EXPLORE
Vol. 16 No. 2, July 2026, hlm. 127-136
ISSN : 2656-615X (Media Online), ISSN : 2087-894X (Media Cetak)
DOI : https://doi.org/10.35200/exp.v16i2.1

Design of a Web-Based Café Cashier Information System with a Self-Service Concept Using Laravel

Muhammad Andre Hasan*[1], Najwa Raudatul Jannah[2], Muhammad Mahsan Asgar[3]

[1,2,3]Teknik Informatika, *Faculty of Computer Science and Artificial Intelligence, Universitas Teknologi Mataram, Mataram, Indonesia*
Email: [1]andreutmti23@gmail.com, [2]najwaraudatul2@gmail.com , [3]mmahsanasgar@gmail.com

ARTICLE HISTORY
Received: June 07, 2021
Revised: July 05, 2021
Published: July 29, 2021

Copyright © 2026 by the Authors: This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License.

ABSTRACT

The development of information technology has encouraged various business sectors, including the culinary industry, to improve service quality through the implementation of digital-based systems. One of the common problems faced by cafés is the ordering and payment process, which still relies on waiters or cashiers, potentially causing queues, extending service time, and increasing the risk of transaction recording errors. Therefore, a self-service cashier system is needed to allow customers to place orders independently, making the service process faster, more accurate, and more efficient. This study aims to design and develop a Web-Based Self-Service Cashier System using the Laravel Framework to improve service effectiveness in cafés. The software development method used is the Waterfall method, which consists of requirements analysis, system design, implementation, testing, and maintenance. The system design uses an Entity Relationship Diagram (ERD), while the implementation uses the Laravel Framework with a MySQL database. The developed system provides features for managing products, categories, inventory, customers, shopping carts, orders, payments, and transaction reports. Customers can view the menu, select products, place orders, and complete payments independently, while administrators can manage café operational data through an integrated dashboard. System testing was conducted using Black Box Testing. The results show that the system can improve the effectiveness of the ordering and transaction processes, increase data management accuracy, and assist café management in monitoring transactions and preparing sales reports
.

Keyword: cafe management; kitchen display system; laravel framework; self-service ordering.

How to Cite: [Caption completed by the editor]

1. INTRODUCTION

The development of information technology has driven the digitalization of business processes across various sectors, including the culinary sector. The use of technology in culinary business operations enables ordering, sales transactions, data management, and report generation to be carried out in a more structured and efficient manner. One form of technology implementation is a web-based cashier information system, which not only functions to record transactions but also assists in managing sales data and supports the operational needs of businesses. The implementation of web-based cashier systems has demonstrated benefits in improving the efficiency of transaction recording, data management, and sales report generation in culinary businesses. [1], [2]. This digitalization can also reduce reliance on manual record-keeping, which may lead to errors and difficulties in data management as the volume of transactions increases[3].

In café businesses, the need for information systems is becoming increasingly important because ordering and transaction activities can occur in relatively high volumes, particularly when the number of customers increases. Conventional ordering processes may require customers to wait for service and increase the possibility of errors in recording orders [4], [5]. developed a website-based cashier system utilizing QR Codes to help accelerate the ordering process in cafés. The use of QR Codes allows customers to access the menu and place orders more independently, thereby helping to improve service effectiveness. Problems related to manual record-keeping were also identified in

previous research [5] which developed a web-based cashier system to assist with sales calculations, transaction recording, and report generation at DailyFood Kitchen. In line with this research, [2] demonstrated that the digitalization of transaction processes can address problems arising from the use of paper receipts and manual report recording.

The development of information systems in culinary businesses is no longer limited to cashier digitalization but has also expanded to the implementation of self-service ordering concepts. This concept allows customers to access menus and place orders directly through the system without being fully dependent on waiters. The use of QR Codes has become one of the commonly used mechanisms to support this process. [6] demonstrated the use of QR Codes to accelerate the ordering process, while [7], [8] demonstrated that the use of digital ordering technology can provide convenience for customers while helping to reduce errors in the ordering process. The study also developed a sales information system for Terash Café with QR Code-based ordering integration to reduce human error and improve service efficiency. These findings indicate that the implementation of self-service is relevant to supporting faster and more structured café service processes.

