MIKAILALSYS
Journal of Multidisciplinary Sciences

e-ISSN : 2987-2286
p-ISSN : 2987-3924
Index: Harvard, Boston, Sydney, Copernicus, Sinta 4, Scilit, Semantic, Google, etc

https://doi.org/10.58578/mikailalsys.v4i3.12097

Integrated Web-Based Point of Sale and Sales Analytics: A Case Study of Ngunjuk

Gusti Muhamad Sardana & Binastya Anggara Sekti
Universitas Esa Unggul, Indonesia
sardanagusti501@gmail.com; Anggara@esaunggul.ac.id

Submitted:	Revised:	Accepted:	Published:
Aug 2, 2026	Aug 30, 2026	Sep 11, 2026	Sep 16, 2026

Abstract

Limited transaction records constrain microenterprises' ability to trace sales and evaluate business performance, highlighting the need for integrated digital tools that convert operational data into actionable information. This study developed a web-based point-of-sale system incorporating sales analytics through a case study of Ngunjuk, a beverage microenterprise in Tangerang, Indonesia. The system was developed using Rapid Application Development, encompassing requirements planning, user design, construction, and cutover. Data were collected through direct observation, semi-structured interviews, and a literature review. The resulting application integrates employee-facing transaction processing with an owner-facing administrative dashboard and provides structured transaction recording, searchable transaction histories, product and category performance analysis, period-based sales trends, monthly sales reports, promotional-package evaluation, and internal financial estimates. Black-box testing covered 62 functional scenarios, comprising 28 employee point-of-sale scenarios and 34 administrative-dashboard scenarios, all of which produced the expected outcomes. User acceptance testing with the owner and an employee further confirmed the acceptance of all nine role-specific scenarios. These findings demonstrate that integrating transaction processing with sales analytics can transform daily sales records into traceable information that supports operational control and managerial evaluation. This study contributes context-specific evidence concerning the digitalization of sales management in microenterprises. Nevertheless, because the findings derive from a single case study, they do not establish the system's generalizable effectiveness across

Mikailalsys is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License

microenterprise settings. Future research should therefore evaluate longitudinal use, usability, security, and system performance in broader operational contexts.

Keywords: Microenterprise; Point-of-Sale System; Rapid Application Development; Sales Analytics; Web Application

INTRODUCTION

Digital transformation has become increasingly important for micro, small, and medium enterprises (MSMEs) seeking to improve operational efficiency, information accessibility, and competitiveness. Digital tools enable routine activities to be organized more consistently and provide business owners with information that can support timely decisions. Nevertheless, many Indonesian MSMEs still face constraints in adopting technology and converting daily operational activities into reliable business information (Aini et al., 2024). Digitalization should therefore be understood not merely as the use of software, but as the systematic restructuring of business processes and data.

Sales transactions constitute one of the most important data sources in a commercial microenterprise. When recorded at item level, transaction data can reveal revenue, sales frequency, product demand, and changes in performance across periods. Conversely, manual notes and aggregate cash calculations often leave transaction details fragmented, difficult to verify, and unavailable for later analysis (Nuha & Nurdewanto, 2026). Inadequate documentation consequently limits traceability and reduces the owner's ability to evaluate the business using historical evidence.

A point-of-sale (POS) system offers a structured mechanism for processing sales while simultaneously documenting each transaction. Web-based POS applications can accelerate checkout and reduce recording errors (Mulyana & Rusmawan, 2023; Prayudha et al., 2024). They can also preserve transaction histories and support periodic reporting (Rusmara & Ma'sum, 2025). However, these benefits depend on whether the system reflects the enterprise's workflow, product characteristics, user responsibilities, and managerial information needs. A generic cashier application may record payments but still fail to generate the contextual information required by the business owner.

Ngunjuk is a beverage microenterprise established in 2018 in Solear, Tangerang Regency, Indonesia. The business is operated by one owner and one employee and offers

beverages across multiple categories, product variants, and sizes. Its daily process begins when a customer places an order, after which the owner or employee prepares the beverage, communicates the total price, receives payment, and continues serving subsequent customers. At the end of operating hours, the owner manually calculates the total cash received as the daily revenue.

The existing procedure preserved only the aggregate daily total and did not create connected records of product name, size, quantity, selling price, order time, or cashier identity. Consequently, earlier transactions were difficult to retrieve, differences between high- and low-performing products could not be established reliably, and sales development across periods could not be compared. This limitation also made evaluation dependent on memory and cash totals rather than detailed transactional evidence. Ngunjuk therefore required a system that could capture transactions at the point of sale and subsequently transform them into accessible sales information.

Previous studies have developed POS applications using web, progressive-web-app, and mobile technologies. Some systems emphasize inventory and purchasing functions (Agustina et al., 2026; Angellin et al., 2023), whereas others concentrate on cashier transactions and stock control (Kumaini et al., 2025; Tanamal & Mourent, 2025). Web-based applications have also been implemented in different retail product contexts (Sakinah et al., 2025; Suprianto et al., 2021), while Sumarto (2023) focused on a mobile POS interface prototype. RAD has supported iterative POS construction (Avivah et al., 2025; Melvin et al., 2023), and its integration with design thinking has strengthened user-centered prototyping (Dimentieva et al., 2026). Bibliometric evidence further indicates growing attention to POS acceptance and accounting-information functions (Paramastri & Yuhertiana, 2023).

