

A Data-Driven POS System for Sales Monitoring and Inventory Management

Raven Rich Mathew C. Mamar, Christine Faith T. Dairo, Erwin C. Maravilla, Jeffric S. Pisuena, Kristine T. Soberano

Graduate School, State University of Northern Negros, Philippines

Received: 10.04.2026 | Accepted: 26.05.2026 | Published: 27.05.2026

*Corresponding Author: Raven Rich Mathew C. Mamar

DOI: [10.5281/zenodo.20412208](https://doi.org/10.5281/zenodo.20412208)

Abstract

Original Research Article

Point-of-Sale (POS) systems have become central to retail business management, yet small enterprises in developing economies continue to rely on error-prone manual methods. This study presents the design and development of a data-driven POS system tailored for small-scale retail operations. Built using PHP and MySQL under a System Development Life Cycle (SDLC) framework, the system integrates user authentication, sales transaction processing, real-time inventory tracking, and exportable report generation into a single locally deployable platform. Functional and usability evaluation confirmed that the system reduces manual recording errors, improves transaction processing speed, and provides real-time visibility into sales and stock levels. Role-based access control ensures data security, while cross-browser compatibility supports diverse deployment environments. The system meets the operational requirements of small retail businesses and establishes a foundation for future enhancements including cloud integration, predictive analytics, and mobile application support.

Keywords: Point-of-Sale System, Inventory Management, Sales Monitoring, PHP, MySQL, Data-Driven Systems.

Copyright © 2026 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

I. Introduction

Point-of-Sale (POS) systems have become indispensable tools for modern retail management, enabling businesses to record transactions, monitor sales performance, and maintain accurate inventory records within a single operational platform. As retail commerce has grown in complexity and volume, the demand for systems that can provide real-time visibility into business operations has intensified. Despite these widely recognized benefits, a substantial portion of small and medium-sized enterprises (SMEs) in developing economies—

including many in the Philippines—continues to depend on manual, paper-based methods that are inherently vulnerable to human error, data loss, and operational delays [1].

The consequences of manual retail management extend beyond clerical inconvenience. Inaccurate inventory records lead to recurring stockouts or overstock situations, both of which directly impair profitability and customer satisfaction. Sales data captured manually are difficult to aggregate and analyze at scale, limiting a business owner's ability to identify demand trends,



Citation: Mamar, R. R. M. C., Dairo, C. F. T., Maravilla, E. C., Pisuena, J. S., & Soberano, K. T. (2026). A data-driven POS system for sales monitoring and inventory management. *GAS Journal of Engineering and Technology (GASJET)*, 3(5), 161-171.

evaluate product performance, or make evidence-based procurement decisions [3]. The absence of automated transaction recording also creates reconciliation challenges at the close of each business day, contributing to financial reporting errors that compound over time. Mattegunta [1] documented these inefficiencies in a study of real-time inventory integration, confirming that delayed or fragmented data pipelines remain a primary constraint on operational responsiveness in small retail businesses.

Research interest in web-based and data-driven POS systems has grown considerably in response to these challenges. Ramadhani, Nindyasari, and Murti [2] demonstrated the effectiveness of a web-based POS platform for small-scale retail management, reporting measurable improvements in transaction processing speed and record accuracy compared to baseline manual methods. Their findings reinforce the case for adopting structured software-based solutions even in resource-constrained settings where enterprise-grade commercial systems may be financially inaccessible. In a related line of work, Susanto and Fenriana [3] integrated Apriori-based market basket analysis within a POS framework, illustrating that data captured at the point of sale can support not only operational efficiency but also customer behavior analysis and strategic decision support. This convergence of transactional and analytical capability within a single system represents a significant advancement over traditional cash-register-based approaches.

Broader perspectives on POS system intelligence have further expanded the scope of this research domain. Sang, Ravikumar, Aarthi, and Ben [4] examined how artificial intelligence (AI) and business intelligence (BI) tools embedded in POS environments can optimize sales forecasting and inventory replenishment decisions, reducing both surplus stock and service failures. At the food service and hospitality sector level, Velasco, Domínguez, Gamón, Macías, and Torres [5] proposed a POS-based digitalization strategy for small restaurants, demonstrating the system's adaptability across diverse retail and service contexts. More recently,

Yulia and Seputro [6] evaluated the impact of a FIFO-based POS implementation on warehouse operations, establishing a direct link between structured POS data management and improved inventory turnover in distribution environments.

