internal system processes. This testing focuses on core functions such as login, product data processing, sales transactions, and other system features. Test results are presented in a table containing information such as test scenarios, expected outcomes, test results, and conclusions for each feature.[8]Through this method, it is expected that the system responds to the input correctly in its entirety.

5. System Implementation

To ensure the POS application is ready for use by users, system training is required for relevant users on how to operate the system. This training also serves as a trial run to ensure the system requirements at Toko Karunia meet the owner's expectations. The tools used to run this POS system are the Google Chrome browser, XAMPP as the local host, and MySQL as the database.

III. RESULT AND DISCUSSION

A. System Design

1. *Use Case*Diagram

*Use case*A diagram is a type of diagram that functions to briefly describe the interaction between actors (users) and the system.[9]This diagram presents the system requirements from a user perspective and shows how each actor interacts with the system's features. Figure 2 below is a use case diagram of the system:

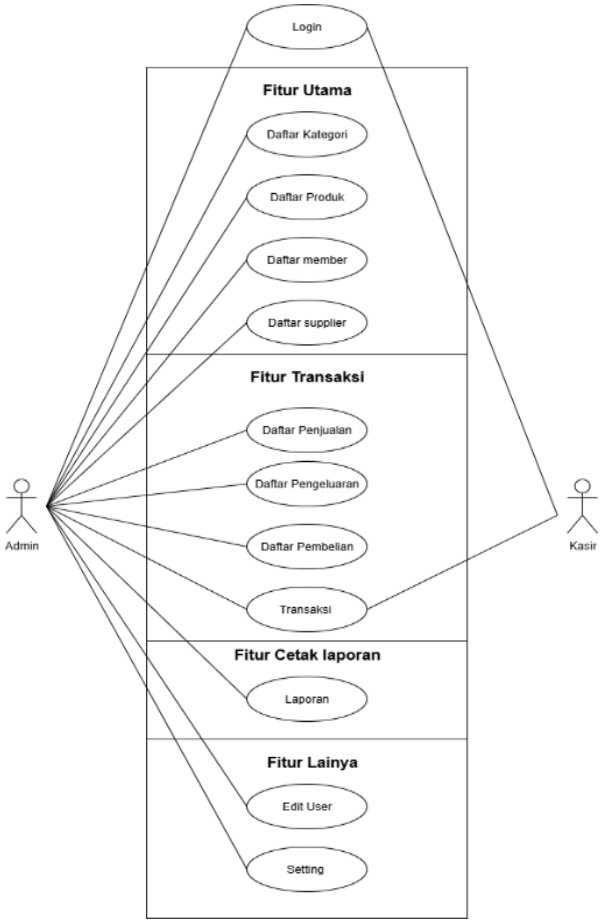

Fig 2. *Use Case* Diagram

2. Class Diagram

*Class*A diagram is a model used to describe the structure of a database and object class.[10]This diagram presents an overview of the entities in the system, complete with rules that describe the system's behavior. Each class is depicted as a rectangle containing the class name, attributes, and methods, making it easier to understand the flow of the data structure in the system designed, as shown in Figure 3 below.[11].

Fig 3. *Class* Diagram

3. Sequence Diagram

*Sequence*A diagram is a modeling tool used to illustrate the sequence of interactions between objects in a system. This diagram presents the process flow of a feature or activity, from initiation to the response received, by showing the activities between objects sequentially. This diagram allows the system's logical flow to be visualized coherently and clearly, as in Figure 4 below.