In addition to the ordering mechanism, advances in system development technologies also play an important role in building café information systems that can meet operational needs in an integrated manner. Several studies have implemented the Laravel framework and the Model-View-Controller (MVC) architecture in developing web-based cashier and Point of Sale systems [9], [10] demonstrated that the use of structured frameworks and architectures can support system management and code maintainability. Laravel was also implemented in the Natta Coffee cashier system to support transaction, inventory, and report management. [11], as well as in the development of the Point of Sale system for Café Calandra. Meanwhile, [7] developed a web-based Point of Sale system to assist in managing increasingly complex transactions. The results of the study indicate that the use of web frameworks and structured databases can support the development of systems with multiple operational modules.

Although previous studies have extensively discussed web-based cashier systems, QR Code-based ordering, inventory management, Point of Sale systems, and the implementation of Laravel in culinary businesses, the focus and level of system integration developed in these studies vary. Some studies focus on transaction recording and reporting, [3], [12] while other studies focus on QR Code-based ordering [1], [8], [13] or the development of Point of Sale systems and the implementation of Laravel [11]. This condition indicates an opportunity to develop a system that integrates customer self-service ordering with more comprehensive café operational management within a single system.

Based on this research gap, this study develops a web-based café cashier information system by implementing a self-service concept. This concept allows customers to place orders independently through the system interface, while café managers can centrally manage various operational activities. The developed system includes user management, product management, category management, product inventory, inventory history, customer management, shopping carts, orders, order details, payments, payment methods, and table management. Through this integration, the system functions not only as a transaction recording tool but also as a system that connects the customer ordering process with café operational management. This study uses the Laravel Framework with a MySQL database and implements the Model-View-Controller (MVC) architecture to produce a structured and easily maintainable system.

The system development process uses the Waterfall method, which consists of the stages of requirements analysis, system design, implementation, testing, and maintenance. This method was chosen because the system requirements can be defined from the beginning, allowing the development process to be carried out gradually and systematically[1]. Through this approach, the study aims to produce a café cashier system that integrates customer self-service ordering with the management of products, inventory, customers, tables, orders, payments, and reports.

Based on the above discussion, the objectives of this study are to design and implement a web-based café cashier information system with a self-service concept using Laravel and MySQL, integrate the customer ordering process with café operational management, and apply the Waterfall method in the system development process. The resulting system is expected to enable customers to place orders more independently and assist café managers in managing transactions and operational data in a more structured, accurate, and efficient manner.

2. RESEARCH METHODS

This study adopts a step-by-step approach based on the Waterfall model to develop a web-based Self-Service Cashier System integrated with a self-service ordering module in the context of café operations. The main stages consist of requirements analysis, system design, implementation, testing, and evaluation and refinement. These stages are summarized in the research flow diagram (Figure 1) and carried out sequentially to ensure consistency between the system design and implementation and the users' requirements [9], [13].

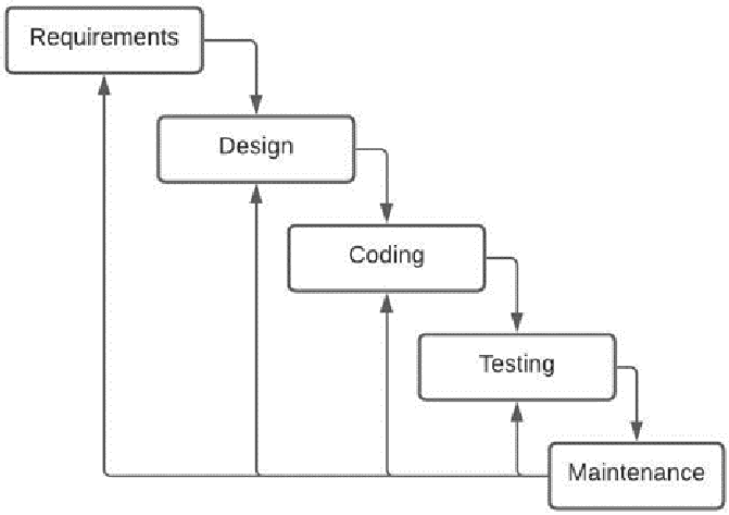


Figure 1. Waterfall Method

a. Requirements Analysis
The requirements analysis stage aims to identify user and system requirements based on the formulated problems. The requirements are divided into two main actors: customers and café managers. Customers require the ability to view menus, select products, add items to the shopping cart, place orders, and make payments. Café managers require functionalities for managing products, categories, inventory, customers, tables, orders, payments, and transactions. This analysis is consistent with findings that transaction digitalization encompasses not only payment processing but also transaction recording and reporting integrated with table and inventory management[12].

