

Web-Based Cashier Management, Inventory, And Sales Management Application Design

Muhammad Khotibul Umam^{1*}, Tutik Khotimah², Ahmad Jazuli³

^{1,2,3} Teknik Informatika, Fakultas Teknik, Universitas Muria Kudus, Jawa Tengah, Indonesia

*Corresponding Author:

Email: khotibul2018ab@gmail.com

Abstract.

The rapid development of information technology has had a significant impact on various industries, including the retail sector. However, Atika Store still relies on manual methods in managing its cashier and inventory, leading to inaccuracy, inefficiency, and poor control over stock and sales reports. To address these issues, this study employs an applied research method aimed at producing practical solutions that can be directly implemented. A web-based application is developed using the Waterfall system development method, which consists of requirement analysis, system design, implementation, testing, and maintenance. The system integrates sales, inventory management, and cashier functions into a single platform. Key features include automatic receipt printing, sales transaction recording, real-time stock updates, and role-based user access. It is expected that this system will help MSME owners improve operational productivity, minimize manual errors, and support more effective decision-making.

Keywords: Retail System; Cashier Application; Inventory Management; Web-Based and Waterfall Method.

I. INTRODUCTION

The rapid development of information technology has had a significant impact on various industries, including retail. As a small retailer, Toko Atika faces various challenges in business management, particularly cashier management, inventory management, and sales management. The ongoing manual processes create several problems, such as recording errors, delayed product deliveries, and inaccurate sales reports. This can impact Toko Atika's ability to compete and thrive in an increasingly competitive market. From the background presented[1] It can be concluded that for traditional retail stores like Toko Atika, the implementation of a modern, integrated information system is crucial. Non-computerized business processes often lead to errors in inventory recording and management, impacting operational efficiency and profitability. In this context, the development of a web-based cashier, inventory, and sales management application using Rapid Application Development (RAD) techniques is expected to improve the efficiency and effectiveness of store operational management.

In the context of designing a web-based cashier, inventory, and sales management application for Toko Atika, it can be concluded that Toko Atika, as a micro, small, and medium enterprise (MSME), faces various challenges in transaction and inventory management. The manual processes that are still used increase the risk of loss and errors in recording. Observations show that transactions carried out manually are inefficient, often experience errors in profit calculations, and irregular management of accounts receivable and payables, which can disrupt cash flow and operations.[2] The development of a web-based application is expected to automate and improve the efficiency of Toko Atika's business processes. This application is designed to simplify transaction recording, inventory management, and profit and loss reporting. The application must include essential features, such as cashier management, inventory management, and sales reporting, accessible to various users in the store.[3]

II. METHODS

Research methodology is a systematic method or procedure used to collect, analyze, and interpret data to achieve research objectives. This study uses the Applied Research approach, as used in research. Applied research is a type of applied research that aims to solve practical problems and produce real-world solutions to specific needs. This research is focused, contextual, and oriented towards direct application, particularly in the development of information systems capable of improving operational efficiency in real-world environments, such as Toko Atika.[4] Data Collection Literature Study Observation Interview.

Validation Method

The research article entitled "Design of a Web-Based Cashier, Inventory, and Sales Management Application" has been examined, reviewed, and declared worthy of approval as part of a scientific paper. This research uses the Applied Research method with a Waterfall system development approach, which has gone through the stages of needs analysis, system design, implementation, testing, and maintenance. Based on the results of the review and validation, this work is declared to meet the academic criteria and can be used as a reference in the development of a web-based cashier, inventory, and sales management information system in the MSME environment.[5] This application was developed using the waterfall methodology, a development approach that is now quite well-known and widely adopted by software developers. This method falls into the category of application development models and is classified as a classic life cycle, emphasizing the implementation of sequential and systematic stages. This model can be likened to a waterfall, where each step is carried out gradually from the top to the bottom. This model consists of several stages, namely:

a. Analysis

The purpose of data collection through literature review, observation, and interviews is to identify and formulate system requirements. The analysis is conducted to understand the current business processes and clarify functional requirements such as transaction recording, inventory management, and financial reporting, as well as non-functional requirements such as system security and access speed.[6]

b. Design

After determining all system requirements, the design phase begins, which involves designing the system architecture, user interface, and database structure. At this stage, modeling diagrams such as Use Case Diagrams demonstrate system functions, Sequence Diagrams show the interaction flow between components, and Class Diagrams define interconnected data structures.