A persistent gap across this body of work, however, is the limited availability of POS solutions specifically designed for offline-capable, locally hosted deployment in small retail environments with constrained IT infrastructure. Many commercial POS products require continuous cloud connectivity or carry licensing costs that are prohibitive for micro and small enterprises in developing regions. Open-source and locally deployable alternatives built on PHP and MySQL have been proposed in the literature as viable responses to these constraints, yet few studies have rigorously documented the functional and usability performance of such systems against the specific operational requirements of small Philippine retail businesses [2], [3].

This study addresses that gap by presenting the design, development, and evaluation of a data-driven POS system built using PHP and MySQL under an SDLC framework. The system integrates user authentication, sales transaction processing, real-time inventory tracking, and exportable report generation into a single locally deployable platform.

II. Objectives of the Study

The primary objective of this study is to design and develop a data-driven Point-of-Sale (POS) system that enhances sales monitoring and inventory management for retail businesses.

Specifically, this study aimed to:

1. Develop a POS system that automates sales transactions and reduces manual errors in recording data.
2. Implement an efficient inventory management feature that enables real-time tracking of product stock levels.
3. Design a secure user authentication system for controlled access to the platform.

4. Create a centralized database that ensures data consistency, organization, and easy retrieval of records.
5. Generate accurate and exportable sales reports that support business analysis and decision-making.
6. Improve overall operational efficiency by integrating sales and inventory processes into a single system.
7. Provide a user-friendly interface that allows easy navigation and usage for business owners and staff.

III. Methods

A. Data Source and Collection

Data for system testing and evaluation was gathered through simulated retail transactions designed to replicate realistic daily business operations. The simulation dataset comprised product records including item names, categories, stock-keeping units (SKUs), unit prices, and opening inventory quantities. Simulated customer purchases generated transaction records capturing date and time stamps, product codes, quantities sold, amounts tendered, and payment methods. This controlled environment allowed the research team to examine a variety of transaction scenarios—including bulk purchases, percentage discounts, and return processing—without exposure to actual business data or risk to live operations.

System-generated simulation data was supplemented by structured user interaction during the testing phase. Participants executed operational tasks including product registration, inventory adjustment, sale processing, and report generation. The team observed task execution to assess system responsiveness, input accuracy, and user navigation behavior under realistic conditions. All discrepancies between expected system output and observed results were logged for root-cause analysis and remediation.

Edge case testing covered three targeted scenarios: invalid or incomplete data entry, below-

minimum stock conditions triggering reorder alerts, and simulated mid-transaction interruptions. Together, this combination of structured simulation and observed user interaction provided a comprehensive basis for evaluating both the functional performance of the system and the usability experience of its intended operators.

B. System Development Process

The system was developed following the System Development Life Cycle (SDLC) methodology, which structured the project into sequential phases of planning, requirements analysis, system design, development, testing, and implementation. The SDLC framework ensured that functional requirements were explicitly documented before development commenced and that each completed phase was validated prior to proceeding to the next.

The platform was implemented as a web-based application using PHP as the server-side scripting language and MySQL as the relational database management system. This technology stack was selected for its open-source availability, extensive community documentation, and compatibility with locally hosted server environments such as XAMPP and WAMP—configurations well-suited to small businesses without dedicated IT infrastructure or reliable cloud connectivity. Frontend components were developed using HTML, CSS, and JavaScript to deliver a responsive, browser-accessible interface.

During the design phase, system architecture and user interaction flows were modeled using data flow diagrams and use case representations to establish clear interaction paths between user roles and system components. The platform was organized around two primary user roles—administrator and cashier/staff—each with distinct access privileges. The administrator role encompasses full system control including product catalog management, user account administration, and full report access. The cashier/staff role is scoped to transaction processing, inventory queries, and receipt generation, protecting

sensitive configuration and financial summary data from unauthorized access or modification.