b. System Design
The system design is based on the results of the requirements analysis, focusing on the design of business processes, use case diagrams, activity diagrams, class diagrams, ERD, database structure, system architecture, and user interfaces. This design provides a framework for implementation to ensure that the developed functions are consistent with the requirements of both customers and café managers. The database design includes key components such as users, products, categories, product inventory, inventory history, customers, shopping carts, orders, order details, payments, payment methods, and tables, enabling real-time and secure data integration across modules[1].

c. Implementation
The implementation stage translates the system design into a functional system. The system is developed using the Laravel Framework with MySQL as the database, following the MVC approach to separate business logic, presentation, and data. The use of Laravel in café self-service contexts has been documented in previous studies, demonstrating benefits in terms of security, efficiency, and code maintainability, particularly in research involving Natta Coffee and Café Calandra [9]. The implementation focuses on the self-service concept, allowing customers to place orders independently through a web-based interface, while café managers have full control over integrated operational modules.

d. Testing
Testing is conducted after the implementation stage using Black Box Testing to ensure that each function operates according to the specified requirements. The testing scope includes login, product management, category management, inventory, customer management, table management, shopping cart, ordering, payment, and transaction functions. Black Box Testing is selected because it evaluates system functionality without requiring knowledge of the internal system structure, focusing on whether the system produces the expected outputs based on user inputs [12]

e. Evaluation and Refinement
The testing results are evaluated to identify deviations from the specified requirements. If any discrepancies are found, system improvements and refinements are carried out. The ultimate goal is to produce a web-based Self-Service Cashier System that has undergone structured requirements analysis, design, implementation, and testing, thereby supporting café operations in a more structured, accurate, and responsive manner in addressing fluctuations in customer demand[1].

3. RESULTS AND DISCUSSION

a. System Analysis

The analysis stage is conducted to determine the system requirements that support the ordering and transaction processes in the café. Based on these requirements, the system is developed with two main user groups: customers and system administrators. Customers use the system to place orders independently, while administrators manage product data, inventory, orders, payments, and system users.

b. System Design

he results of the analysis are then translated into a database design. The Caffe Cashier database consists of tables for categories, products, customers, tables, shopping carts, orders, order details, payment methods, payments, product inventory, inventory history, and users. This structure demonstrates that the database design covers the main requirements of the self-service cashier process.

Table 1. Database Design Results

No.	Komponen	Hasil Perancangan	Keterkaitan dengan Sistem
1	Category	Category table	Product classification
2	Product	Product table	Stores menu/product data
3	Customer	Customer table	Stores customer information
4	Table	Table table	Table identification and QR Code
5	Shopping Cart	Shopping cart table	Stores customers' selected products
6	Order	Order table	Stores ordering transactions
7	Order Details	detail_pesanan table	Stores product details within an order
8	Payment	Payment table	Stores payment transactions
9	Payment Method	metode_pembayaran table	Determines the payment method
10	Inventory	stok_produk table	Stores inventory quantities
11	Inventory History	riwayat_stok table	Records incoming and outgoing inventory
12	User	users table	Manages administrator and cashier access

The design structure serves not only as a medium for storing transaction data but also establishes logical relationships among business processes. The product entity is connected to the category entity for menu classification and linked to stok_produk and riwayat_stok to ensure inventory accuracy. In the transaction flow, the shopping cart entity connects customers with their selected products. Once an order is confirmed, the order entity is directly related to the customer and table entities, while the details of each item are recorded in detail_pesanan, which references the corresponding product. Subsequently, the payment entity validates the transaction by linking the order to the payment method used.