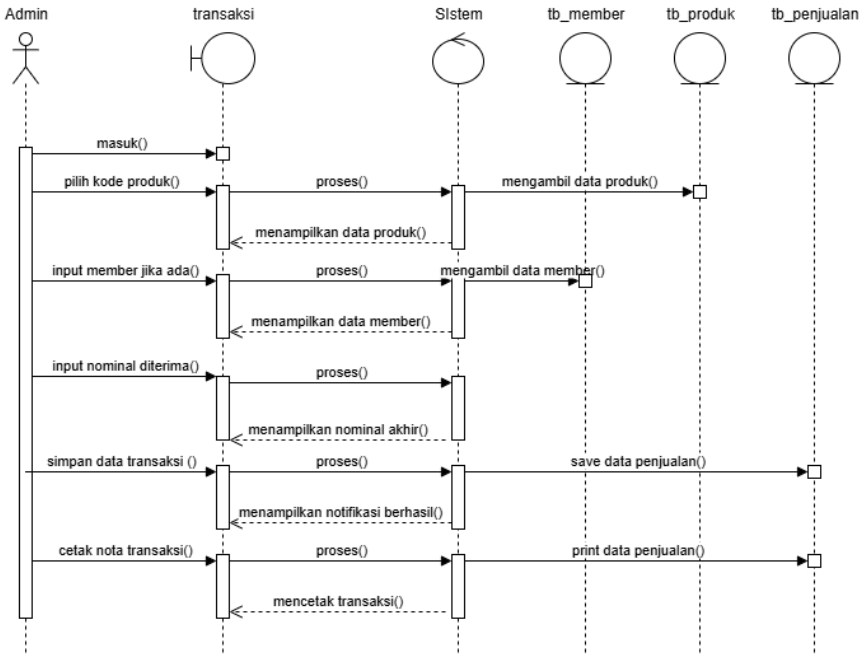

Fig 4. *Sequence* Transaction Diagram

Figure 4 shows the user interaction flow during a transaction. The process begins when the user accesses the transaction menu and the system displays the transaction features. Next, the user selects a product code, and the system displays the price of the selected product. The user can select a member if needed, and the product price will automatically receive a discount. The user then enters the amount the buyer paid. The transaction data can then be saved and printed as a transaction receipt.

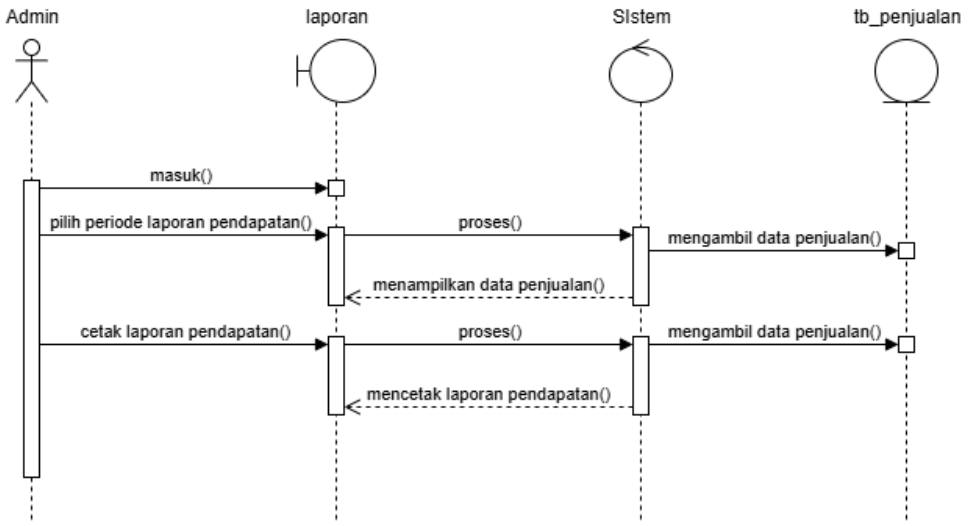

Fig 5. *Sequence* Print Report Diagram

Figure 5 shows the process flow for a user to print a report. The process begins when the user accesses the report menu in the application. The user selects the time period to print, which will then be displayed as a PDF file. The user can then print the selected report directly.

4. Activity Diagram

*Activity*A diagram is a model that depicts the processes within a system. This model systematically depicts the overall system activity, as shown in Figure 6 below.