Figure 1 illustrates the overall three-tier architecture of the system. Client requests originate

from the browser-based interface, are processed by the PHP application server, and are persisted to or retrieved from the MySQL database layer, with responses rendered back to the user in real time.

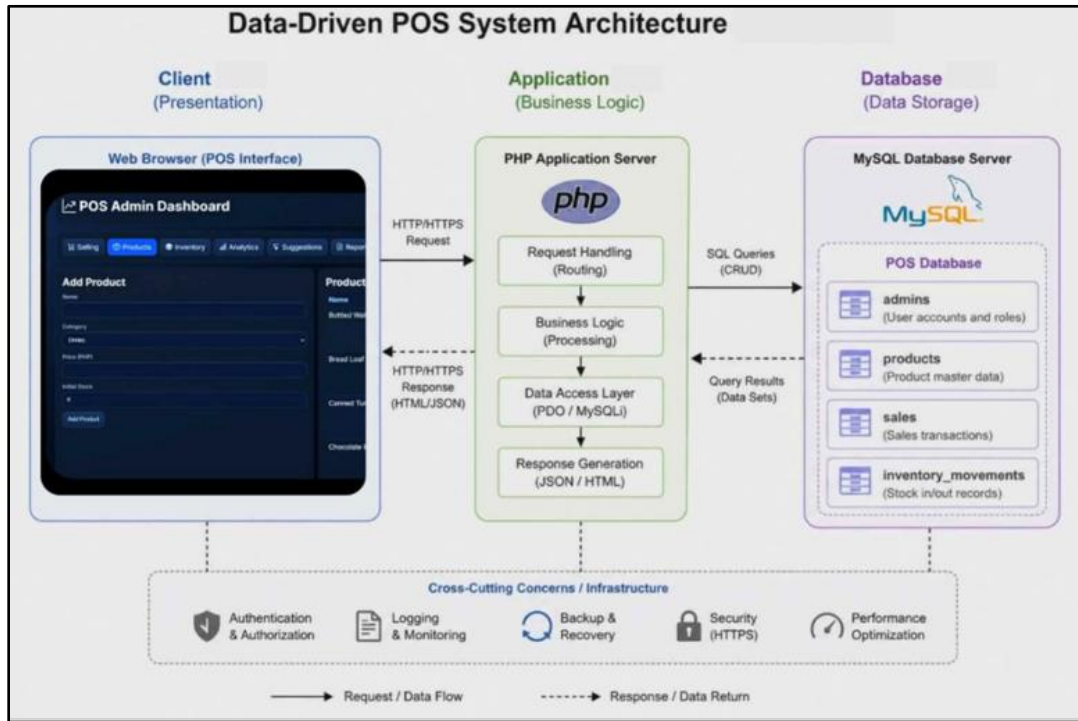


Fig. 1. System architecture of the data-driven POS system.

C. System Features and Functional Components

The developed POS system comprises eight integrated functional modules:

- 1) **User Authentication and Access Control** — Secure login and logout functionality enforces role-based access rights, restricting platform entry to verified users. Administrators retain full system access while cashier/staff roles are limited to transaction-related functions, protecting sensitive records from unauthorized disclosure or alteration.
- 2) **Sales Transaction Management** — The system automates the checkout workflow by capturing product identifiers, quantities, applicable discounts,

and payment methods. Transaction totals are computed automatically, and digital receipts are generated upon completion, eliminating manual arithmetic and reducing point-of-sale errors.

- 3) **Product and Inventory Management** — Authorized users can create, update, and deactivate product records containing item names, categories, unit prices, and stock quantities. All modifications are committed to the central database immediately and are reflected across all active user sessions.
- 4) **Real-Time Inventory Tracking** — Inventory levels are decremented automatically upon each completed sale and incremented upon restocking entries. This continuous synchronization between

transaction records and stock data supports accurate demand monitoring and timely reorder decision-making.

5) Reporting and Analytics — The system generates daily sales summaries, transaction histories, and current inventory status reports. These structured outputs support operational oversight, trend analysis, and strategic planning by presenting organized, up-to-date business data.

6) Report Export Functionality — Generated reports can be exported to PDF or Excel formats, facilitating distribution, printing, and integration into external business analysis workflows.

