APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 7 No. 1 (2026) 579-595
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

# Point of Sales System for Maizan Solution

**Nur Syafiqah Ayop Mahathir[1], Nur Liyana Sulaiman[1]***

[1] *Fakulti Sains Komputer dan Teknologi Maklumat,*
*Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA*

*Corresponding Author: nrliyana@uthm.edu.my
DOI: https://doi.org/10.30880/aitcs.2026.07.01.031

**Article Info**

Received: 31 July 2025
Accepted: 21 June 2026
Available online: 30 June 2026

**Keywords**

Artificial Intelligence, Data Analytics, Point of Sales, Machine Learning Algorithm, MVC, Firebase

**Abstract**

The current manual by Maizan Solutions in Bachok, Kelantan, has led to several operational inefficiencies, including poor inventory tracking, difficulty in identifying near-expiry products, and challenges in forecasting stock demands. To overcome this problem, three objectives have been established: to design a system based on the Object-Oriented Approach, to develop a web-based system, and to perform alpha and beta testing. Agile methodology was applied with eight sprints using the Laravel framework, Firebase and Artificial Intelligence, especially on the machine learning algorithm(linear regression and random forest) with the help of scanner. Eight use cases were applied with sequence diagram, activity diagram and class diagram to apply object oriented approach. The architecture pattern applied was Model-View-Controller (MVC). The results of functional testing (100%) and User Acceptance Testing (UAT) for Administrator in terms of functionality (94.98%), usability (96.65%), design (93.33%) and UAT for Cashier in terms of functionality (97.50%), usability (98.75%), design (98.75%). This project is significant to inventory management and transaction processing systems in the retail industry.

## 1. Introduction

In today's competitive retail industry, businesses rely more on data-driven solutions to improve operational efficiency and customer satisfaction [1]. A Point-of-Sale (POS) system is essential for managing retail operations beyond basic transaction processing [2]. Maizan Solution, located at Bachok, Kelantan, now uses a traditional POS system, which limits its capability to gather insights into business performance and prevents effective decision-making.

The store's manual methods for handling inventory create issues such as difficulty managing stock levels and a lack of reliable sales data, which reduces management efficiency and profit potential. Incorporating AI into POS systems allows for data analysis and reporting, which provides actionable insights that help organizations optimize operations and improve customer experience [3]. Systems based on artificial intelligence can help handle concerns with expired products by providing alerts for things approaching their expiration dates.

Hence, AI-powered POS systems improve operational efficiency, customer service, and predictive decision-making. This innovation enables retailers to remain competitive while lowering inventory losses and increasing consumer loyalty. This report aims to achieve three main objectives which are to design a Point of Sales (POS) system using an object-oriented approach, to develop it as a web-based system, and to conduct alpha and beta testing to ensure its functionality.

The rest of the report is organized as follows. Section 2 discusses the literature review and explores existing technologies related to POS systems. Section 3 explains the methodology used throughout the development process. Section 4 describes the analysis and design of the system. Section 5 focuses on the implementation and presents the results of system testing. Finally, the conclusion summarizes the key findings and offers recommendations for future improvements to enhance the system.

## 2. Literature Review

This section provides a literature review on topics related to the inventory management system. This section aims to gain a comprehensive understanding of the system's project background and conduct a comparative analysis of the system and the existing system.

### 2.1 Inventory Management

Most of the retail stores around the world need to manage their inventory efficiently [2]. Inventory management consists of operations such as managing products, managing categories and generating reports [4]. The stakeholders need to know real-time information about all the products at one time to make important decisions for the grocery store. Sell price, cost price, expiration date and quantity play an important role in maximizing profits for the shop. However, the traditional method does not meticulously keep all this information efficiently and the owner cannot retrieve all the data instantly. Stakeholders need to open files and find receipts from the supplier or calculate the quantity from the shelves directly. Both methods have cons of wasting time and effort.

Modern supply chains demand real-time decision-making, challenging traditional inventory management strategies that rely on static models and manual forecasting [2]. Modern inventory management systems track products from purchase to sale. Store owners can collect quantity statistics from the system to determine the resupply level of products over one month. Modern inventory management provides essential benefits such as cost savings, improved cash flow, and customer satisfaction. Artificial intelligence has grown significantly both in the country and in other countries. A machine learning algorithm such as linear regression is a collection of mathematical techniques used by a system with artificial intelligence to execute tasks. These functions include extracting important insights, trends, and future forecasts from the raw data required to train the algorithm [5]. With modern technologies, predictive analysis, real-time decision making and customer preferences can achieve an efficient Point of Sales System.