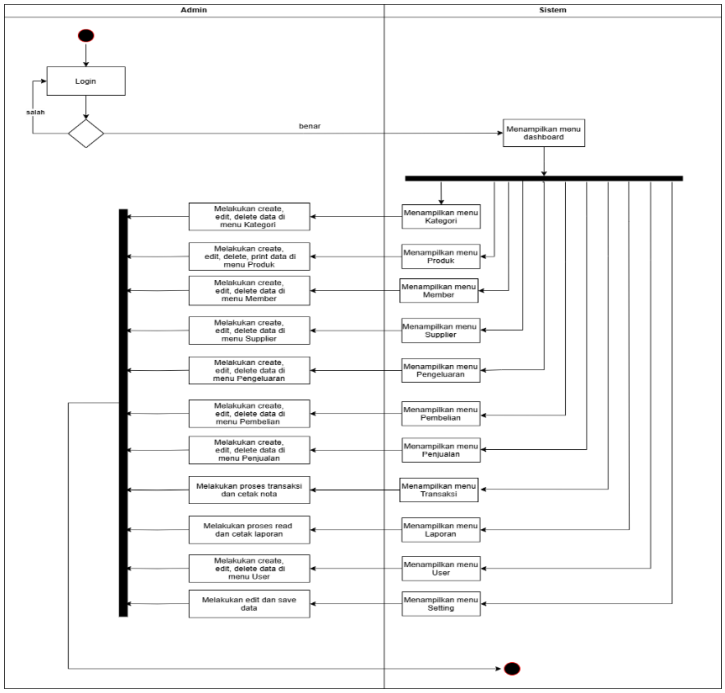

Fig 6. *Activity* Diagram

B. System Implementation

Admin Features

1. Login Page

This page is used to restrict access to registered users only. The login page is designed with a simple and responsive interface, containing username and password input, and a login button. Once the login form is correctly filled out and the login button is pressed, users will be redirected to the application appropriate to their role. Those who fail to log in will remain on this page.

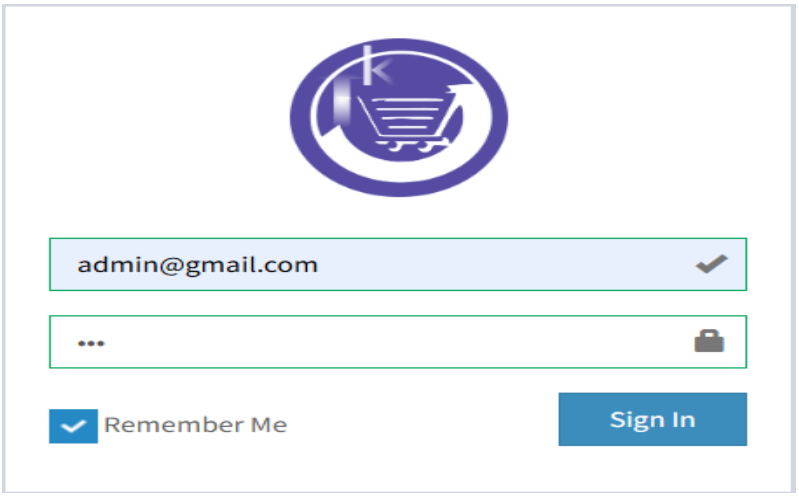

Fig 7. Login Page

2. Dashboard Page

This page is the main page in this system, a page that displays information on the number of product categories, number of products, number of members, number of suppliers and below it displays a graph of income with a time period each month.

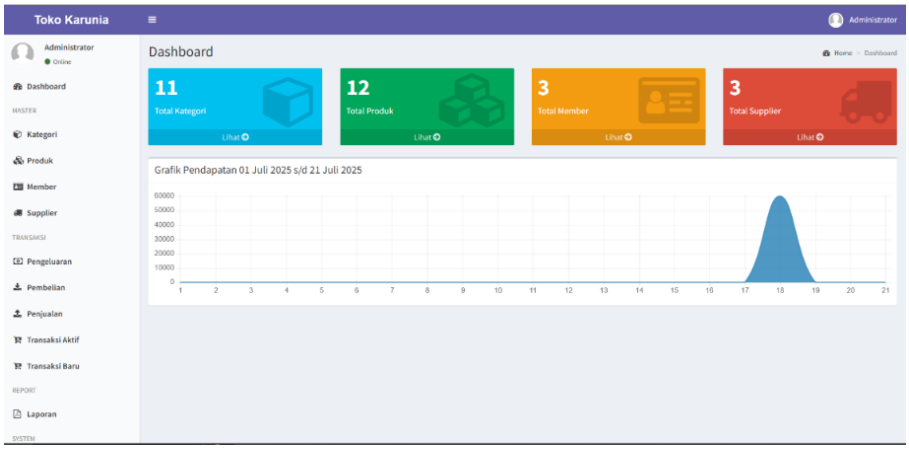

Fig 8. Dashboard Page

3. Product Category Page

To display a list of product categories, users must click the category menu. This will then display the product categories available in the store, such as groceries, snacks, beverages, and other product categories. Users can add, delete, and edit product categories as needed.