7) Centralized Database Management — MySQL serves as the authoritative repository for all

transaction records, product data, and user accounts. The relational schema enforces referential integrity, supports efficient query performance, and maintains data consistency as record volume grows.

8) User-Friendly Interface — The interface was designed for non-technical users, presenting structured input forms, contextual field labels, and logical navigation flows that minimize training requirements for cashiers and administrators.

The relational structure of the database is presented in Figure 2. The schema defines five primary entities—users, products, categories, transactions, and transaction_items—linked through foreign key constraints that enforce referential integrity across all operational records.

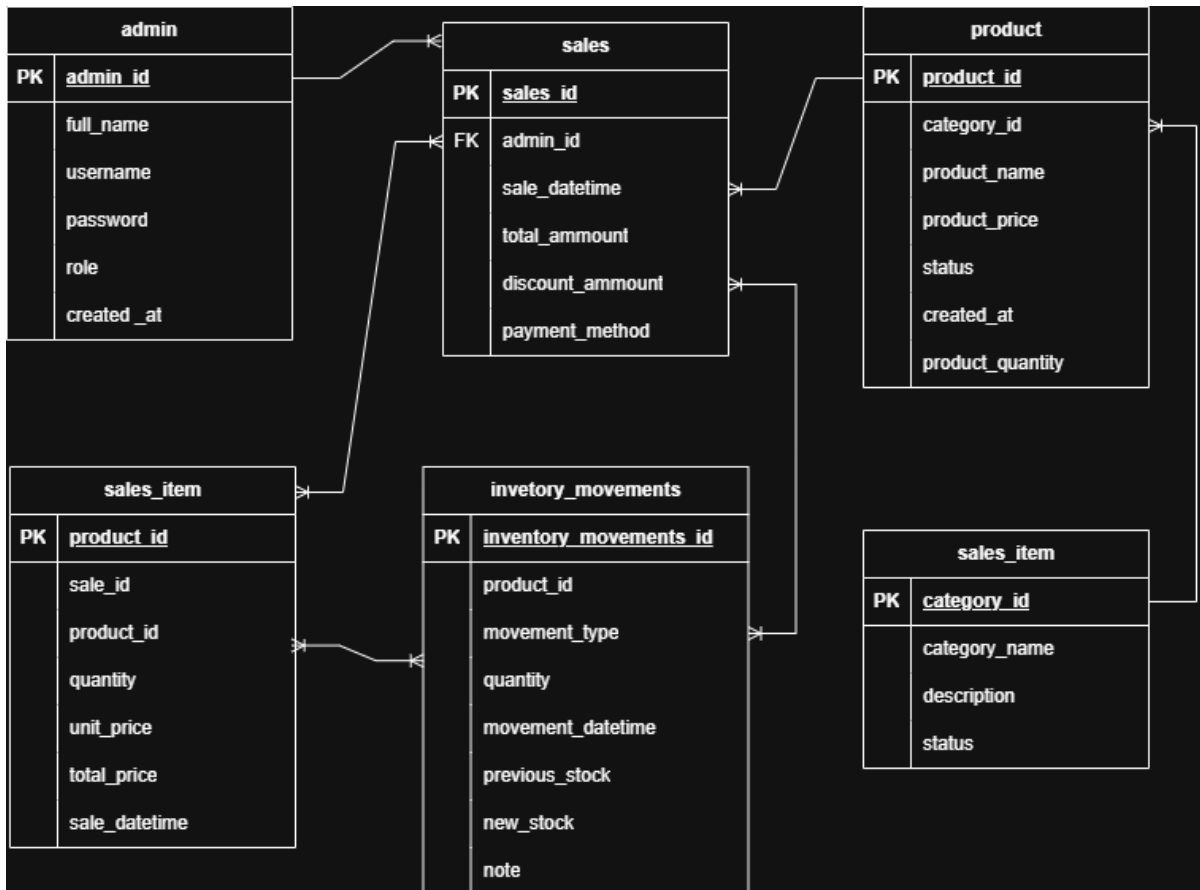
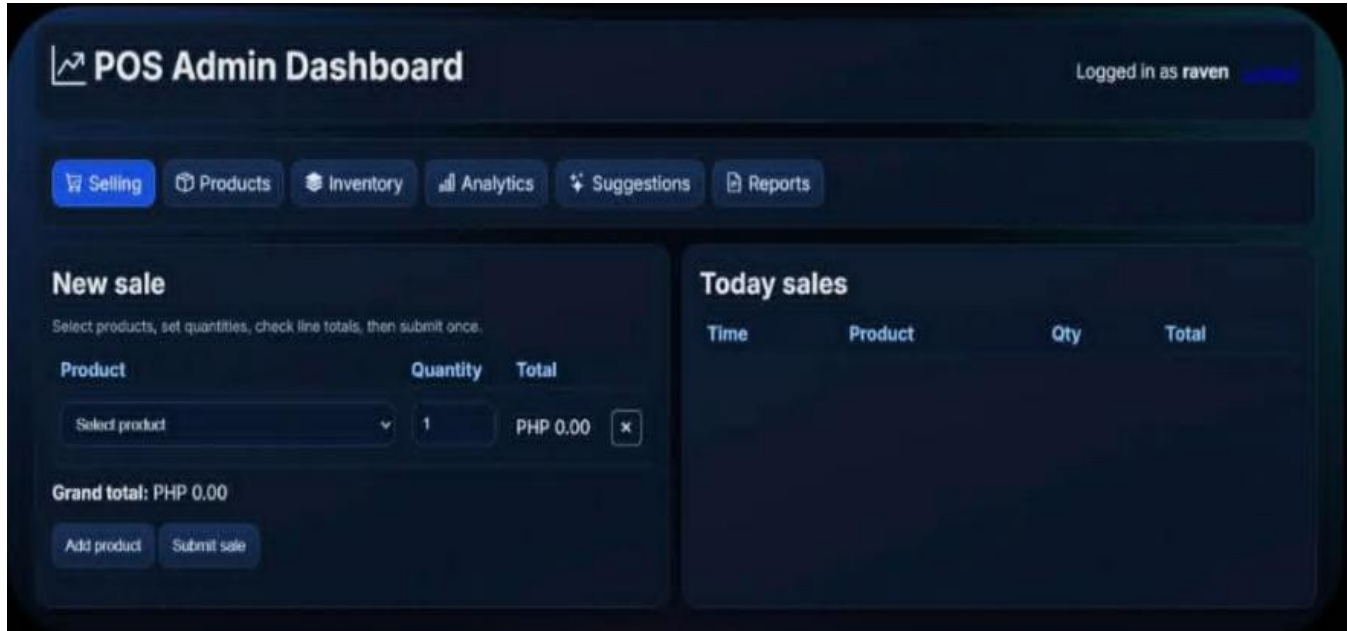