The overall relationships among the entities are modeled in an Entity Relationship Diagram (ERD), as shown in the figure below.

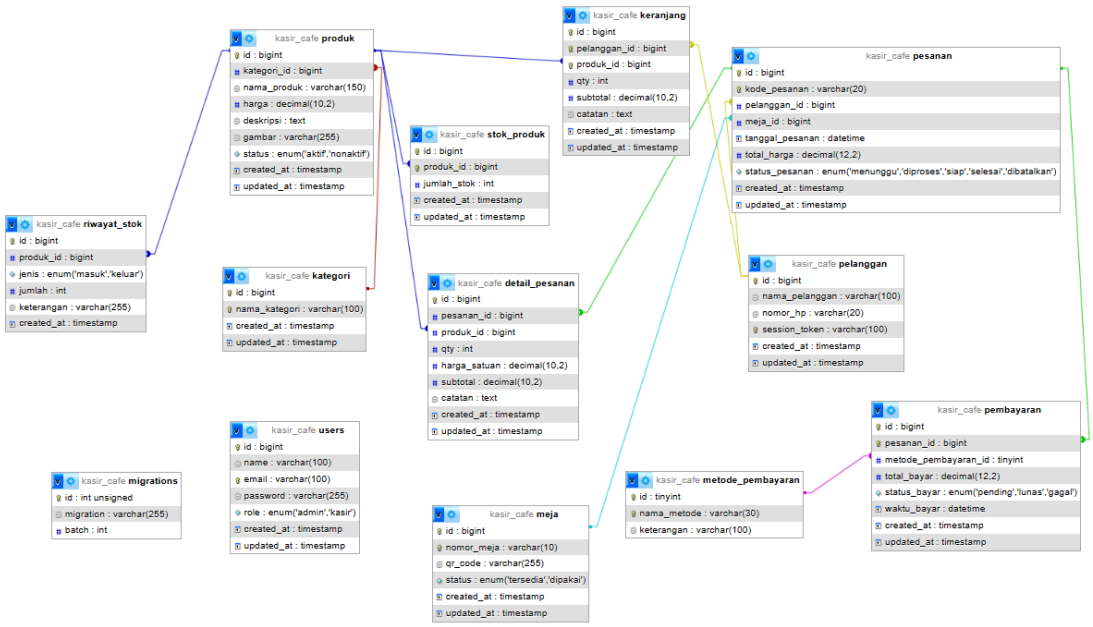


Figure 2. Entity Relationship Diagram (ERD)

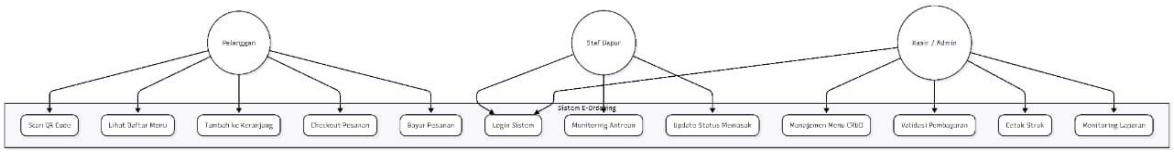


Figure 3. Use Case Diagram

The self-service system integrates the restaurant's entire operational process into a centralized system by connecting three main roles: Customer, Kitchen Staff, and Cashier/Administrator. The transaction process begins with the Customer, who scans the QR Code provided at the dining table to access the system without having to wait in a manual queue. Through the interface, customers can view the available menu, select their desired dishes and add them to the shopping cart, proceed to checkout, and complete the payment digitally.

Once an order is submitted, the data is automatically forwarded to the Kitchen Staff interface, where staff members log in to the system. Kitchen staff are responsible for monitoring incoming orders in real time and updating the cooking status at each stage until the order is fully prepared. On the management side, the Cashier/Administrator has full access to authenticate into the system, manage food master data through CRUD-based menu management, such as adding, editing, or deleting menu items and prices, validate customer payments, print physical transaction receipts, and monitor integrated daily sales reports to evaluate the restaurant's business performance.

f. Implementation

The coding and implementation stage is carried out based on the requirements analysis and system design developed in the previous stages. The system is implemented as a web-based application, with the primary focus on implementing a self-service ordering mechanism for customers and operational management for café administrators.