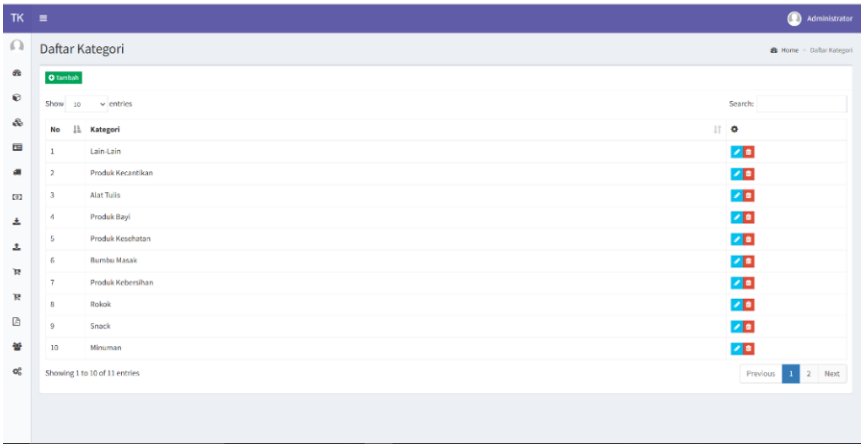

Fig 9. Product Category Page

4. Product Page

This page displays products by category, tailored to your store's needs. This menu displays information such as the product code, product name, product category, product brand, purchase and sale prices, and stock availability. Admins can also add, delete, edit, and print barcodes.

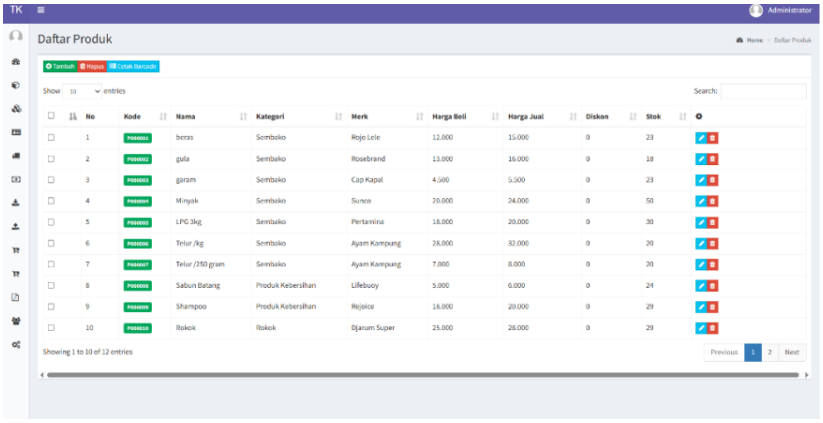

Fig 10. Product Page

5. Member Page

This member menu displays the names of members who have subscribed to Toko Karunia. This menu displays their name, phone number, and address. Users can add, delete, and edit member details as desired.

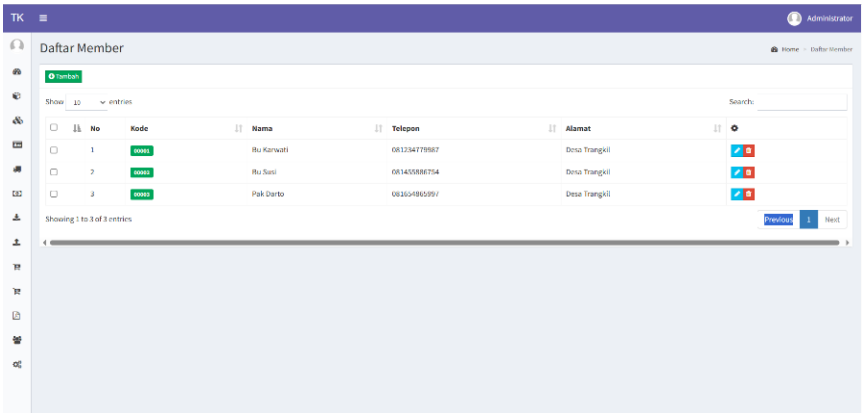

Fig 11. Member Page

6. Supplier Page

This page displays the names of the store's suppliers, along with addresses and phone numbers. Users can add, delete, and edit data according to the store's needs.

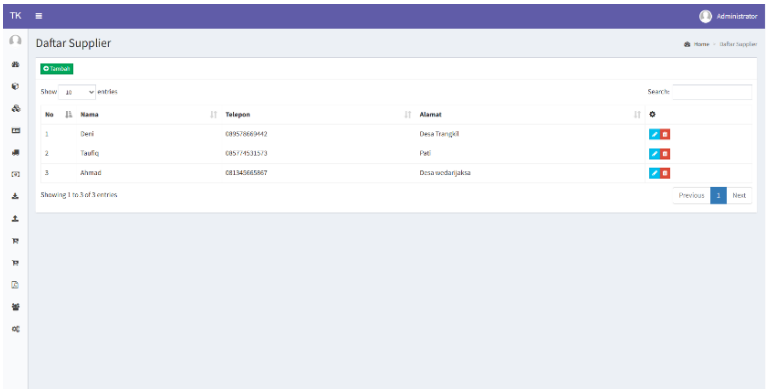

Fig 12. Supplier Page

7. Expenses Page

This page allows users to process Toko Karunia's spending data. In this menu, users can add, delete, and edit each spending item, which will automatically be displayed in a chart on the main dashboard.