Although earlier studies demonstrate the operational value of POS technology, their functions are shaped by different organizational contexts and frequently prioritize inventory, purchasing, or basic cashier activities. Limited evidence is available on a compact microenterprise application that integrates item-level transaction preservation, category and size variants, role-based access, promotional packages, standard product costs, operating costs, sales targets, product-performance analytics, and period-based reporting without imposing inventory complexity. This study addresses that gap through a system tailored to Ngunjuk's actual division of work. Its novelty lies in connecting front-line transaction capture

with owner-oriented sales and internal financial analytics while preserving the selling price and standard cost recorded at the time of each sale.

Accordingly, this study aimed to develop and evaluate a web-based POS application using Rapid Application Development. The first objective was to provide a systematic process for recording, storing, and retrieving complete transaction details. The second objective was to transform the recorded transactions into period-based information on sales development, product performance, category contribution, and estimated financial performance. By aligning transaction processing, historical-data integrity, and managerial analytics in a single role-based application, the study is expected to offer a practical digitalization model for a small beverage enterprise with limited personnel and focused information requirements.

METHODS

Data Collection and Requirement Analysis

This study applied a single-case design and development approach using Rapid Application Development (RAD) to develop a sales management system for Ngunjuk, a beverage microenterprise in Tangerang Regency, Indonesia. The case was selected because its existing sales recording method only generated daily aggregate reports without structured transaction histories or period-based business insights. Conducted from November 2025 to July 2026, the study involved the owner and one employee as key informants and system users. Data were collected through observation, semi-structured interviews, and literature review to identify operational workflows, recording problems, reporting needs, and required system features. Supporting data, including product information, pricing, costs, and transaction procedures, were analyzed to define functional and nonfunctional requirements, which were refined through iterative prototype validation with users to ensure alignment with business processes.

Table 1. Core Functional Requirements

Actor	Required Capability
Owner and employee	Authenticate with registered accounts and access functions according to assigned roles.
Employee	Select active products and sizes, manage the cart, add order notes, choose payment, confirm an order, and print a receipt.

Actor	Required Capability
System	Generate an order code and atomically store date, time, user, payment total, and item-level transaction details.
Owner	Manage categories, products, sizes, selling prices, standard costs, promotional packages, operating costs, sales targets, users, roles, and permissions.
Owner	Search transaction histories and review period-based revenue, order volume, units sold, product performance, category contribution, and sales trends.
Owner	Access monthly revenue reports, internal financial estimates, and login/logout activity records.

Table 1. Core Functional Requirements, shows that the proposed application was designed to support the interconnected responsibilities of the owner, employee, and system. Both users require authentication and role-based access, while the employee requires functions for processing complete sales transactions. The system is responsible for generating order codes and storing transaction information at the order and item levels. Meanwhile, the owner requires broader administrative and analytical capabilities, including master-data management, transaction-history retrieval, period-based sales evaluation, monthly reporting, internal financial estimates, and activity-record monitoring. These requirements establish the functional boundaries of the application and reflect the division of responsibilities within Ngunjuk's operational process.

RAD Development Procedure

The development followed four RAD stages. Requirements planning documented the current workflow, problems, system boundary, user roles, data entities, and functional and nonfunctional requirements. User design translated these requirements into a proposed business process, use-case, activity, sequence, and class diagrams, followed by interface wireframes. Construction implemented authentication, role-based authorization, master data, POS transactions, transaction history, receipts, dashboards, internal financial analysis, and reports with Laravel, PHP, MySQL, and Docker. User design and construction were repeated whenever feedback from the owner or employee indicated a mismatch between the prototype and the validated business requirements. Cutover integrated all modules, executed functional and acceptance tests, corrected discrepancies, and prepared users to operate the application.

Use Case Diagram

The use-case diagram defines the functional scope of the proposed application and the interactions between its two primary actors. The employee can log in, access the POS interface, process sales transactions, review transaction history, and log out. The owner can log in, access the administrative dashboard, review sales and financial summaries, manage operating costs and sales targets, manage categories and products, monitor transactions and monthly reports, manage users, review activity records, and log out. Role-based authorization restricts each actor to the functions associated with their operational responsibilities.