1. User Interface

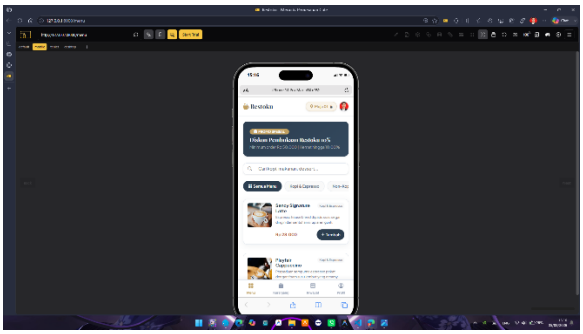


Figure 4. Homepage

Homepage & Menu: Displays a visual catalog of food and beverages grouped by category, complete with product images, prices, ingredient descriptions, and an "Add to Cart" action button.

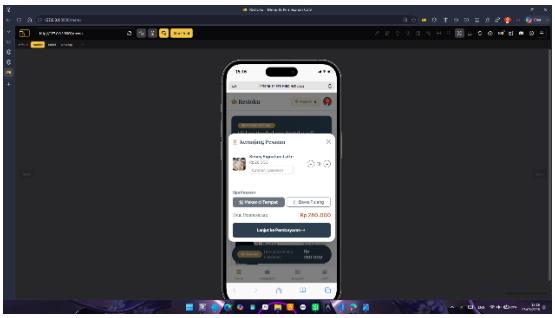


Figure 5. Checkout Page

Cart & Checkout Page: Displays an order summary, subtotal, a special request field (such as spice level or allergy information), and the automatically detected table number.

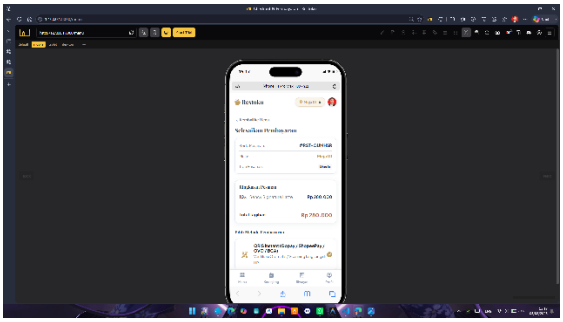


Figure 6. Payment Page

Payment Page: Displays the final bill amount with integration of various digital payment methods (QRIS, e-wallets, or virtual accounts), along with real-time payment verification status.

2. Administrator Dashboard

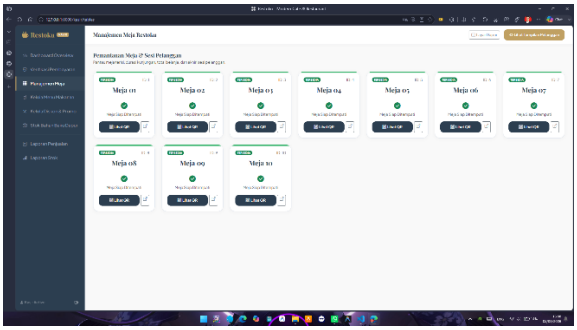


Figure 7. Table Management Page

Restoku Table Management: Displays a visual representation of the restaurant's table layout in the form of interactive cards. Each table card (Table 01 to Table 10) is equipped with an availability status (shown in green with the label "AVAILABLE") and a table ID.

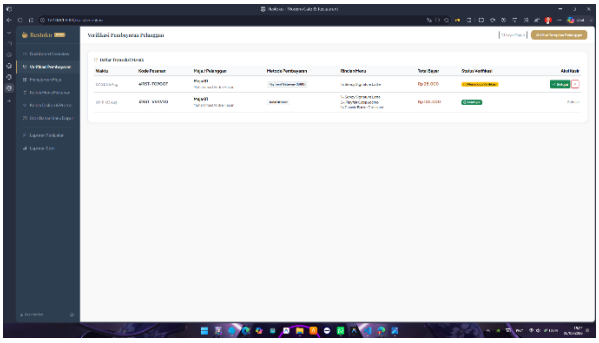


Figure 8. Payment Verification Page

The Customer Payment Verification Page functions as a control center for ensuring that every incoming transaction is valid and has been properly verified.