c. Implementation

This stage involves translating the system design into program code. To support the Model-View-Controller (MVC) architectural pattern, development is conducted using the PHP programming language and the CodeIgniter framework. Currently, data is managed through a MySQL database system, which allows for fast and organized access.

d. Testing

To ensure the system performs as expected, black-box testing methods are used to test application functionality based on input and output, regardless of the internal structure of the code. This method is used on all major system modules, including inventory recording, sales, receipt printing, and financial reporting.

e. Maintenance

Once the system is operationally deployed, this phase is completed. Maintenance involves fixing bugs, improving system performance, and adjusting features based on user feedback. This is done consistently to ensure the system remains relevant and meets evolving operational needs.

III. RESULTS AND DISCUSSION

System Design

In the system design stage, Data Flow Diagrams (DFDs) are used to illustrate data flow and key processes such as order entry, shipping, billing, and cash receipts. DFDs are also elaborated down to Level 1 to show activity details, such as stock checking, invoicing, and payment recording. In addition, Entity Relationship Diagrams (ERDs) are also used to model the data structure and relationships between entities in the system. This DFD and ERD approach was chosen because it is able to provide a clear **Fig.** of the data flow and relationship structure required in the development of the Atika Store cashier and inventory management information system.[7]

Data Flow Diagram (DFD Level 0)

A DFD is used to illustrate how data flows within a system and how processes interact with each other. In the Toko Atika system, the main data flow centers on sales transactions, inventory management, payment recording, and report generation.

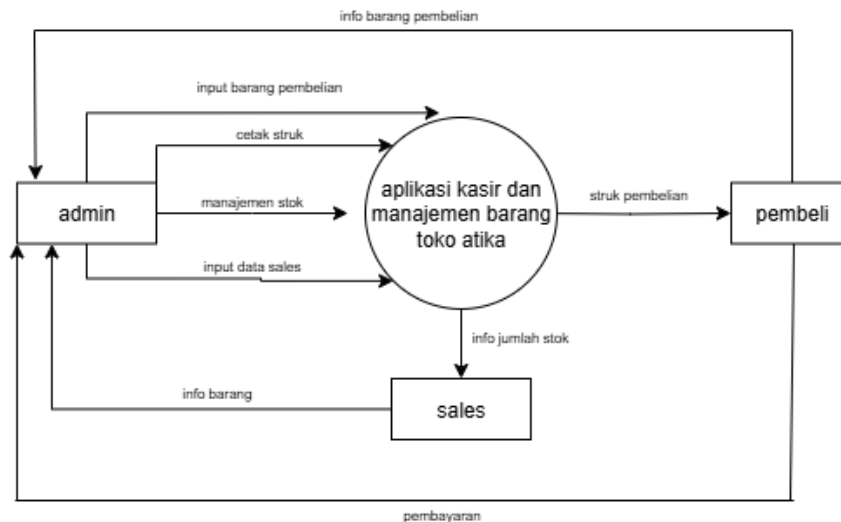


Fig 1. DFD Level 0

Figure 1 shows a Level 0 DFD divided into three, namely the admin, buyer, and seller interacting with the system in the system context diagram "Atika Store Cashier and Inventory Management Application". The administrator monitors item and transaction data, buyers make purchases, and sales outside the store are recorded. The system processes input from all three entities and then provides output according to its function. The arrows indicate data flow, but not the details of internal processes because this is a general overview of the system.