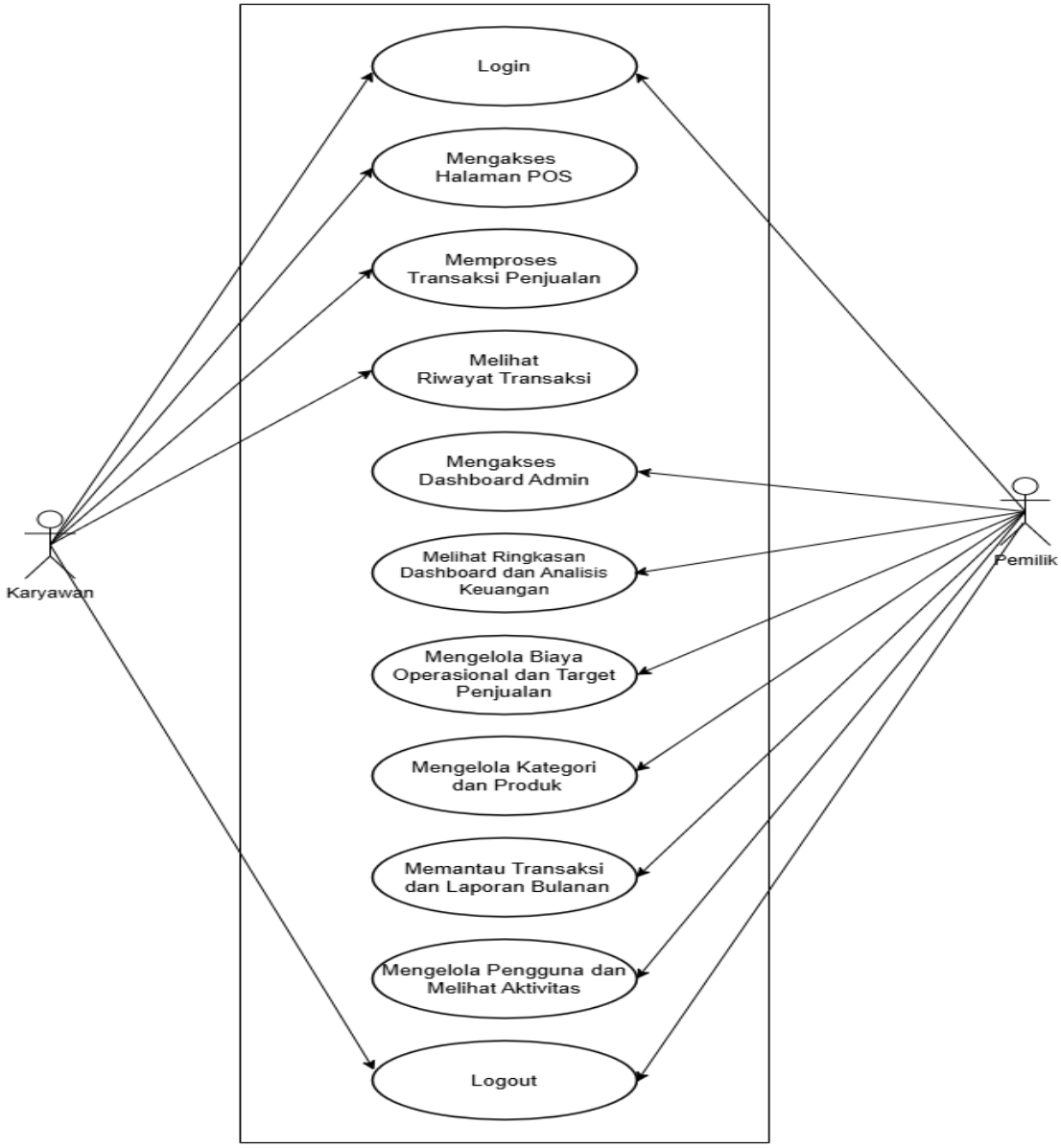


Figure 1. Use-case diagram of the proposed POS application

Figure 1. Use-Case Diagram of the Proposed POS Application, shows the functional boundaries of the application and the interactions between the employee and owner roles.

The employee is provided with functions for authentication, sales-transaction processing, transaction-history retrieval, and logout. In contrast, the owner is provided with administrative and analytical functions, including sales monitoring, financial-summary review, master-data management, transaction and report review, user management, and activity-record monitoring. The diagram also demonstrates that role-based authorization limits each user's access according to their operational responsibilities.

Proposed POS Business Process

The proposed business process integrates employee-facing transaction activities with system-based transaction recording, as illustrated in Figure 2. The process begins when the employee logs into the application and the system verifies the account. The employee then selects a product or promotional package, specifies the size and quantity, adds optional order notes, reviews the cart, selects the payment method, and confirms the transaction.

After confirmation, the system validates the transaction, generates a unique order code, and stores the order header and item-level details. The system then displays a successful transaction notification and generates a printable receipt. Completed transactions remain accessible through the transaction-history function and provide the data used by the owner-facing sales-performance dashboard, monthly sales reports, and internal financial analysis.

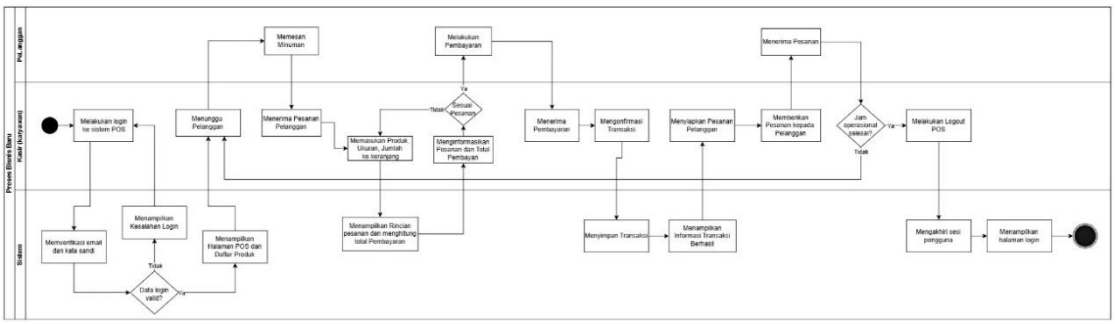


Figure 2. Proposed point-of-sale business process

Figure 2. Proposed Point-of-Sale Business Process, shows the sequence of activities connecting customer ordering, employee transaction processing, and system-based data recording. After successful authentication, the employee selects the ordered product or promotional package, determines its size and quantity, adds optional notes, reviews the cart, and selects the payment method. The system subsequently validates and stores the transaction, generates a unique order code and receipt, and makes the completed transaction available for transaction-history retrieval, sales-performance analysis, monthly reporting, and internal financial analysis.

Financial Information Logic

All financial indicators were calculated for a selected month and year. Revenue was the sum of completed orders. Total cost of goods sold (COGS) was the sum of the stored unit standard cost multiplied by item quantity. Estimated gross profit equaled revenue minus total COGS. Operating cost included one-time costs in the payment month, active monthly costs, one-twelfth of annual costs, and recorded monthly adjustments. Estimated profit after operating costs equaled estimated gross profit minus operating costs. Target achievement was calculated by comparing actual performance with the configured sales target. These indicators were explicitly treated as internal estimates rather than formal accounting profit because taxes, interest, depreciation, and other accounting elements were outside the system scope.