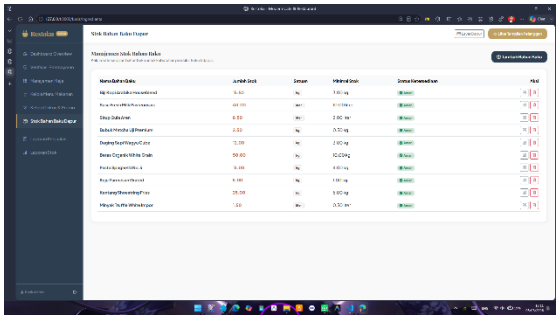


Figure 9. Raw Material Inventory Page

The Kitchen Raw Material Inventory Management Page is an important module for managing the availability of raw materials in the kitchen. The table records the name of each raw material, the current stock quantity

(in kilograms or liters), the minimum stock threshold, and the availability status (safe or warning when stock levels are low).

3. *Kitchen Display System Interface*

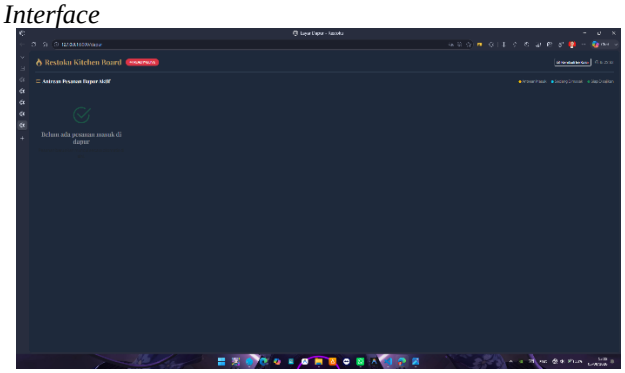


Figure 10. Kitchen Board Page

The initial page of the Restoku Kitchen Board is designed with a dark mode interface for comfortable use in the kitchen environment. In the upper-left corner, there is a “REALTIME LIVE” indicator, showing that the system is directly connected and automatically updates data without requiring a manual refresh

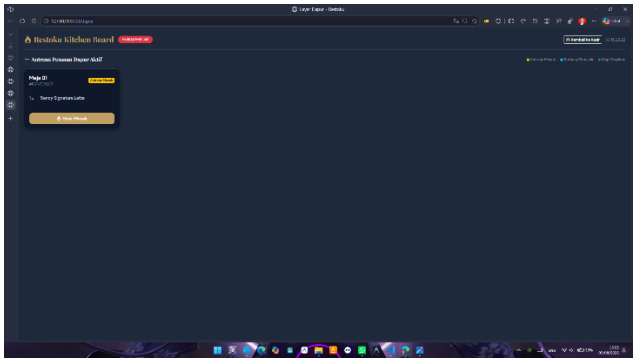


Figure 11. Incoming Orders Queue Page

The Incoming Orders Queue Page displays the initial interface of the Restoku Kitchen Board. It is designed with a dark mode interface to provide a comfortable experience in the kitchen environment. In the upper-left corner, there is a “REALTIME LIVE” indicator, showing that the system is directly connected and automatically updates data without requiring a manual refresh

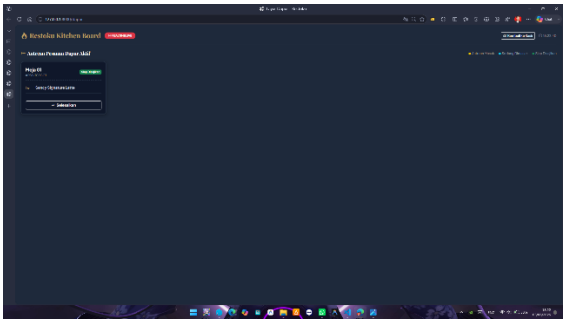


Figure 12. Completed Orders Page

The page displays the final stage of the kitchen workflow. After the chef presses the “Ready to Serve” button, the order card status is updated with a green “Ready to Serve” label. This indicates that the cooking process has been completed and the food is ready to be delivered to the customer’s table. The action button on the card then changes to “Complete” (white with a border). When this button is pressed, the order is marked as completed, and the card disappears from the kitchen display, indicating that the workflow for the order has been fully completed.