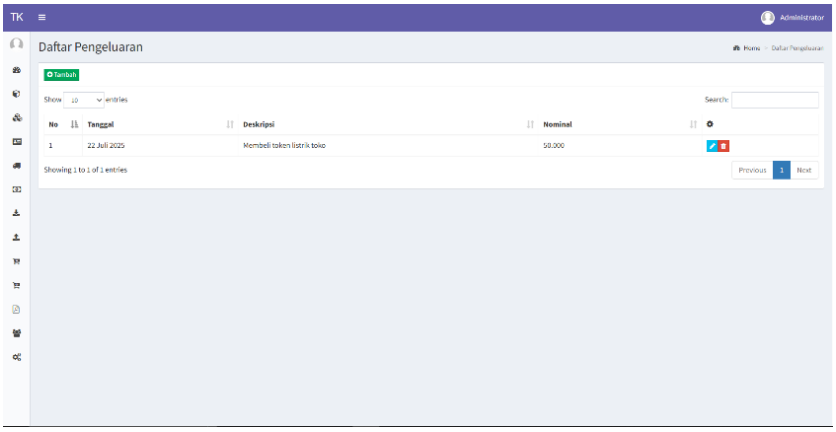

Fig 13. Expenses Page

8. Purchase Page

This page displays data on product stock purchases from suppliers by the owner of Toko Karunia. Users can view the supplier's name, price, and the number of products purchased. The data will automatically appear in a chart on the main dashboard.

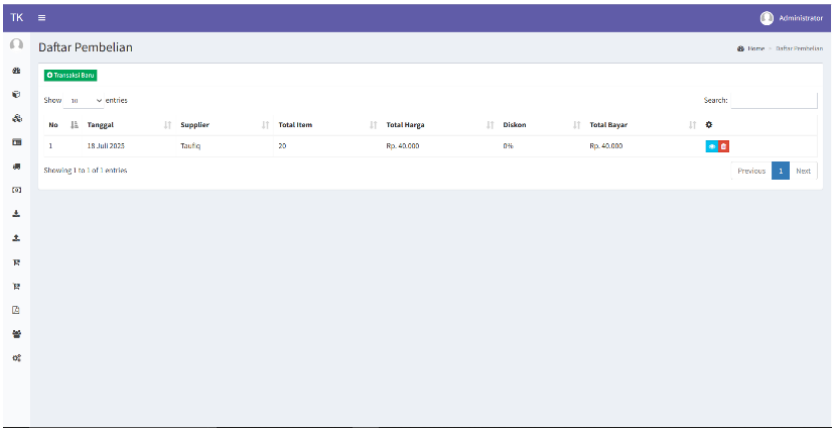

Fig 14. Purchase Page

9. Sales Page

This page displays the product sales history during the transaction. Users can delete and view products that have been sold. This data will automatically appear in the chart on the main dashboard.

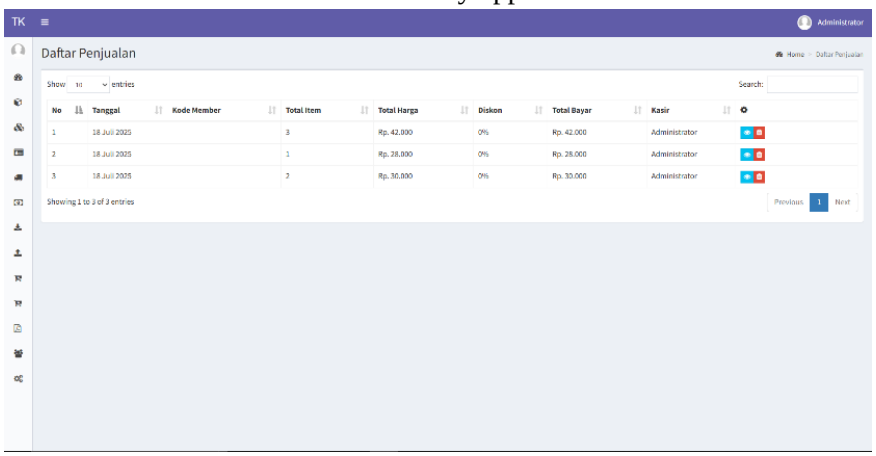

Fig 15. Sales Page

10. Transaction Page

This page is the main feature of the Toko Karunia POS application. This menu allows users to conduct transactions with customers. Users can save transaction data and print receipts directly.