Testing and Data Analysis

Black-box testing evaluated observable system behavior without inspecting source-code logic, consistent with functional-testing applications reported by (Fahrullah et al., 2025; Hidayat & Muttaqin, 2020). The researcher operated the application in Google Chrome in July 2026 while the owner observed the process and assessed whether outputs matched business requirements. Twenty-eight scenarios covered the employee POS and 34 scenarios covered the administrative dashboard. Each scenario was classified as valid when the actual system output matched the predefined expected output. After functional testing, scenario-based user acceptance testing (UAT) was conducted by the two intended users. The employee completed four end-to-end POS scenarios, and the owner completed five dashboard scenarios. A UAT scenario was accepted when the intended user could complete the assigned task and confirmed that the resulting process and output corresponded to the validated business requirements. Acceptance decisions and explanatory responses were recorded for each scenario. The results were analyzed descriptively using scenario counts and success percentages. No inferential statistical analysis was conducted because the UAT involved the two intended users within a single-case setting.

RESULTS

1. Implemented Transaction Process

The resulting application separates the cashier-facing POS from the owner-facing administrative dashboard. After authentication, the employee sees only active products and permitted POS functions. A transaction begins with product and size selection, quantity and optional notes, and cart review. The application recalculates subtotals and the final amount whenever an item is added, changed, or removed. On confirmation, it validates the order, generates a unique order identifier, and stores the order header and its item details as one process. It then presents a success message and a printable receipt containing the product, size, quantity, price, and total payment. The employee can subsequently locate previous transactions through the history page.

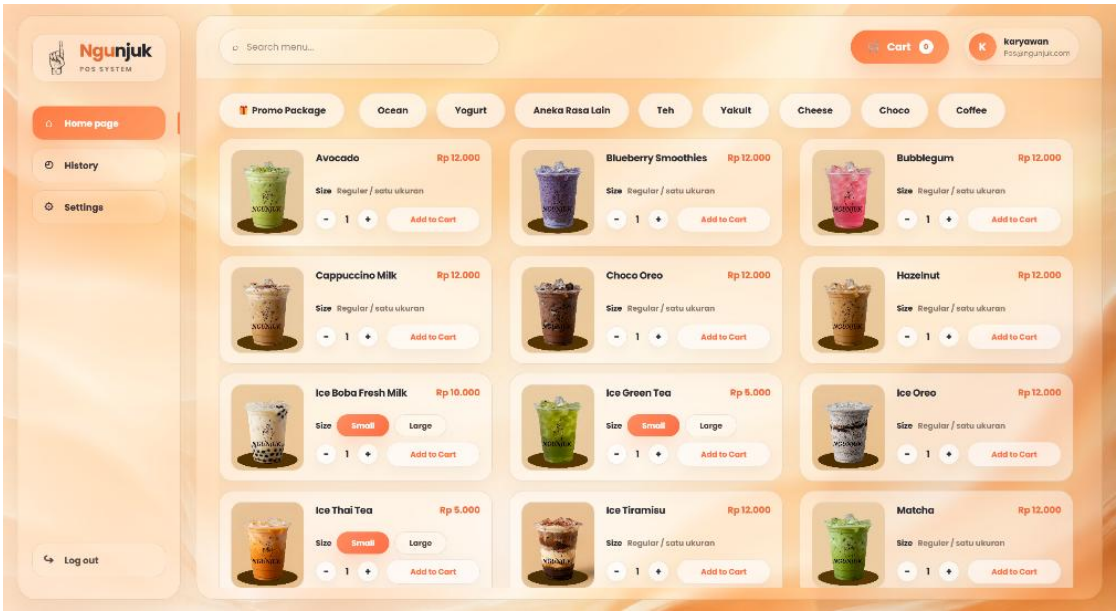


Figure 3. Implemented point-of-sale transaction interface

Figure 3. Implemented Point-of-Sale Transaction Interface, shows the employee-facing interface used to process customer orders. The interface presents active products and their available variants, supports product and quantity selection, and displays the selected items in the transaction cart. Through this interface, the employee can review order details and the calculated payment total before confirming the transaction. The focused interface reflects the employee's operational role by providing transaction-related functions without exposing administrative features.