c. Testing

The testing stage was conducted to ensure that the implemented system operates according to the specified requirements and design. The testing method used was Black Box Testing, which focuses on system functionality based on inputs and outputs without directly examining the program code. The testing covered login, product data

management, categories, inventory, tables, orders, customers, and payments, as well as the ordering, payment, and report-generation processes. Each function was tested using predetermined scenarios to ensure that the system produces outputs that meet user requirements.

Table 2. System Testing

ID Pengujian	Aktor	Skenario Uji	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-01	Customer	Scan QR Code	Scan the QR code displayed on the dining table using a mobile phone camera.	The system directly redirects to the menu catalog page with a valid table number parameter.	Passed
TC-02	Customer	View Menu List	Browse the menu page and select a product category.	The system displays a complete list of food and beverages, including photos, names, descriptions, prices, and availability status.	Passed
TC-03	Customer	Add to Cart	Select a menu item, specify the quantity/additional notes, and press the cart button.	The item is successfully added to the temporary shopping cart and the order total is updated.	Passed
TC-04	Customer	Order Checkout	Open the cart, review the order details, and press the checkout button.	The order summary and total price are accurately displayed before proceeding to payment.	Passed
TC-05	Customer	Order Payment	Select a payment method, such as QRIS/E-Wallet, and complete the transaction.	The payment is successfully processed and the system displays the transaction status along with the ticket/queue number.	Passed
TC-06	Kitchen Staff	System Login	Enter the kitchen staff username and password on the authentication page.	The user is successfully redirected to the kitchen staff queue dashboard (Kitchen Order Display).	Passed
TC-07	Kitchen Staff	Queue Monitoring	Monitor the queue page when a new order is received from a customer.	The new order appears in real time according to the order sequence (FIFO).	Passed
TC-08	Kitchen Staff	Update Cooking Status	Change the order ticket status from Processing to Cooking and then to Completed.	The order status is immediately updated in the cashier system and customer notification display.	Passed
TC-09	Cashier / Admin	System Login	Enter the cashier/admin account credentials.	The user is successfully redirected to the main restaurant management dashboard (Back-Office).	Passed
TC-10	Cashier / Admin	Menu Management (CRUD)	Add, edit price/stock information, or delete a menu item.	The menu master data is successfully updated and the changes are immediately reflected in the customer catalog.	Passed
TC-11	Cashier / Admin	Payment Validation	Check and verify the customer's payment status (cash/digital).	The order payment status changes to Paid in the transaction database.	Passed
TC-12	Cashier / Admin	Receipt Printing	Press the print button on the details of a validated transaction.	The cashier thermal printer prints the transaction receipt in the correct format.	Passed

TC-13	Cashier / Admin	Report Monitoring	Select a date range to view the restaurant's sales and revenue summary.	The system displays daily/monthly sales charts and provides options to download the report in PDF/Excel format.	Passed

4. CONCLUSION

Based on the results of the research entitled "Design and Development of a Web-Based Self-Service Cashier System Using the Laravel Framework," it can be concluded that the research successfully developed a web-based self-service cashier system using the Laravel Framework and the kasir_cafe database.

The developed system covers the main cashier activities in a café, including category and product management, customer and table management, product selection through a shopping cart, ordering processes, order details, payment processing, inventory management, and inventory history. The database structure also supports relationships among product, customer, table, order, order detail, payment, and inventory data, allowing transaction processes to be stored in an integrated manner.

The implementation of the self-service concept enables customers to independently select products and place orders through a web-based system. Meanwhile, café managers can manage product, order, payment, and inventory-related data through the system. The development using the Waterfall method provided a structured process covering requirements analysis, system design, implementation, and system testing.