### 2.2 Point of Sales System

A POS, or point of sale, allows retailers to accept payments and record transactions. It is a computer-based cash register with software that calculates purchases, accepts payments, tracks inventory and purchasing trends, generates invoices, and collects marketing data [4]. The Point of Sales system can store various kinds of information effectively using a database server. Common features include tracking transactions, managing inventory, and generating reports for sales. The features require any version of a web browser with a stable internet connection. Creating, updating, and deleting are fundamental operations for managing entities such as users and products in any system [5].

The overall goal of point-of-sale (POS) systems in retail is to decrease the workload on business owners by transitioning from manual, paper-based data handling to a digital system that gives real-time insights that are accessible at any time. Manual inventory tracking not only increases staff workload but also takes time, especially in retail outlets where significant data insights are difficult to obtain. A POS system that combines predictive analytics can successfully solve issues such as overstocking, stockouts, and failure to monitor product expiration dates [4]. In response to these requirements, this project was created to provide Maizan Solutions with a customized POS system that streamlines everyday operations, increases inventory accuracy, and enables better decision-making via real-time tracking and smart forecasting.

## 2.3 Comparison with the Existing System

The study analyses current systems and compares them to the developed Point of Sales System for Maizan Solution are comparable. Existing systems that have been chosen include QQ Mart POS [6], Kim Kee Mini Market POS [7], and Mr. DIY POS [8]. Table 1 compares the existing and developed systems.

**Table 1** *Comparison with Existing System and Developed System*

| Features | QQ Mart POS | Kim Kee Mini Market POS | Mr. DIY POS | Maizan Solution POS |
|---|---|---|---|---|
| AI (predict stocks) | ✗ | ✗ | ✗ | ✓ |
| Managing product near expired dates discounts | ✗ | ✗ | ✗ | ✓ |
| AI (customer purchasing behavior) | ✗ | ✗ | ✓ | ✓ |
| Real-time | ✓ | ✓ | ✓ | ✓ |
| Integrate with e-commerce | ✓ | ✓ | ✓ | ✗ |
| Inventory management | ✓ | ✓ | ✓ | ✓ |
| Generate report | ✓ | ✓ | ✓ | ✓ |
| Accept multiple payments | ✓ | ✓ | ✓ | ✓ |
| Customizable promotions | ✗ | ✓ | ✓ | ✓ |

The POS system supports machine learning, which can forecast suggested restocking levels and evaluate top-selling products. These capabilities promote successful inventory management and increase customer satisfaction with real-time data integration. Using artificial intelligence to identify customer trends has a huge impact on sales techniques and shop revenue. Furthermore, the function of controlling discounts for near-expired items reduces unreturnable products while increasing the store's profit margin.

## 2.4 Technology Used

In this project, the Point-of-Sale system is built using the Laravel framework and designed as a web-based application. This means it can be accessed from any device with an internet connection. Laravel was chosen because it has a clean structure and useful tools that make web development faster and more organized. This helps developers build web applications quickly and efficiently. It comes with built-in features like routing, middleware, authentication, and database management, which reduce development time and improve code quality. Compared to other PHP frameworks such as CodeIgniter or Symfony, Laravel provides a more modern development environment and better support for large-scale applications (Shahzad et al., 2020).

Firebase Realtime Database was chosen because it allows real-time data synchronization across all connected clients. Unlike traditional SQL databases, Firebase automatically updates data on all devices when changes are made, which is ideal for Point-of-Sale systems that require instant updates. In addition, Firebase includes features like secure user authentication, analytics, and a user-friendly dashboard. When compared to other backend-as-a-service platforms like AWS Amplify or MongoDB Realm, Firebase stands out for its ease of use and seamless integration with web and mobile apps.

In this system, linear regression and random forest are used to forecast sales and quantity of inventory sold. Linear regression is a simple algorithm that finds the relationship between variables. For example, the number of items sold depends on the day of the week or time of the year. This helps predict future sales trends based on past data.

On the other hand, random forest is a more advanced method that builds many decision trees and combines their results. It can handle more complex data patterns and provides more accurate forecasts. In this project, random forest is used to predict customer traffic by analyzing various features like product type, season

and past sales behavior. Both methods help the system provide smarter suggestions for inventory management and business planning.

## 3. Methodology

The Agile model was used as the software process model because it allows for incremental and iterative development. The design sprint emphasizes an iterative process of designing, prototyping and testing to effectively integrate the principles of design thinking with agile development which ensures rapid feedback and improvement.[9].