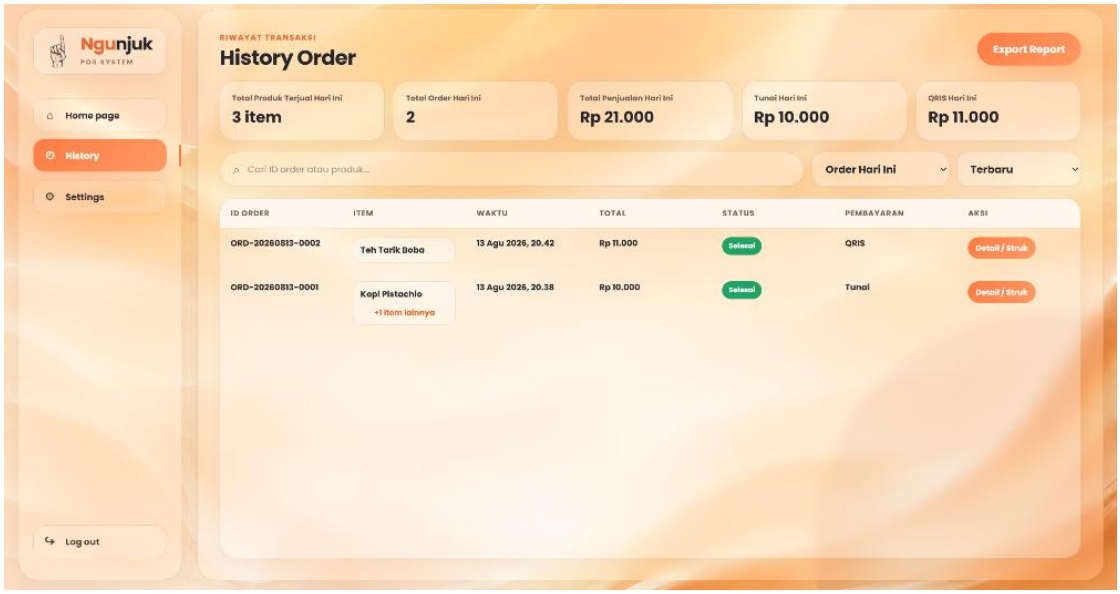


Figure 4. Implemented transaction-history interface

Figure 4. Implemented Transaction-History Interface, shows the function used to retrieve and review previously completed transactions. The interface organizes transaction records using identifiable information such as the order code, transaction date, payment details, transaction status, and payment total. Its searching and filtering functions allow users to locate specific transactions without reconstructing earlier records manually. This interface therefore supports the traceability of the transaction data generated through the POS process.

2. Administrative and Analytical Functions

The administrative dashboard consolidates revenue, number of orders, units sold, product ranking, category contribution, sales by time, and recent orders for a selected period. The owner can move from aggregated indicators to product-level and order-level details. Master-data modules manage categories, products, sizes, selling prices, standard COGS, promotional packages, users, roles, and permissions. Separate modules record operating costs and sales targets, while the financial-analysis page combines transaction, cost, and target data into estimated gross profit, operating cost, estimated profit after operating costs, margin, and target-achievement indicators.

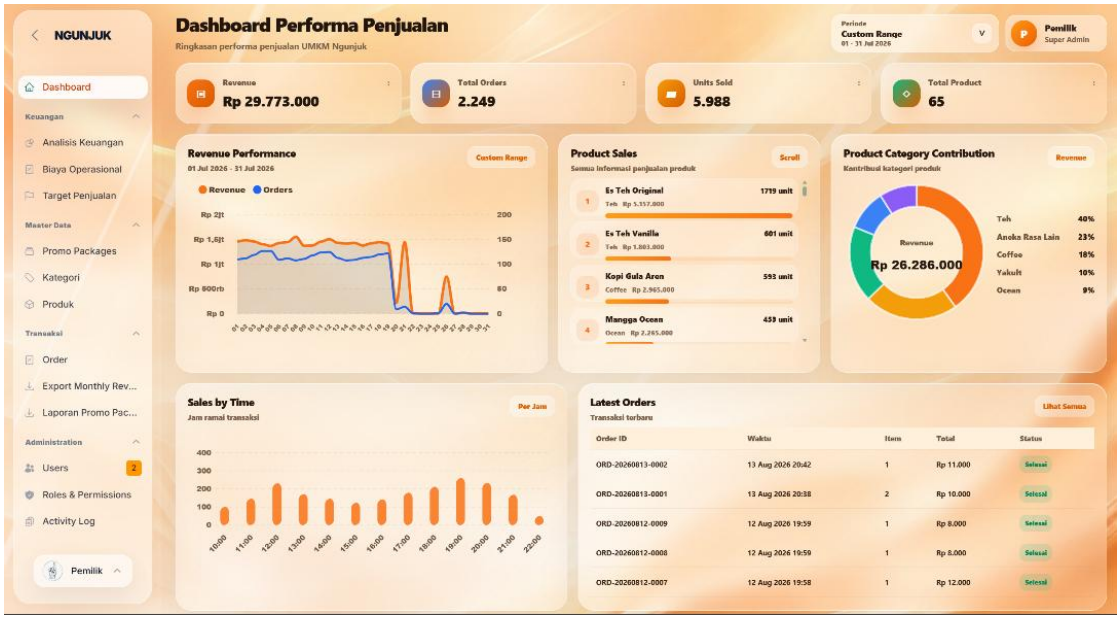


Figure 5. Implemented product and sales-performance dashboard

Figure 5. Implemented Product and Sales-Performance Dashboard, shows the owner-facing analytical interface used to review sales performance for a selected period. The dashboard presents product rankings and category contributions together with revenue trends, order activity, and recent transactions. By combining aggregated indicators with product- and transaction-level information, the dashboard enables the owner to identify differences in product performance and examine the transactional records underlying the summarized results.



Figure 6. Implemented financial-analysis dashboard

Figure 6. Implemented Financial-Analysis Dashboard, shows the managerial interface that combines transaction, operating-cost, and sales-target data for a selected period. The dashboard presents revenue, estimated gross profit, operating costs, estimated profit after operating costs, margin, and progress toward the configured sales target. These indicators provide an internal estimate of commercial performance and are not intended to represent formal net income because taxes, interest, depreciation, and other accounting adjustments are outside the application's scope.