Based on the testing results, the main functions of the system operated according to the predetermined requirements. Therefore, the developed system has achieved the primary research outcome, namely a Web-Based Self-Service Cashier System Using the Laravel Framework.

REFERENCES

[1] Y. Handoko Agustin, A. Latifah, and A. F. Nugraha, "Perancangan Sistem Informasi Aplikasi Kasir pada Kafe Restorasi Kopi Berbasis Web," *Jurnal Algoritma*, vol. 18, no. 1, pp. 302–312, Nov. 2021, doi: 10.33364/algoritma/v.18-1.947.

[2] Z. Rashifah and E. S. Budi, "Rancangan Sistem Informasi Pada Kasir Berbasis Web," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 3, no. 4, p. 529, Jun. 2022, doi: 10.30865/json.v3i4.4241.

[3] R. Kaban and R. J. Nasution, "Penerapan Metode Rapid Application Development (RAD) dalam Perancangan Sistem Pemesanan Menu menggunakan Quick Response (QR) Code," *MEANS (Media Informasi Analisa dan Sistem)*, pp. 144–152, Dec. 2020, doi: 10.54367/means.v5i2.920.

[4] R. R. Agustin, "Geo-Location Clustering Untuk Menentukan Tempat Tujuan Wisata Dengan Menggunakan Algoritma PAM (Partitioning Around Medoids)," *Journal of Informatics and Communication Technology (JICT)*, vol. 4, no. 2, pp. 1–9, Jan. 2023, doi: 10.52661/j_ict.v4i2.115.

[5] B. H. Pomo, S. Moreta, and D. E. Pranoto, "Perancangan Aplikasi Sistem Informasi Kasir Berbasis Web Pada DailyFood Kitchen."

[6] A. D. Prastya and N. A. Indartono, "Implementasi Penggunaan Kode QR Pada Pemesanan Menu di Cafe Tamulang Banyuwangi," *Jikom: Jurnal Informatika dan Komputer*, vol. 12, no. 2, pp. 52–61, Feb. 2023, doi: 10.55794/jikom.v12i2.87.

[7] A. E. Widodo, F. F. Wati, and A. Widayanto, "Sistem Informasi Pemesanan Menu Kafe Menggunakan QR-Code," *J-INTECH*, vol. 11, no. 1, pp. 102–110, Jun. 2023, doi: 10.32664/j-intech.v11i1.865.

[8] Budi Hartono and Danang Danang, "Sistem Pemesanan dan Pembayaran Menggunakan Teknologi Quick Response Code (QR Code) Berbasis Web pada Kedai Cangkir Gubug," *Jurnal Manajemen Informatika & Teknologi*, vol. 1, no. 2, pp. 62–81, Oct. 2021, doi: 10.51903/mifortekh.v1i2.34.

[9] G. H. Rea, W. J., Pan, Y., Fenyves, E. J., Sujisawa, I., Suyama, H., Samadi, N., & Ross, "Electromagnetic Field Sensitivity," *Journal of Bioelectricity*, pp. 241–256, 1991, doi: https://doi.org/10.3109/15368379109031410.

[10] R. H. Rusmara, "PENGEMBANGAN SISTEM KASIR BERBASIS WEB DI NATTA COFFEE: STUDI IMPLEMENTASI FRAMEWORK LARAVEL," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, Apr. 2025, doi: 10.23960/jitet.v13i2.6414.

[11] M. A. T. Taufiqurahman, N. Halizah, M. M. K. Abadi, I. F. Adibah, and A. S. Fitri, "PERANCANGAN SISTEM CT-POS BERBASIS WEB UNTUK MENINGKATKAN EFISIENSI PROSES BISNIS PADA MINIMARKET 'CAFE TUMAN,'" *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 2, Apr. 2023, doi: 10.23960/jitet.v11i2.2886.

[12] E. Wijaya and S. Hermawan, “APLIKASI PELAYANAN PEMESANAN UNTUK PELANGGAN BERBASIS MOBILE ANDROID PADA EDUPLEX CAFE,” *Komputa : Jurnal Ilmiah Komputer dan Informatika*, vol. 6, no. 2, pp. 73–82, Oct. 2017, doi: 10.34010/komputa.v6i2.2480.

.