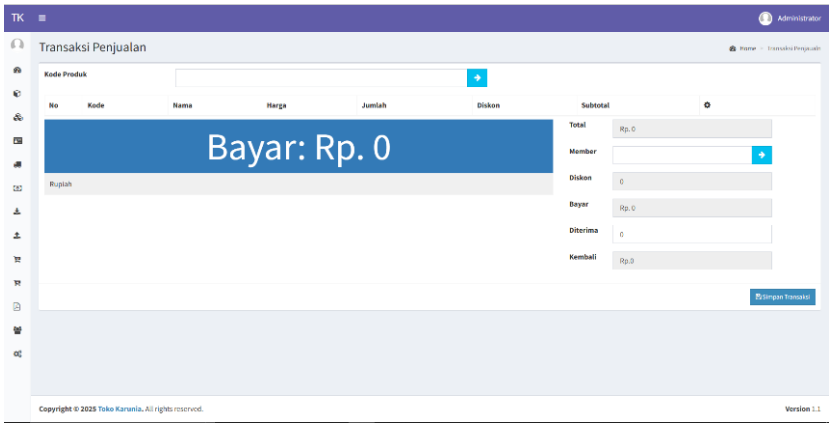

Fig 16. Transaction Page

11. Report Page

On this page, users can view and print their revenue report data. Printing a revenue report can be done by selecting the time period to print, and the system will then display it in PDF format. Users can then print the report directly according to their needs.

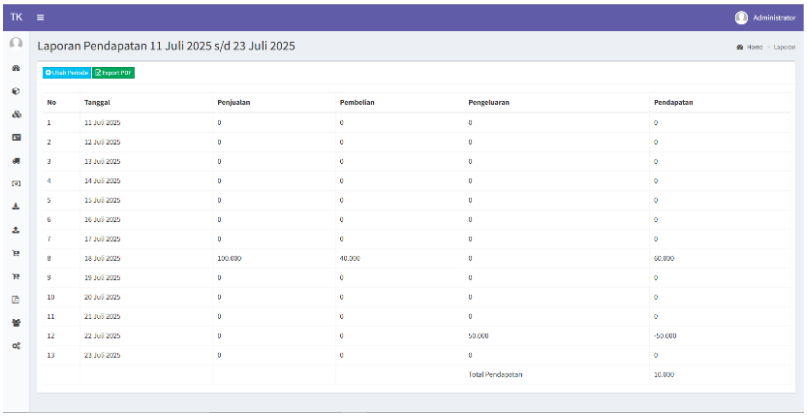

Fig 17. Report Page

12. User Page

This page is an application feature that allows you to add users to the Toko Karunia POS application. This feature allows you to add, delete, and change email addresses and passwords directly according to your needs.

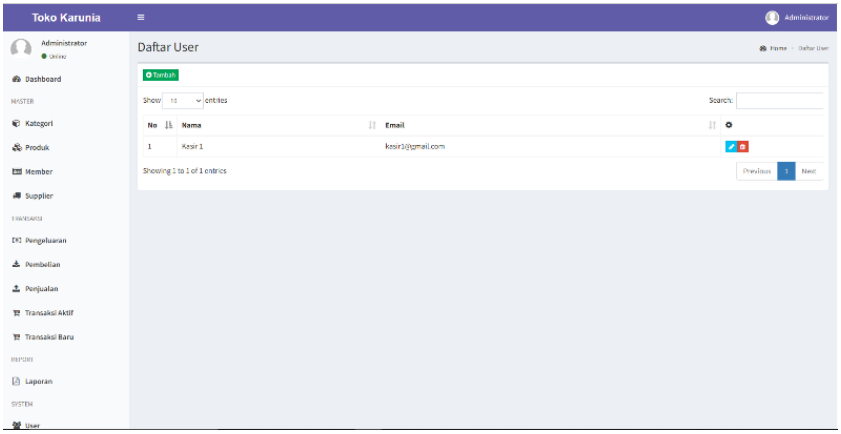

Fig 18. User Page

13. Settings Page

On the settings page, users can edit app content, such as the store name and address. This feature can also change the store logo to JPG image format.