Fig. 1 illustrates three sequential sprints, which are sprint 1, sprint 2 and sprint 3 that emphasizes the cyclical nature of agile. This project applies eight sprints in the development process. This project also starts on 6 October 2024. The Gantt Chart has been attached to Appendix A.

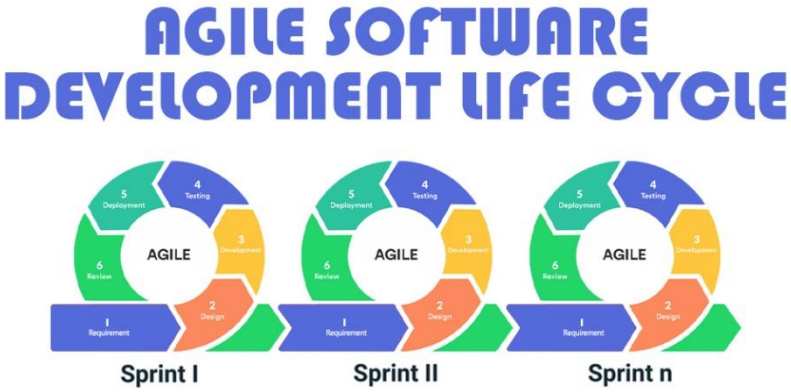



**Fig. 1** *Agile Methodology[10]*

Table 2 explores the agile methodology particularly each sprints focus area and output. Each sprint have their own focus area such as login, sales and checkout. In each sprint, developer need to finish tasks such as define requirements,design interface,implement, test, deploy and review feedback in two until four weeks time range.

**Table 2** *Sprints Overview*

| Sprint | Module | Output |
|---|---|---|
| Sprint 1 | Login Module | Secure login feature |
| Sprint 2 | User and Supplier Module | Manage User and Supplier feature |
| Sprint 3 | Product and Category Module | Manage Product and Category feature |
| Sprint 4 | Sales Module | Manage the transactions list feature |
| Sprint 5 | Checkout Module | Checkout feature |
| Sprint 6 | Process Payment Module | Secure payment feature |
| Sprint 7 | Generate Report Module | Reporting feature |
| Sprint 8 | Data Analytics Module | Analytics dashboard with AI capabilities |

Applied Information Technology and Computer Science Vol. 7 No. 1 (2026) p. 579-595

These sprints have been explored and analysed in the Sprint Planning Meeting as discussed in the next subsection.

## 4. Sprint Planning Meeting

In the Sprint Planning Meeting, the analysis of the system for 8 sprints applies an object-oriented approach while using Unified Modeling Language (UML) as the requirement modelling. The use case diagram defines the scope and context, whereas the class diagram refines and implements the system's functional needs. Furthermore, the architecture and interface design are described in depth.

### 4.1 Unified Modelling Language (UML)

This project applies UML with use case diagram, sequence diagram, activity diagram and class diagram.

#### 4.1.1 Use Case Diagram

The use case diagram consists of eight use cases which are login, manage user, manage supplier, manage product categories, manage products, manage sales, checkout and generate report. The primary actor is an administrator and cashier while customers are the secondary actor of this system as illustrated in Fig. 2.