Data Flow Diagram (DFD Level 1)

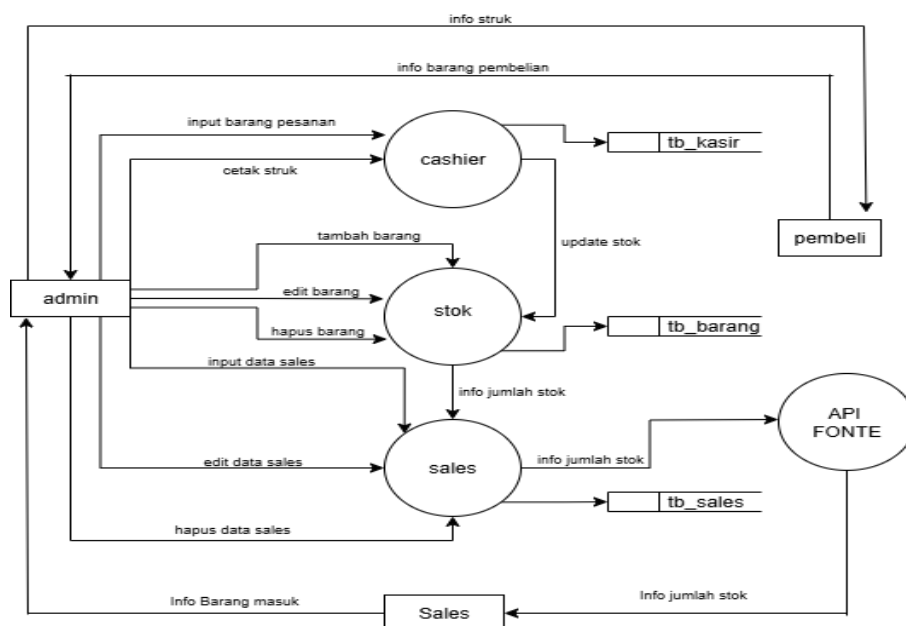


Fig 2. Dfd Level 1

Figure 2 shows the Level 1 DFD for the Atika Store cashier system. The system has three main processes: cashier, stock, and sales. The cashier process handles customer transactions, stock is managed by the admin to monitor item availability, and sales sends stock notifications to salespeople via the FONTE API. All of these processes are integrated, enabling transaction management, stock updates, and automated notifications.

ERD

In journals, Entity Relationship Diagrams (ERD) are used to model databases and relationships between entities, such as customers, sales, inventory, production, and owners, so that information systems can run efficiently and accurately. In designing the Atika Store cashier system, the ERD approach is also used with the main entities of items, transactions, transaction details, users, and customers. Each item can

appear in many transaction details, one transaction contains several transaction details that record items, quantities, and subtotals, while customers act as buyers and users as cashiers/admins. The relationship between entities is generally one-to-many so that transaction and inventory data management is faster, neater, and more organized.[8]