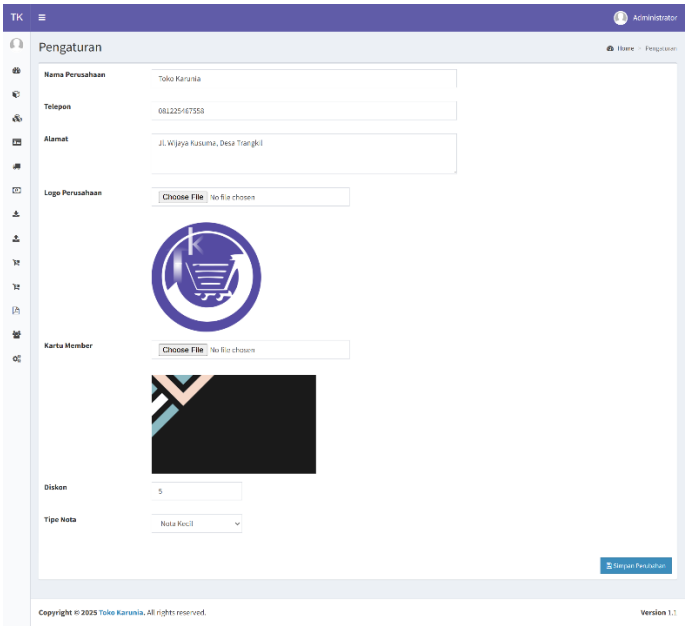

Fig 19. Settings Page

C. System Testing

System testing was conducted using the Black Box Testing method, focusing on core admin functionality. Testing covered various essential features of the website-based POS system, such as login, category management, product management, member management, and other features. Each test scenario was deemed successful if the tested feature produced the expected output. Complete test results are presented in Table 1 below.

Table 1. Admin Blackbox Testing Results

No	Testing Scenario	The results that It is expected	Test Results	Conclusion
1	Login	The system verifies and redirects to the main page.	After entering valid login data, the user enters the main page.	Succeed
2	Access the dashboard page	The system displays descriptions and chart graphs.	The menu display is displayed in a structured manner and graphics can be displayed.	Succeed
3	Category menu	The system can add, edit and delete data.	Data can be added, edited or deleted as needed.	Succeed
4	Product Menu	The system can add, edit, delete and print barcodes.	Data can be added, edited or deleted as needed and barcodes can be printed.	Success
5	Member menu	The system can add, edit and delete data.	Data can be added, edited or deleted as needed.	Succeed
6	Supplier menu	The system can add, edit and delete data.	Data can be added, edited or deleted as needed.	Succeed
7	Expense menu	The system can add, edit and delete data.	Data can be added, edited or deleted as needed.	Succeed
8	Purchase menu	The system can add, edit and delete data.	Data can be added, edited or deleted as needed.	Succeed
9	Sales menu	The system can display transaction history and delete data.	Transaction history is displayed in order and data can be deleted.	Succeed
10	Transaction menu	The system can process transactions and transaction data can be saved and printed.	The transaction process is carried out properly and data can be saved and notes can be printed.	Succeed
11	Report menu	The system displays income reports according	Income reports are displayed according to the selected time	Succeed

No	Testing Scenario	The results that It is expected	Test Results	Conclusion
		to the required time period and the report data can be printed.	period and data can be printed.	
12	User menu	The system can add, edit and delete users.	*User*can be added, deleted, and edited, such as changing email and password	Succeed
13	Settings menu	The system displays a settings form such as editing the shop name, address and changing the application logo.	Shop information such as name, address and logo can be changed according to the shop owner's needs.	Succeed

Table 2. Cashier Blackbox Testing Results

No	Testing Scenario	The results that It is expected	Test Results	Conclusion
1	Login	The system verifies and redirects to the main page.	After entering valid login data, the user enters the main page.	Succeed
2	Transaction menu	The system can process transactions and transaction data can be saved and printed.	The transaction process is carried out properly and data can be saved and notes can be printed.	Succeed

Table 3. Recapitulation of System Testing Success Rate

Role	Number of Scenarios	Succeed	Fail	Success Rate
Admin	13	13	0	100%
Cashier	2	2	0	100%
Total	**15**	**15**	**0**	**100%**

Final System Testing Result Calculation

$$\text{Success Rate} = \left(\frac{\text{Jumlah Skenario Berhasil}}{\text{Jumlah Total Skenario}}\right) \times 100\%$$

$$= \left(\frac{15}{15}\right) \times 100\% = 100\%$$

Based on the results of black-box testing conducted on 15 scenarios on the website-based POS system, 15 scenarios were successful, resulting in a 100% success rate. This indicates that all system functions, both from the admin and cashier sides, are running well and in line with expectations.