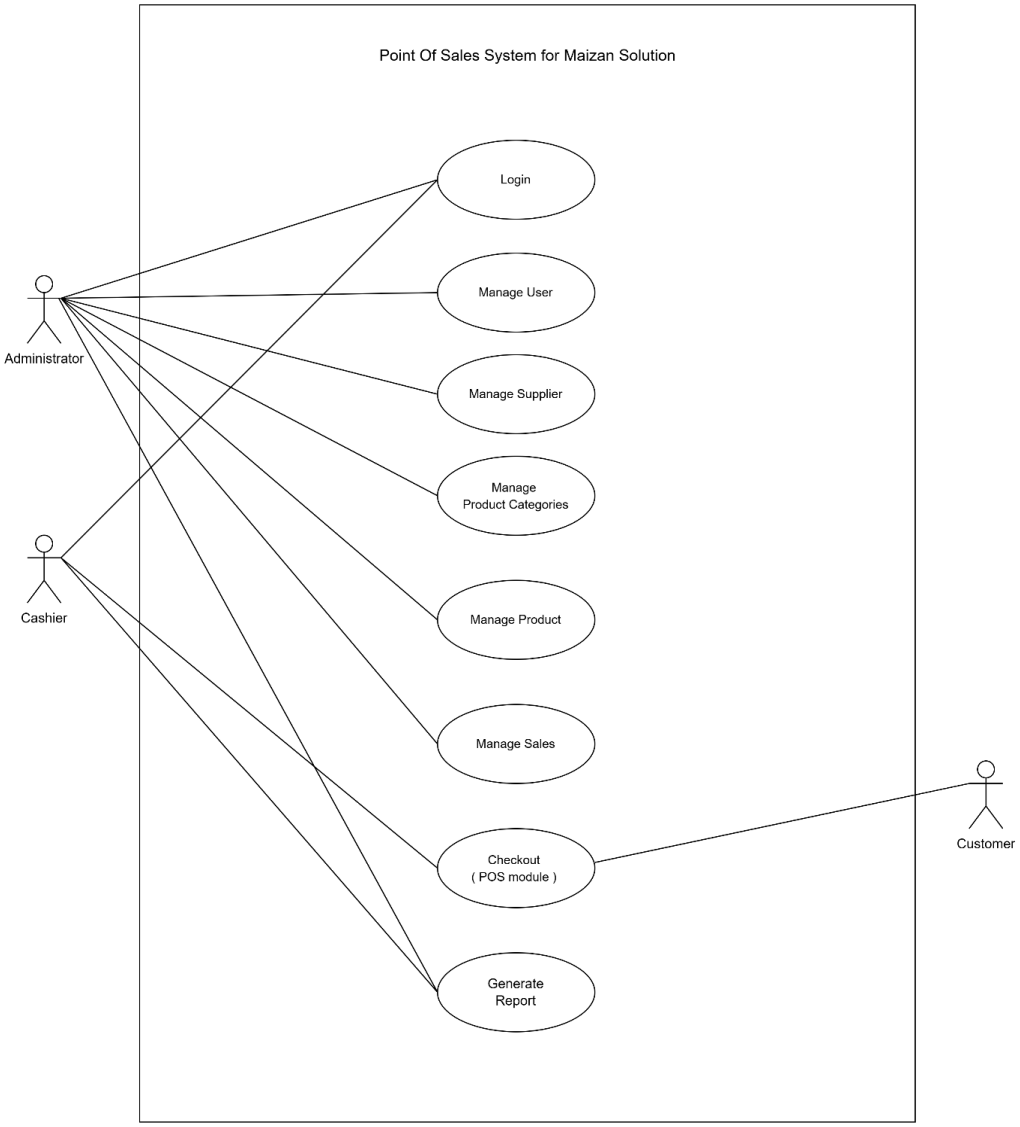



**Fig. 2** *Use Case Diagram*

#### 4.1.2 Activity Diagram for Login (Sprint 1)

Fig. 3 illustrates the steps involved in logging into the system, which were analyzed in Sprint 1 as examples. It involves actions like entering login information, verifying the system and presenting error messages when input appears to be invalid.

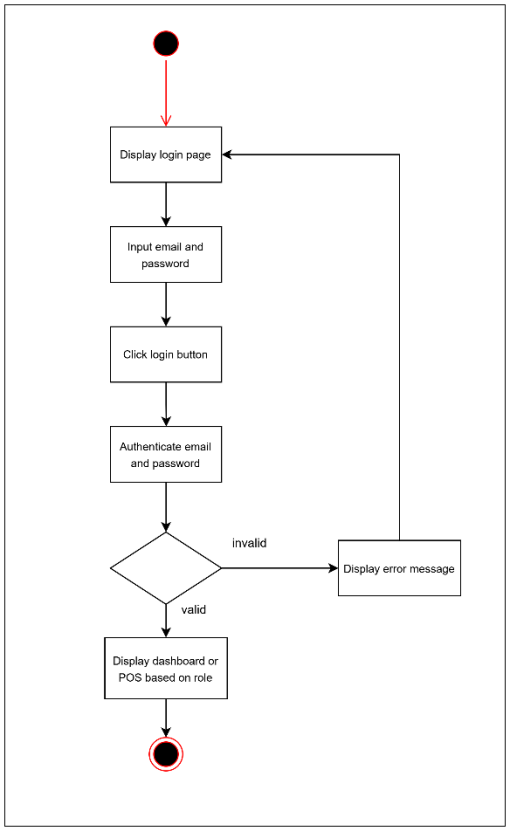



**Fig. 3** *Activity Diagram for Login Module*

### 4.1.3 Sequence Diagram for Login

Fig. 4 shows how the user, login interface and system backend interact throughout the authentication procedure. After a successful login, it reveals the validation and reroutes to the appropriate dashboards or POS page.