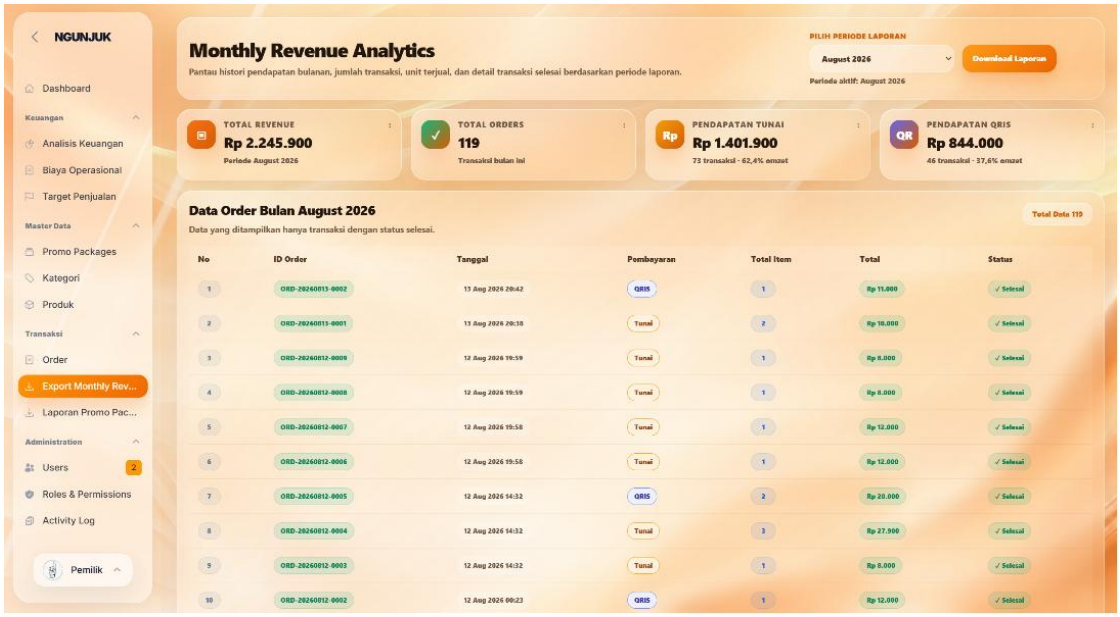


Figure 7. Implemented monthly sales-report interface

Figure 7. Implemented Monthly Sales-Report Interface, shows the reporting function used by the owner to review completed transactions for a selected month and year. The interface summarizes total revenue, the number of completed orders, and the distribution of revenue between cash and QRIS payments. Its transaction table presents the order identifier, transaction date, payment method, number of items, payment total, and transaction status underlying the summary. The selected report can also be downloaded to support business documentation and further review.

3. Black-Box Testing Results

Table 2. Summary of Black-Box Testing

Application Area	Scenarios	Valid	Invalid	Success
Employee POS	28	28	0	100%
Administrative dashboard	34	34	0	100%
Total	62	62	0	100%

Table 2. Summary of Black-Box Testing, shows that all 62 functional testing scenarios produced outputs consistent with the predefined expected results, resulting in a 100% success rate. The 28 employee POS scenarios covered authentication, product and size selection, cart operations, payment confirmation, order-code generation, transaction-data storage, receipt presentation, transaction searching and filtering, and account settings. The 34 administrative-dashboard scenarios covered master-data management, period filtering, order inspection, dashboard calculations, internal financial estimates, report exports, user and permission controls, and activity records. No invalid functional outcome remained at the end of the cutover stage.

4. User Acceptance Testing Results

Table 3. User Acceptance Testing Summary

User	Test Focus	Scenarios	Accepted
Employee	End-to-end transaction, result and receipt review, transaction retrieval, and overall daily POS use	4	4
Owner	Sales monitoring, product and category evaluation, business-data management, transaction/report review, and overall dashboard use	5	5

Table 3. User Acceptance Testing Summary, shows that all nine role-specific scenarios were accepted by the intended users. The employee accepted all four scenarios and confirmed that the sequence from product selection to payment, receipt review, transaction retrieval, and daily POS use was understandable and consistent with Ngunjuk's operational process. The owner accepted all five scenarios and confirmed that the sales-monitoring, product- and category-evaluation, business-data-management, transaction-review, and reporting functions corresponded to the requirements identified during the requirements-planning stage. These results demonstrate acceptance by the two intended users within the tested operational setting but should not be interpreted as generalizable usability evidence.

DISCUSSION

1. From Aggregate Cash Totals to Traceable Transaction Data

The application changes Ngunjuk's sales recording from an aggregate end-of-day cash total into structured order- and item-level data. This result directly addresses the first research objective because each transaction can be retrieved through its order code, date,

product, and reporting period. Comparable POS studies show that structured digital recording improves transaction accuracy and makes sales information easier to retrieve (Mulyana & Rusmawan, 2023; Nuha & Nurdewanto, 2026). The present finding also supports evidence that web-based cashier systems can improve the efficiency of transaction documentation and reporting (Rusmara & Ma'sum, 2025). In the Ngunjuk context, the value of the system therefore lies not only in replacing manual calculation, but also in establishing a persistent and searchable transaction history.

Historical-data integrity further strengthens this contribution. Selling prices, discounts, standard product costs, and calculated values are stored in transaction-detail records, so subsequent changes to master-product data do not alter earlier sales information. The role-based separation between the employee and the owner likewise reflects the actual division of work: the employee receives a focused checkout flow, whereas the owner receives data-management and analytical functions. This arrangement reduces unnecessary interface complexity for the cashier and restricts administrative functions to authorized users. Such alignment is consistent with system-design principles that connect authorization, interaction, and organizational processes rather than simply reproducing a manual form in digital format (Fernando & Wiratama, 2023; Hindarto, 2023).

2. Integrated Sales and Internal Financial Information

The second research objective required stored transactions to be converted into period-based managerial information. The implemented dashboard meets this need by presenting revenue, order volume, units sold, product rankings, category contribution, hourly sales patterns, recent transactions, and monthly reports. These outputs give the owner a more interpretable view of performance than the previous daily cash total. The findings indicate that transaction data become substantially more useful when operational records and analytical summaries are connected within the same system. This connection allows the owner to move from a general indicator to the underlying product or transaction detail when further examination is required.