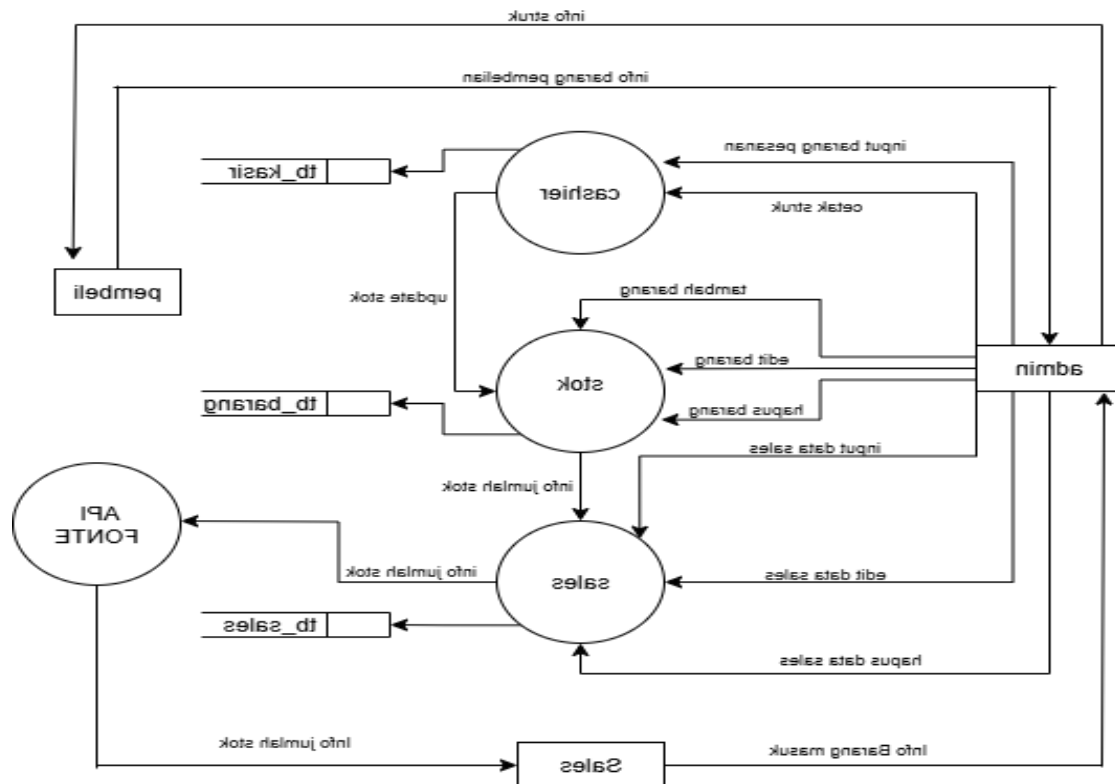


Fig 3. ERD

Figure 3 explains the relationship between entities in the cashier management system depicted in an entity relationship diagram (ERD). This ERD shows the relationship between entities such as Stock, Cashier (Chasier), Check Out, Sales, Admin, and the Fonte API. The Stock entity stores item data such as name, price, stock, and photo, which are used by the cashier through the Chasier entity. The admin performs the data input process through the incoming relationship. The sales company's WhatsApp number is integrated into the Fonte API for automatic messaging and is connected to stock. Starting with external communication, sales, and stock management, this diagram shows the system flow.

Implementation

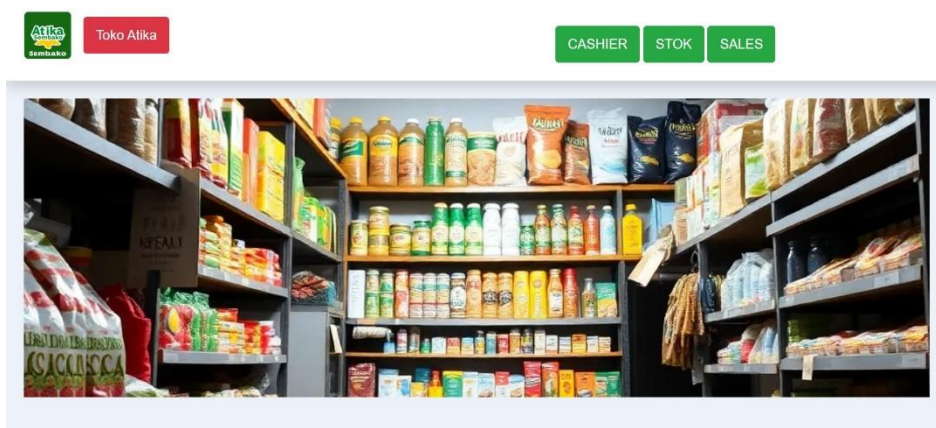


Fig 3. Main Page

Figure 3 shows the main page of the website. This section displays the main navigation menu, which includes Cashier, Stock, and Sales, making it easier for users to access important features.