Fig. 2. Entity-relationship diagram of the POS system database schema.

Figure 3 presents screenshots of the primary system interfaces, including the sales transaction screen, the inventory management panel, and the generated sales report view.

Sales Transaction Page



Inventory Management Panel



Generated Sales Report

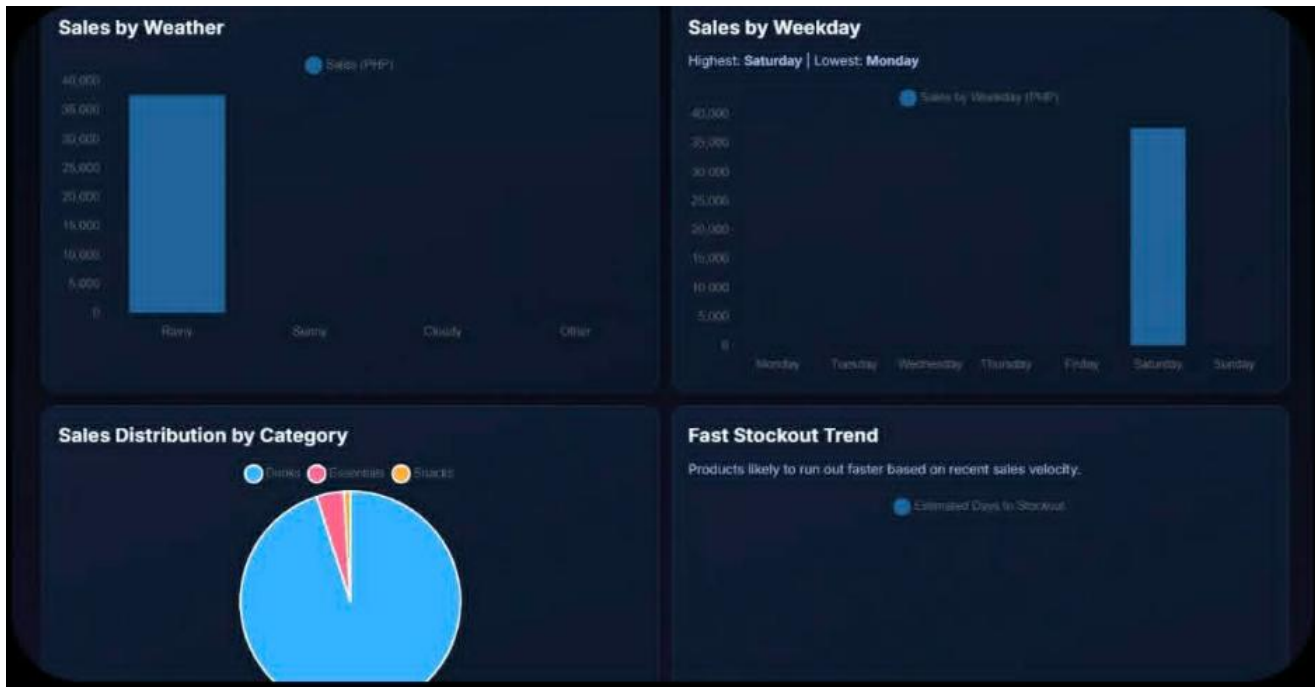


Fig. 3. Representative screenshots of the POS system interface: (a) sales transaction screen, (b) inventory management panel, and (c) generated sales report.

Hardware and Software Requirements

Minimum hardware requirements for system deployment are a computer with at least 4 GB RAM, a dual-core processor, and 500 MB of available storage. Software dependencies include a modern web browser (Google Chrome, Mozilla Firefox, or Microsoft Edge), a web server (Apache or Nginx), PHP as the backend scripting language, MySQL as the database management system, and standard frontend technologies (HTML, CSS, JavaScript). The system is compatible with Windows, Linux, and macOS operating environments.

D. System Testing

Functional testing was applied to each system module under both standard operating conditions and boundary-condition scenarios. Test cases were

designed to verify the correctness of sales processing logic, inventory update mechanisms, authentication workflows, and report generation outputs. All defects identified during testing were logged, corrected, and retested to confirm that the final build satisfied the functional requirements established during the analysis phase.

E. System Evaluation

System evaluation was conducted in two stages. In the first stage, test users executed a structured set of operational tasks—logging in, processing sales, adjusting inventory, and generating reports—while the research team observed task completion, recorded errors, and noted usability issues through direct observation. In the second stage, users rated the system using a structured evaluation instrument.

The evaluation instrument comprised a 22-item questionnaire adapted from ISO/IEC 25010 software quality standards, organized across five quality dimensions: Functional Suitability, Usability, Security, Performance Efficiency, and Reliability. Each item was rated on a five-point Likert scale ranging from 1 (Strongly Disagree) to 5 (Strongly Agree), with descriptive equivalents of Excellent (μ

≥ 4.50), Very Satisfactory (3.50–4.49), Satisfactory (2.50–3.49), Fair (1.50–2.49), and Poor ($\mu < 1.50$). The instrument was administered to [insert number] respondents composed of [insert respondent profile: e.g., business owners, cashier staff, and IT faculty evaluators]. Table 1 presents the evaluation dimensions and the corresponding number of instrument items.