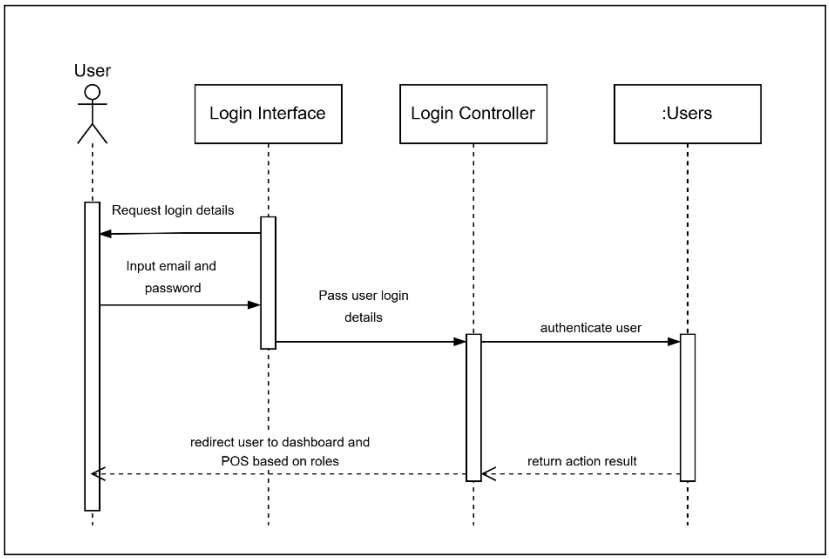



**Fig. 4** *Sequence Diagram for Login Module*

Applied Information Technology and Computer Science Vol. 7 No. 1 (2026) p. 579-595

### 4.1.4 Class Diagram

There are 8 classes inside Fig. 5, which are users consisting of administrator and cashier, supplier, product category, product, sales, sales report, inventory report and profit report. Based on Fig. 5, one table in the database is used to store user information which is the user class. The administrator is associated with supplier, product, product category and sales meanwhile, the cashier is associated with the POS. However, both users are associated with reports, though both the administrator and cashier have different access roles.