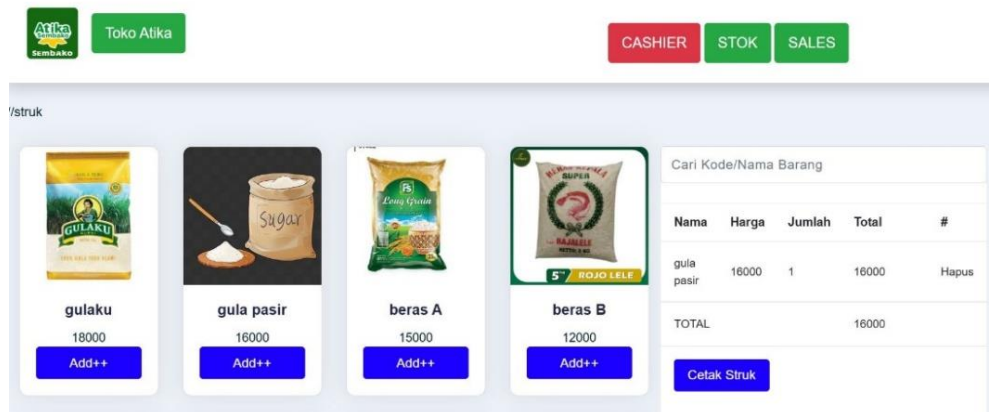


Fig 4. Cashier Page Containing A List Of Basic Food Products Such As Granulated Sugar, Rice, And Packaged Sugar