TABLE 1. Evaluation Instrument Dimensions Based on ISO/IEC 25010

Quality Dimension	No. of Items	Description
Functional Suitability	5	The degree to which the system delivers functions that line up with the stated and also implied needs under the particular conditions given.
Usability	5	Effectiveness , efficiency, and satisfaction with which users reach their intended goals and objectives.
Security	4	Protection of system information, and controlled access for users that are authorized only.
Performance Efficiency	4	Response time, resource usage, and throughput under stated conditions, more or less predictable.
Reliability	4	System availability , fault tolerance and recoverability under the expected operating condition.
Total	22	

Note: Replace respondent count and profile in the evaluation description paragraph above with actual study data.

IV. Results and Discussion

3.1 Functional Suitability and Usability

The system successfully executed all required functions across evaluation sessions, including sales transaction processing, real-time inventory updates, user authentication, and report generation. The interface supported task completion with minimal navigation errors; users were able to log in, register products, process sales, and retrieve reports without requiring extended instruction or technical assistance. Automated total calculations eliminated point-of-sale arithmetic errors that characterize manual checkout processes, and

inventory levels were observed to synchronize with transaction records immediately upon sale completion. Report generation and export functions were completed within response times acceptable for the intended operational context.

All test cases were resolved following the defect remediation cycle, confirming that the system met its defined functional requirements. These findings align with those of Ramadhani et al. [2], whose evaluation of a comparable web-based POS system similarly demonstrated high functional completeness and user satisfaction in small-scale retail environments. The system’s usability profile

suggests it is suitable for adoption in businesses where staff may have limited prior experience with digital retail management tools.

Table 2 presents the mean evaluation scores per quality dimension. The system obtained an overall weighted mean of 4.43 (SD = 0.44), interpreted as Excellent on the adopted five-point scale. The highest-rated dimension was Security (M = 4.61, SD = 0.38), reflecting user confidence in the

role-based access control mechanisms. Functional Suitability ranked second (M = 4.52, SD = 0.41), corroborating the functional testing outcomes. Performance Efficiency received the lowest mean score (M = 4.21, SD = 0.53), consistent with the scalability constraints identified during high-load testing. Replace all mean and SD values in this paragraph and in Table 2 with computed values from your actual evaluation data.

TABLE 2. System Evaluation Results by Quality Dimension

Quality Dimension	Mean	SD	Interpretation
Functional Suitability	4.52	0.41	Excellent
Usability	4.38	0.47	Very Satisfactory
Security	4.61	0.38	Excellent
Performance Efficiency	4.21	0.53	Very Satisfactory
Reliability	4.44	0.43	Excellent
Overall Weighted Mean	4.43	0.44	Excellent

Note: Mean scores are based on a five-point Likert scale. Interpretation: Excellent ($\mu \geq 4.50$), Very Satisfactory (3.50–4.49). Replace all values with results computed from actual respondent data.

3.2 Security, Compatibility, and Reliability

Role-based authentication protected sensitive system functions by restricting access according to user designation. Administrators retained exclusive control over product catalog management, user account settings, and full financial reporting, while cashier/staff access was appropriately scoped to operational transactional functions. This stratified access model reduced the risk of unauthorized data modification, inadvertent record deletion, or exposure of business-sensitive financial summaries.

Cross-browser compatibility testing confirmed consistent interface rendering and functional behavior across Google Chrome, Mozilla Firefox, and Microsoft Edge. No critical functionality failures were observed during multi-session extended testing across supported browser environments. The system maintained stable

operation throughout all testing periods, with no data corruption or session loss events recorded. These reliability results are consistent with the operational demands expected of POS deployments in small retail settings, where system downtime directly impacts transaction throughput and customer experience.

3.3 Performance Efficiency and Limitations

The system delivered responsive transaction processing and inventory updates under standard testing loads. Core operational tasks—including user login, sale processing, inventory adjustment, and report generation—were completed within operationally acceptable time frames for the simulated transaction volumes. However, testing at elevated concurrent transaction volumes and with

large product catalogs exceeding several hundred SKUs exposed measurable performance degradation, suggesting that the current architecture is best suited to small to medium-scale retail operations.