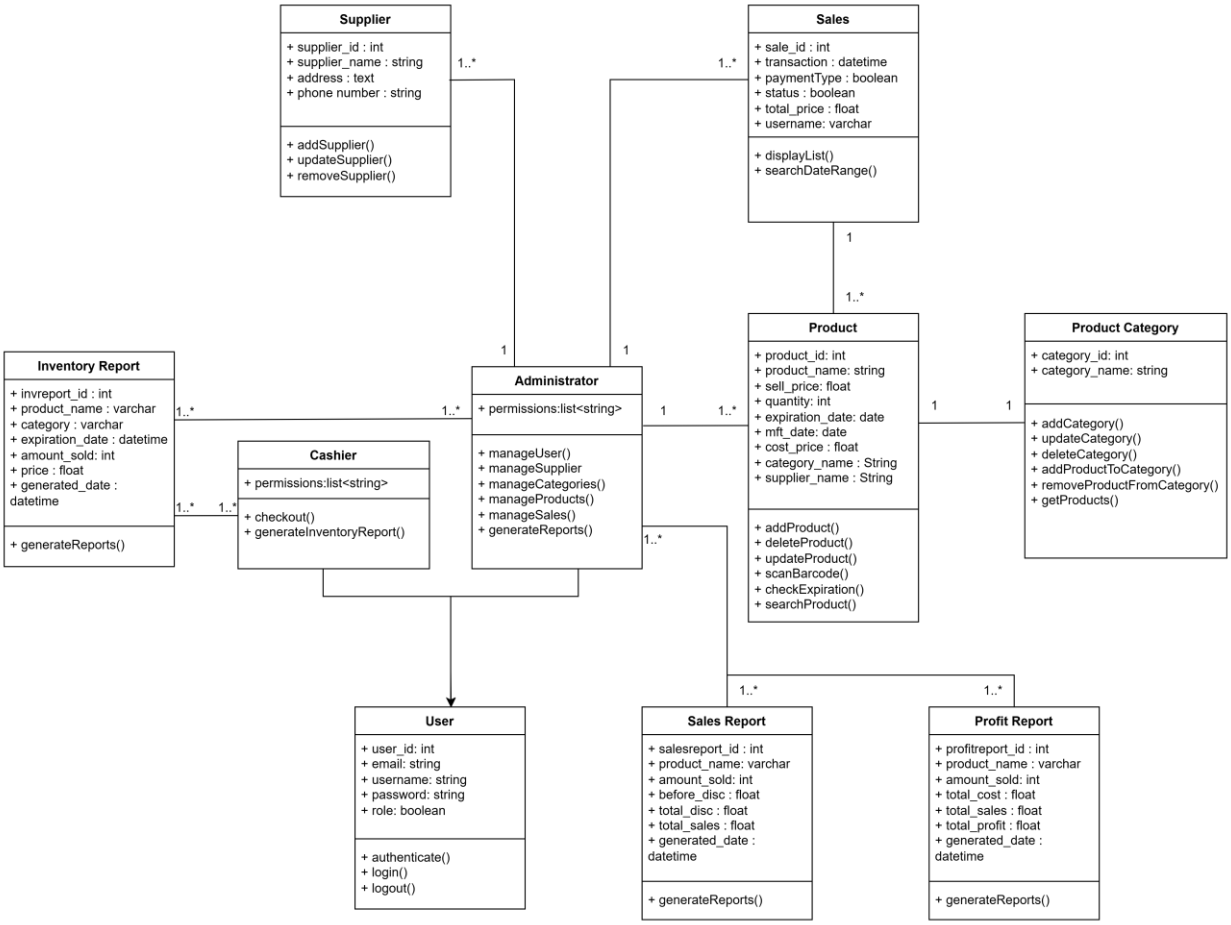



**Fig. 5** *Class Diagram*

## 4.2 System Architecture

This project applies the Model View Controller as an architecture pattern to develop the Point-of-Sales System. Laravel is an MVC framework where the website will be organized into Models, Views and Controllers. These styles contribute to easier data management, separating the front end, the backend and the data. Moreover, Laravel offers built-in tools for authentication and request validation. The model represents data such as products, sales and inventory. Hence, the CRUD operation can be performed.

This model also enforces business rules, which are that when a sale is made, the model updates the inventory and records the transaction. The view displays data using the user interface (UI). The system will retrieve data from the model and render it into a user-friendly format. The controller part will handle user input and update the model based on that input. Fig. 6 displays the POS System architecture pattern. MVC architecture with the Laravel framework standardizes the development process and dramatically lowers unexpected errors, resulting in increased total web application efficiency [11].

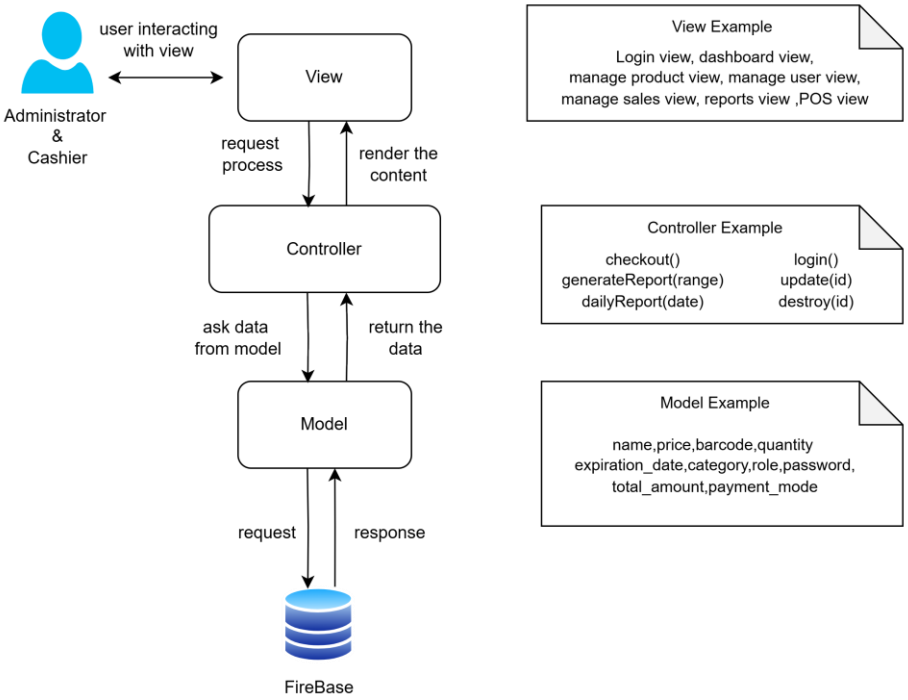



**Fig. 6** *Architecture pattern of POS System for Maizan Solution*

## 4.3 System Interface Design

In this project, Figma is used to develop a high-fidelity wireframe. Fig. 7 displays the transaction list consisting of ID, transaction date and time, amount, status and cashier name. Each successful payment will create a transaction in the sales menu. The administrator can know whether the transaction is complete, canceled, void or returned. The payment method is also stated in the transaction list.