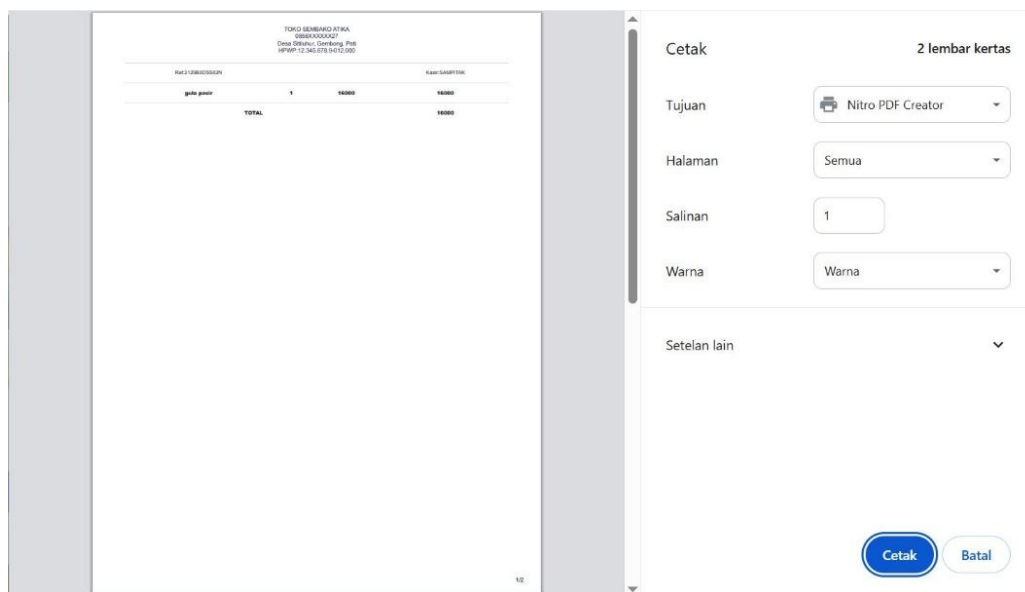


Fig 5. Print Receipt Feature

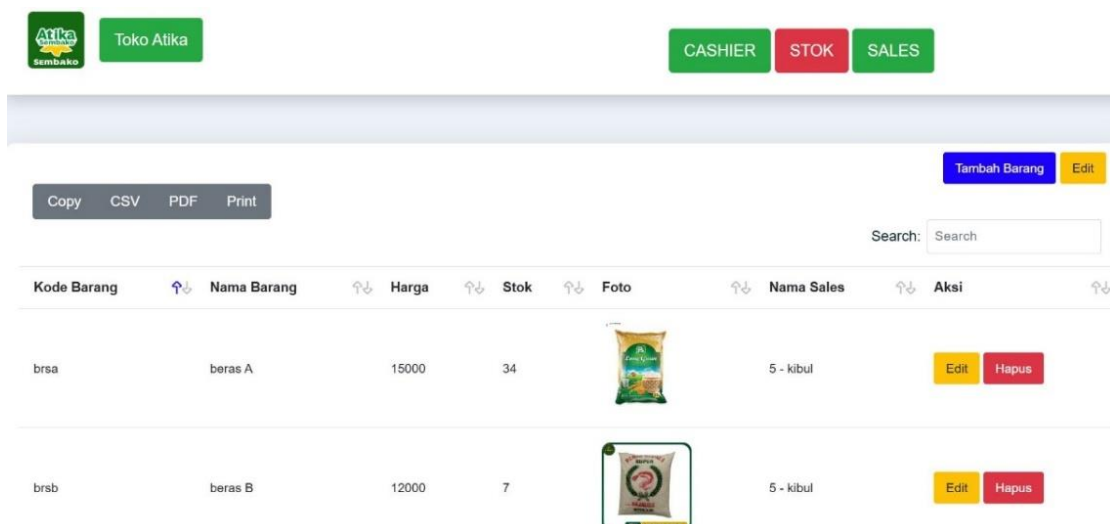


Fig 6. Search Item Menu by Item Code

In Figure 6, this menu makes it easier for the admin to search for items purchased by customers by entering the item code.

Fig 7. Adding Items for Sale

In Figure 7, the admin can add merchandise that has not been registered in the application or change the price of the item.

#	Id Sales	Nama Sales	No Wa	
				Tambah
1	5	kibil	085229360102	Edit Hapus
2	6	ofikul rokok	081770263372	Edit Hapus

Fig 7. Sales List

Figure 7 shows a list of salespeople connected to the application. The table contains information about the salesperson's ID, name, and WhatsApp number, facilitating communication and recording sales activity.

Fig 8. Adding No Sales to the Application

In Figure 8, the admin can add a sales telephone number to connect to the application so that if stock is running low, there will be an automatic notification from the application.

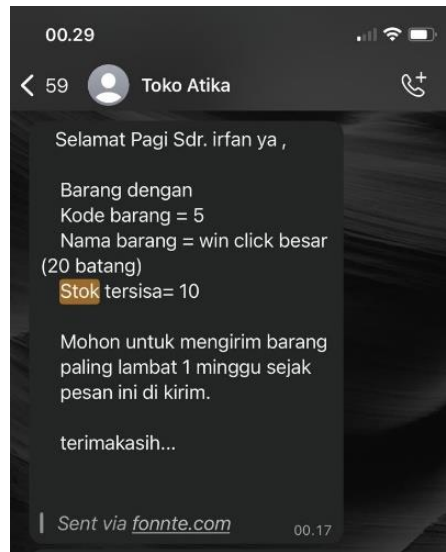


Fig 9. Fonte Api Implementation

Figure 9 shows the Fonte API implementation on the Atika Store system. It can send notifications to buyers via WhatsApp when a product's stock reaches a certain limit, such as 10 or 3, and sends a message containing the product code, product name, and remaining stock. To enable faster, more effective, and more responsive stock management, these notifications are sent through the fonnte.com platform to ensure sales reps promptly replenish products.

System Testing

To ensure that each feature functions as intended, the system was tested using the Black Box Testing method. Testing involved the login module, decoration data input, customer data input, transaction data input, payment data input, and WhatsApp booking data input. Additionally, the automatic stock notification feature was tested using the Fonte API. This sends a message to sellers when stock reaches 10, 5, or runs out. Test results showed that all features functioned well, were responsive to input, and supported the store's smooth operation.