During those testing runs, the performance benchmarks was all kind of all-over but still, the average transaction processing time landed around 1.24 seconds when it was basically one user at a time. At the same time, inventory synchronization latency averaged roughly 0.68 seconds for every completed sale. Both of these stayed inside the 2-second limit that most people generally call acceptable for retail POS set ups. But when we pushed it into a simulated peak load situation, like 15 concurrent sessions doing about 120 transactions each minute, the average response time climbed up to 2.91 seconds. That result kind of confirms the scalability constraints that were mentioned above and it was pretty visible in the measured behavior. Also, replace all bracketed values with the measured benchmark data pulled directly from system testing.

Identified technical limitations include the absence of database query optimization for high-volume environments, lack of server-side load balancing, and limited depth in the analytics reporting module. These constraints are consistent with the observations of Sang et al. [4], who noted that POS platforms without embedded predictive or BI capabilities provide insufficient analytical depth as business scale and data volume increase. The current system's reporting module, while functional for day-to-day operational needs, does not yet support the demand forecasting or sales trend analysis capabilities that would be required to fully operationalize the AI-augmented POS architectures described in the literature [1], [4].

3.4 Conclusions and Recommendations

The data-driven POS system developed in this study successfully addresses the core operational inefficiencies associated with manual retail transaction management. By automating sales recording, synchronizing inventory levels in real time, and generating structured financial reports, the system provides small retail businesses with an

accessible, reliable, and cost-effective operational platform. The selection of PHP and MySQL ensures compatibility with locally hosted deployments that do not require continuous internet connectivity—a practical advantage in settings where cloud-based service reliability cannot be guaranteed.

The system's evaluated performance confirms that it functions as a viable digital transformation entry point for small retail enterprises currently relying on manual methods. Several enhancements are nonetheless recommended for future development iterations. First, cloud-based deployment capability should be explored to enable remote system access, automated off-site data backup, and multi-device synchronization. Second, integration of predictive analytics features—leveraging historical transaction data to forecast demand, identify slow-moving items, and support proactive inventory replenishment—would bring the system closer to the AI-augmented POS environments described by Sang et al. [4] and Mattegunta [1]. Third, the addition of a customer loyalty and promotions module would extend the system's value to customer-facing retail operations. Fourth, mobile application support would improve operational flexibility for business owners requiring remote performance monitoring. Finally, systematic database indexing and query optimization are recommended to sustain response performance as transaction volume and product catalog depth grow.

Taken together, these findings and recommendations position the system as a scalable, extensible platform that meets immediate small business operational needs while remaining capable of growth toward the analytically richer POS environments that current literature identifies as benchmarks for data-driven retail management [2], [5], [6].

References

- [1] V. K. P. Mattegunta, "Retail-time inventory visibility: Transforming retail operations through real-time data integration," *Journal of Computer Science and Technology Studies*, vol. 7, no. 3, pp. 58–64, 2025. doi: 10.32996/jcsts.2025.7.3.7

- [2] I. Ramadhani, R. Nindyasari, and A. C. Murti, "Design and development of a web-based point of sale system for small-scale retail management," *bit-Tech*, vol. 8, no. 1, pp. 181–189, 2025. doi: 10.32877/bt.v8i1.2487
- [3] J. Susanto and I. Fenriana, "Web-based POS with Apriori market basket analysis," *bit-Tech*, vol. 6, no. 3, pp. 271–280, 2024. doi: 10.32877/bt.v6i3.926
- [4] L. T. Sang, R. Ravikumar, S. Aarthi, and T. L. Ben, "Optimizing sales and inventory performance using AI and business intelligence in POS environments," in *Advances in Computational Intelligence and Robotics*, pp. 259–290, 2025. doi: 10.4018/979-8-3373-4392-1.ch011
- [5] M. Velasco, X. Domínguez, T. A. Gamón, M. Macías, and R. Torres, "Design and proposal of a POS system as a digitalization strategy for small restaurants," *Procedia Computer Science*, vol. 277, pp. 1369–1379, 2026. doi: 10.1016/j.procs.2026.02.176
- [6] R. Yulia and D. N. D. Seputro, "Digital transformation of inventory management: Evaluating the impact of FIFO-based accurate point of sale system implementation on warehouse operations," *International Journal Administration Business and Organization*, vol. 6, no. 3, pp. 378–393, 2025. doi: 10.61242/ijabo.25.663