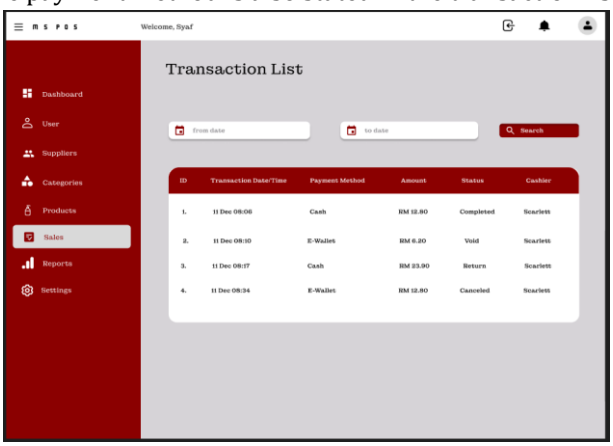



**Fig. 7** *The POS System sales page*

Fig. 8 illustrates that the checkout interface displays date and time at the top right banner and a list of purchasing items. The cashier may click all void buttons to terminate the transaction while clicking the refund product button if there are customers to refund any item. Fig. 9 illustrates the dashboard of the system where only administrators can access this feature. The dashboard displays an overview of each day that consists of total sales, total profit and total sold item quantity. Administrators can also view further information about top-selling products, low stock products and near-expiration date products.

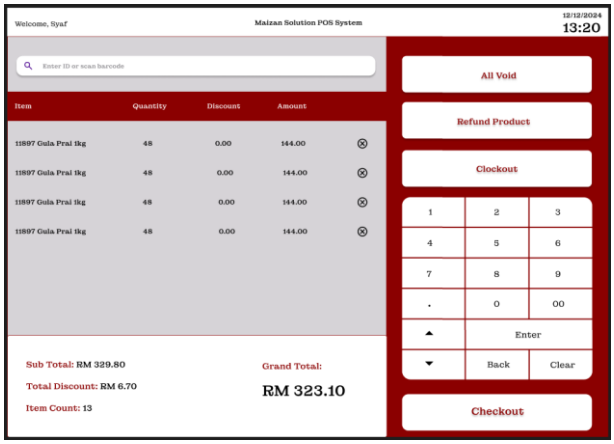



**Fig. 8** *The POS System checkout page*

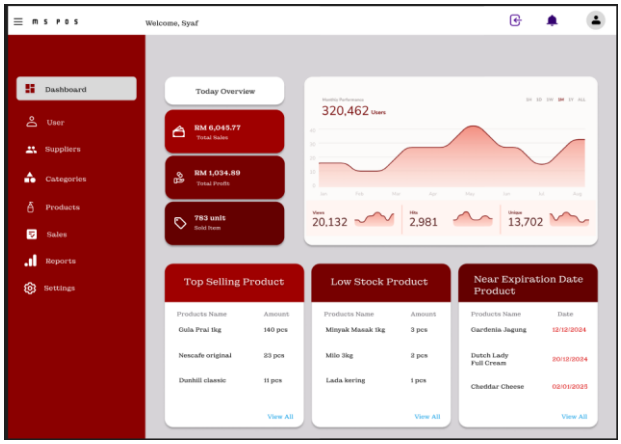



**Fig. 9** *The POS System dashboard interface*

## 5. Implementation and Testing

This section focuses on the implementation and testing of the system. Section 5.1 discusses the implementation of the system, and Section 5.2 discusses the testing of the system, including the requirement traceability matrix, system testing and user acceptance testing.

### 5.1 Functional Modules

In this section, the system implementation, using software code aligns with the design from the earlier stage. Visual Studio Code execute the system code with nine modules that are built with their specific functionality. The screenshot of the user interface and their code logic is presented to show how users interact with the Point-of-Sales System visually especially in Sprint 6. Fig. 10 shows the implementation of the POS module and Fig. 11 displays the refund function interface. Fig. 12 and Fig. 13 show the implementation of the payment process method via cash and QR Code respectively. Fig. 14 displays the code segment of scanning products by barcode scanner.