Table 1. System Testing

No.	Tested Features/Modules	Input Data / Conditions	Expected results	Test Results
1	Home Page (Dashboard)	Access the main page from the browser	The main system dashboard view appears.	Valid
2	Select Item (Figure 4)	Click on the available items in the list	Item successfully added to cart	Valid
3	Print Receipt (Figure 5)	Make a transaction and click the print button	Automatically printed purchase receipt	Valid
4	Search for Items (Figure 6)	Enter a valid item code	Items matching the code appear in the search list.	Valid
5	Search for Items (Figure 6)	Enter invalid item code	The notification "item not found" appears.	Valid
6	Add Item (Figure 7)	Fill in the add item form with complete data	Item data successfully saved to database	Valid
7	Add Item (Figure 7)	Leave one of the required inputs blank.	The system rejects and displays an error message.	Valid
8	Sales List (Figure 7 second)	Access the sales list menu	Displays a list of names and WA numbers of connected sales	Valid
9	Add Sales No. (Figure 8)	Enter a valid sales WA number and save it	Number successfully saved and connected to Fonte API	Valid
10	Add Sales No. (Figure 8)	Input number is empty or incorrectly formatted	The system rejects and displays an error message.	Valid

No.	Tested Features/Modules	Input Data / Conditions	Expected results	Test Results
11	Stock Notification via Fonte API	Remaining stock = 10 after transaction	Sales received a notification: “[Name] has 10 items left in stock, please recheck your stock immediately.”	Valid
12	Stock Notification via Fonte API	Remaining stock = 5 after transaction	Sales receives a notification: “Attention! [Name] has 5 items left in stock.”	Valid
13	Stock Notification via Fonte API	Remaining stock = 0 after transaction	Sales received a notification: “[Name] is out of stock. Please replenish immediately!”	Valid
14	Stock Notification via Fonte API	System is not connected to the network / API access failed	An error log appears “Failed to send stock notification to sales - Fonte API connection failed”	Valid

IV. CONCLUSION

Research and development results show that the application created for Toko Atika, which includes cashier management, inventory management, and web-based sales management, can address various operational issues previously handled manually. The system automates stock recording, receipt printing, and sales and customer information management. By integrating the Fonte API, the system can facilitate smoother distribution and monitor inventory in real time by automatically notifying sellers when inventory is low. Black box testing demonstrated that all features functioned well and were valid according to the test scenarios. It is hoped that MSME owners like Toko Atika will utilize this application to increase productivity, reduce recording errors, and streamline decision-making.

REFERENCES

- [1] (2022). OKTAL: Journal of Computer Science and Science. Designing a Website-Based Cashier Application for a Grocery Store Using the Waterfall Method. *OKTAL: Journal of Computer Science and Science*, 1(08), 1274–1281. <https://journal.mediapublikasi.id/index.php/oktal/article/view/509/440>
- [2] (2023). Amikom Surakarta National Seminar (Semnasa) 2023 Design of a Web-Based Cashier and Inventory System with the Rapid Application Development Method to Improve Stock Management in Small-Scale Retail Stores.2, November, 675–684.
- [3] (2022). Design and Development of a Web-Based Sales Cashier Application for 3Manstore, a Micro, Small, and Medium Enterprise. *Algorithm Journal*, 19(2), 481–492. <https://doi.org/10.33364/algorithm/v.19-2.1117>
- [4] (2024). Android-Based Fish Sales Information System in Kefamenanu Using the Waterfall Method. *Krisnadana Journal*, 3(2), 78–89. <https://doi.org/10.58982/krisnadana.v3i2.527>
- [5] (2023). Development of Goods Inventory Management Information System at ARJ88 Distro Using Waterfall System Development Method. *JoMI: Journal of Millennial Informatics*, 1(1), 35.
- [6] (2021). Design and Construction of a Point of Sale (POS) Software Information System Using the Web-Based Waterfall Method. *JST (Journal of Science and Technology)*, 10(1), 92–103. <https://doi.org/10.23887/jstundiksha.v10i1.29748>
- [7] (2023). Analysis of Revenue Cycle Accounting Information Systems Using DFD and Flowchart in Porobico Business. 7, 1(2), 495–510. <https://doi.org/10.61930/jurbisman.v1i2.181>
- [8] (2022). Efforts to Improve the Inventory Monitoring System by Designing an Entity Relationship Diagram (ERD) as a Basis for Designing a Case Study at Ud “X.” Cylinder: *Scientific Journal of Mechanical Engineering*, 8(1), 29–35. <https://doi.org/10.25170/cylinder.v8i1.3910>.