IV. CONCLUSION

Based on the results of the study entitled "Designing a Website-Based Point of Sales (POS) Application Using the Laravel Framework," it can be concluded that the system is effective. This system was designed to overcome obstacles in the management process at Karunia Store, which was previously carried out manually and with limited capabilities. With a website-based system, data processing becomes more efficient, accurate, and easily accessible to store owners. Furthermore, data security in the system is improved, because the admin can access comprehensive sales report data from the admin and cashier. This way, undesirable things such as transaction data manipulation can be avoided. Customers can easily view the product catalog, select materials, check out, and confirm payments through a responsive and user-friendly system interface. Furthermore, this system has been tested using the black-box testing method with a perfect success rate. These results indicate that all main features can run well. This system is expected to be a relevant and useful solution in supporting Karunia Store to compete in the digital era.

REFERENCES

[1] BS Prayogi, I. Fitri, and R. Nuraini, "Website-Based Point of Sale Application at Tegar Grocery Store,"***J. JTIK (Journal of Technology, Information and Communication)***, vol. 6, no. 2, pp. 260–266, 2022, doi: 10.35870/jtik.v6i2.411.

[2] H. Oktarina and J. Fitra, "Design and Construction of Point of Sale (POS) Application: Case Study of Trustmart Mini Market,"***J. Softw. Eng. Technol.***, vol. 1, no. 2, pp. 37–42, 2021.

[3] A. Cetageti, A. Surahman, and A. Sucipto, "Implementation of Point of Sales (POS) Technology as a Web-Based Ornamental Fish Sales Information Media. Case Study: King Koi Group."***TELEFORTECH J. Telemat. Inf. Technol.***, vol. 2, no. 2, pp. 33–39, 2022, [Online]. Available: https://ejurnal.teknokrat.ac.id/index.php/telefortech/article/view/1999.

[4] O. Herdiansyah, Pratama, "Designing a Website-Based Point of Sale Information System at the Azam Grosir Store Using the Waterfall Method,"***J. Inform. Pamulang University***, vol. 6, no. 2, pp. 388–394, 2021, [Online]. Available: http://openjournal.unpam.ac.id/index.php/informatika/article/view/11773.

[5] A. Mulyana and U. Rusmawan, "Design and Construction of a Web-Based Point of Sale (POS) Information System (Case Study of Andorio Store),"*Major of Science UNIKOM*, vol. 21, no. 1, pp. 43–50, 2023, doi: 10.34010/miu.v21i1.10689.

[6] S. Suprianto, M. Fadlan, and D. Prayogi, "Designing a Web-Based Point of Sale Application at the Salfa Tarakan Project Store,"*Batik*, vol. 25, no. 2, pp. 624–631, 2021, doi: 10.46984/sebatik.v25i2.1519.

[7] D. Saputra*et al.*, "Building a Website at PT Surya Bintang Indonesia Using PHP and MySQL," ***J. Tek. Inform. Mahakarya JTIM***, vol. 5, no. 1, pp. 17–24, 2022.

[8] AR Naufal, DA Nawangnugraeni, and AT Suseno, "Design and Construction of a Multi-Outlet Point of Sale Information System Using the Laravel Framework at the Itsnu Pekalongan Cooperative,"***J. Tech. Inf. and Comput.***, vol. 5, no. 2, p. 280, 2022, doi: 10.37600/tekinkom.v5i2.591.

[9] B. Hanggoro and F. Yanti, "Designing a Web-Based Point of Sale Application at Kang Udin's Shop,"***OKTAL J. Comput. Sci. and Sci.***, vol. 99, no. 99, pp. 379–387, 2022, [Online]. Available: https://journal.mediapublikasi.id/index.php/oktal/article/view/815/933.

[10] F. Amelia Sari Lubis, S. Sahara Lubis, and B. Hendrik, "Designing an Inventory System for Herbicide Stock at UD. Anugrah Jaya Tani Using PHP Programming Language and MySQL Database,"***J. Science Inform. Terap.***, vol. 2, no. 2, pp. 50–55, 2023, doi: 10.62357/jsit.v2i2.167.

[11] S. Ramdany, "Application of UML Class Diagram in Designing Web-Based Library Information Systems,"***J. Ind. Eng. Syst.***, vol. 5, no. 1, 2024, doi: 10.31599/2e9afp31.