The system boundary differs from POS studies that combine sales processing with inventory, procurement, or broader merchandise management (Agustina et al., 2026; Angellin et al., 2023). Progressive web-based POS research has similarly demonstrated the value of integrating several operational functions, although the selected modules depend on the needs of each enterprise (Prayudha et al., 2024). Ngunjuk required detailed sales

intelligence without recipes, ingredients, or stock management. The resulting scope shows that a microenterprise can obtain useful managerial information without immediately adopting a complete enterprise-resource-planning model. Functional breadth should therefore be determined by verified business needs rather than by the assumption that every POS implementation must include inventory.

The financial-analysis layer is decision-oriented but intentionally limited. Standard cost of goods sold, operating costs, and sales targets enable the owner to compare estimated commercial performance across selected periods. However, estimated profit after operating costs is not presented as formal net income because the calculation excludes taxes, interest, depreciation, and other accounting adjustments. This qualification prevents POS-derived values from being interpreted beyond the data included in the calculation. Accordingly, the application improves internal financial visibility while preserving the conceptual distinction between operational sales analytics and formal financial reporting.

3. Methodological and Practical Implications

RAD was appropriate for this study because the owner and employee could evaluate concrete prototypes instead of attempting to specify every requirement before development began. Iteration connected requirements planning, interface design, construction, and user feedback. This interpretation is consistent with studies identifying rapid prototyping and user participation as important characteristics of RAD-based POS development (Avivah et al., 2025; Sanmocte & Costales, 2025). The use of design thinking alongside RAD in previous research also supports the value of continuous user involvement when refining system functions and interfaces (Dimentieva et al., 2026).

The 62 valid black-box scenarios demonstrate that the tested functions produced outputs consistent with their specifications. The accepted UAT scenarios further show that the two intended users could complete activities representing their respective responsibilities. Black-box testing and UAT have similarly been used to evaluate functional conformity and user acceptance in information-system development (Fahrullah et al., 2025; Hidayat & Muttaqin, 2020). Nevertheless, the results establish functional correctness and contextual acceptance only within the tested environment. They do not independently prove long-term reliability, cybersecurity, usability at population level, or performance under a substantially higher transaction load.

Practically, the application provides Ngunjuk with a single source for recording daily transactions and conducting periodic evaluation. Product-level evidence can support menu review, pricing decisions, promotional-package assessment, and the establishment of operational targets. The system also offers a broader design lesson for small beverage enterprises: transaction details should be preserved at the moment of sale, analytical indicators should be derived from those records, historical values should remain stable, and access should correspond to actual job responsibilities. These principles may be transferable, but the specific combination of modules should still be adapted to each enterprise's workflow.

4. Limitations and Future Research

This study was limited to one microenterprise, one owner, and one employee; therefore, UAT reflects contextual acceptance rather than statistical generalizability. Testing was conducted in a controlled development environment and did not measure response time, concurrent workload, penetration resistance, browser diversity, or longitudinal operational impact. The application also excludes inventory, recipes, raw materials, procurement, taxes, depreciation, and formal accounting statements. Future research should deploy the system during routine operations for a defined period, compare transaction duration and recording accuracy before and after implementation, apply standardized usability instruments to a broader user group, and conduct security and performance testing. Additional modules, including inventory or mobile access, should be introduced only when longitudinal evidence confirms that they are required by business growth.

CONCLUSION

This study developed a web-based point-of-sale system with integrated sales analytics through a case study at Ngunjuk, a beverage microenterprise in Tangerang, Indonesia. The system replaces aggregate daily cash recording with structured and searchable transaction data. It integrates employee-facing transaction processing with owner-oriented functions, including transaction history, product and category performance analysis, period-based sales trends, monthly sales reports, promotional-package evaluation, operating-cost management, sales targets, and internal financial estimates. These functions enable stored transaction data to be transformed into traceable operational records and decision-oriented managerial information. Black-box testing showed that all 62 functional scenarios, comprising 28

employee POS scenarios and 34 administrative-dashboard scenarios, produced the expected outcomes. User acceptance testing also confirmed the acceptance of all nine role-specific scenarios by the owner and employee. These results demonstrate functional conformity and contextual acceptance within Ngunjuk's operational setting. However, because the study involved a single microenterprise and two intended users, the findings should not be generalized to other business contexts without further evaluation. Future research should examine longitudinal implementation, transaction efficiency, recording accuracy, standardized usability, security, system performance, and applicability across a broader range of microenterprises.

REFERENCES

Agustina, D. S., Novari, S., & Trimarsiah, Y. (2026). Pengembangan Sistem Informasi Point of Sale (POS) Berbasis Web untuk Optimalisasi Pengelolaan Transaksi dan Persediaan Barang pada Toko Mahesa Print. *INTECH, 7*(1), 83–94. https://doi.org/10.54895/intech.v7i1.3438

Aini, A. N., Safitri, E. N., Majalina, G., Abidin, M. Z., & Khoiriawati, N. (2024). Analisis Peluang dan Tantangan Teknologi terhadap Pemberdayaan UMKM di Indonesia. *SENTRI: Jurnal Riset Ilmiah, 3*(5), 2564–2571. https://doi.org/10.55681/sentri.v3i5.2842

Angellin, K., Oetama, R., & Amri, M. (2023). Web-based inventory and sales information system: Indonesian micro small medium enterprise case study. *JOINS (Journal of Information System), 8*(1), 57–66. https://doi.org/10.33633/joins.v8i1.7977

Avivah, N., Irsyad, A., Islamiyah, Setyadi, H. J., & Ibrahim, M. R. (2025). Implementasi Metode Rapid Application Development (RAD) pada Pembangunan Sistem Point of Sale (POS) Berbasis Website di Toko Branded House Sangatta. *Adopsi Teknologi dan Sistem Informasi (ATASI), 4*(2), 183–193. https://doi.org/10.30872/atasi.v4i2.3481

Dimentieva, I., Lutfina, E., Saraswati, G. W., & Caturkusuma, R. M. (2026). Integrating RAD and design thinking for developing a web-based POS and inventory management system for MSMEs: A case study. *Inform: Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi, 11*(1), 80–88. https://doi.org/10.25139/inform.v11i1.11205

Fahrullah, F., Haerullah, H., & Ridhawani, A. (2025). Analisis Blackbox Testing dan User Acceptance Testing terhadap Sistem Informasi Posyandu Dondang. *Journal of Practical Computer Science, 5*(1), 42–50. https://doi.org/10.37366/jpcs.v5i1.5780

Fernando, R., & Wiratama, J. (2023). Developing web-based point of sales application with SHA-512 encryption on DBMS for Indonesian MSME's culinary industry. *Journal of Information Systems and Informatics, 5*(3), 1020–1032. https://doi.org/10.51519/journalisi.v5i3.544

Hidayat, T., & Muttaqin, M. (2018). Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online Menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis. *Jurnal Teknik Informatika UNIS, 6*(1), 25–29. https://doi.org/10.33592/jutis.Vol6.Iss1.38

Hindarto, D. (2023). Indonesian culinary application system design with UML method. *Journal of Computer Networks, Architecture and High Performance Computing, 5*(2), 612–622. https://doi.org/10.47709/cnahpc.v5i2.2675

Kumaini, I. K., Mutamassikin, M., & Triadi, A. (2025). Rancang Bangun Sistem Informasi Point of Sales Berbasis Web pada Toko Sembako Putri. *Jurnal Komputer Teknologi Informasi Sistem Komputer (JUKTISI), 4*(2), 1104–1112. https://doi.org/10.62712/juktisi.v4i2.596

Melvin, M., Wiratama, J., Sutomo, R., & Sanjaya, S. A. (2023). A web-based point of sales for automotive component industry using rapid application development model. *JOINS (Journal of Information System), 8*(2), 167–176. https://doi.org/10.33633/joins.v8i2.9383

Mulyana, A., & Rusmawan, U. (2023). Rancang Bangun Sistem Informasi Point of Sale (POS) Berbasis Web (Studi Kasus Toko Andorio). *Majalah Ilmiah UNIKOM, 21*(1), 43–50. https://doi.org/10.34010/miu.v21i1.10689

Nuha, M. F., & Nurdewanto, B. (2026). Digitalisasi Transaksi UMKM melalui Sistem Point of Sale Berbasis Web. *Journal of Information System and Application Development, 4*(1), 161–168. https://doi.org/10.26905/jisad.v4i1.16919

Paramastri, A., & Yuhertiana, I. (2023). Penerimaan Teknologi tentang Sistem Informasi Akuntansi Point of Sales (POS): Sebuah Analisis Bibliometrik. *eCo-Buss, 6*(1), 326–333. https://doi.org/10.32877/eb.v6i1.812

Prayudha, I. D., Irwansyah, M. A., & Anra, H. (2024). Rancang Bangun Aplikasi Point of Sale (POS App) Berbasis Progressive Web App untuk Usaha Mikro, Kecil, dan Menengah. *JUSTIN (Jurnal Sistem dan Teknologi Informasi), 12*(2), 330–337. https://doi.org/10.26418/justin.v12i2.76824

Rusmara, R. H., & Ma'sum, H. (2025). Pengembangan Sistem Kasir Berbasis Web di Natta Coffee: Studi Implementasi Framework Laravel. *Jurnal Informatika dan Teknik Elektro Terapan, 13*(2), 1142–1149. https://doi.org/10.23960/jitet.v13i2.6414

Sakinah, D. N., Al'Ilmi, Z., Aji, M. I. S., Abdullah, A., & Fikri, M. (2025). Development of a web-based point of sale (POS) system for K-pop merchandise sales with Laravel framework implementation. *Inventa: Journal of Science, Technology, and Innovation, 1*(2), 233–244. https://doi.org/10.65310/c22dzg59

Sanmocte, E. M. T., & Costales, J. A. (2025). Exploring effectiveness in software development: A comparative review of system analysis and design methodologies. *International Journal of Computer Theory and Engineering, 17*(1), 36–43. https://doi.org/10.7763/IJCTE.2025.V17.1367

Sumarto, M. A. (2023). Analisis dan Perancangan Aplikasi Point of Sale Berbasis Mobile Menggunakan Framework Flutter dengan Metode Rapid Application Development (RAD). *Jurnal Studi Komunikasi dan Media, 27*(1), 17–34. https://doi.org/10.17933/jskm.2023.5115

Suprianto, S., Fadlan, M., & Prayogi, D. (2021). Perancangan Aplikasi Point of Sale Berbasis Web pada Toko Project Salfa Tarakan. *Sebatik, 25*(2), 624–631. https://doi.org/10.46984/sebatik.v25i2.1519

Tanamal, R., & Mourent, J. (2025). Rancang Bangun Aplikasi Point of Sales Berbasis Website pada Toko Ramai Rambipuji. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi), 12*(3), 432–442. https://doi.org/10.35957/jatisi.v12i3.11903