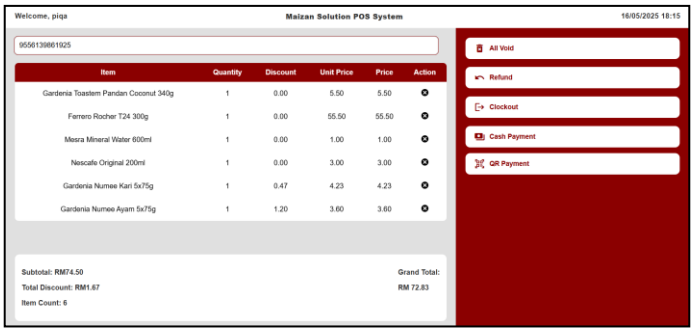



**Fig. 10** *The POS module interface*

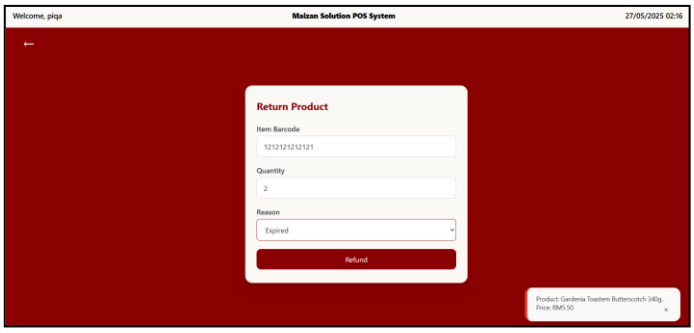



**Fig. 11** *The POS module refund function interface*

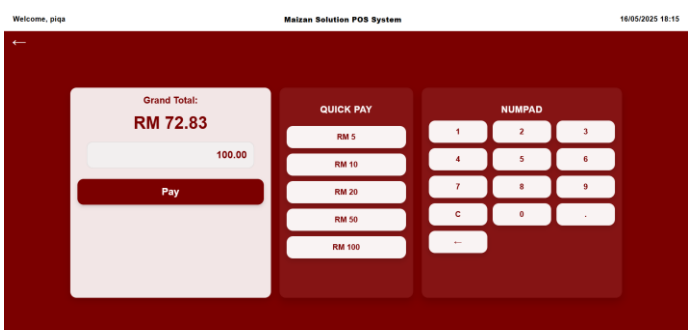



**Fig. 12** *The payment via cash interface*

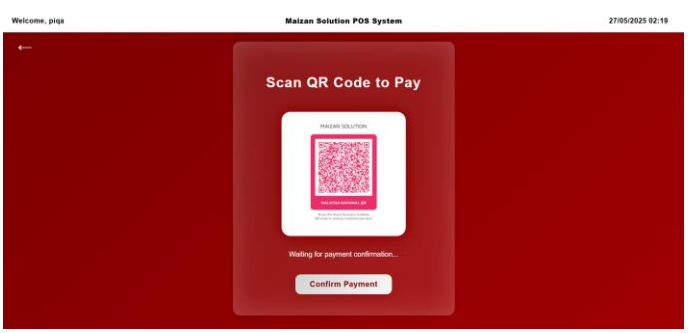



**Fig. 13** *The payment via QR Code interface*

```php
class POSController extends Controller
{
    public function getByBarcode($barcode)
{
    $client = new Client([
        'base_uri' => env('FIREBASE_DATABASE_URL'),
        'verify' => false
    ]);

    $response = $client->get("/products.json");
    $products = json_decode($response->getBody(), true);

    foreach ($products as $product) {
        if (isset($product['barcode']) && $product['barcode'] === $barcode) {

            $name = $product['name'] ?? 'Unnamed';
            $price = (float)($product['selling_price'] ?? 0);
            $category = strtolower($product['category_name'] ?? '');
            $expirationDate = $product['expiration_date'] ?? null;
            $discountRate = 0;

            if ($expirationDate && $category !== 'bread') {
                $daysUntilExpire = (strtotime($expirationDate) - strtotime('today')) / 86400;

                if ($daysUntilExpire < 3) {
                    $discountRate = 0.50;
                } elseif ($daysUntilExpire < 7) {
                    $discountRate = 0.35;
                } elseif ($daysUntilExpire < 14) {
                    $discountRate = 0.25;
                } elseif ($daysUntilExpire < 30) {
                    $discountRate = 0.10;
                }
            }
```

**Fig. 14** *The implementation of adding products to the POS by barcode*

## 5.2 Testing

After finishing the implementation of the Point-of-Sales System for Maizan Solution, the testing phase was carried out to ensure everything worked as expected. This started with referring to the Requirements Traceability Matrix (RTM) to confirm that all the system requirements were met. Each of the nine modules, which are Login, Manage Users and Suppliers, Manage Categories, Manage Products, Sales Management, POS, Process Payment, Generate Reports and Dashboard thorough system testing to check their functionality and integration. Finally, user acceptance testing (UAT) was conducted with real users to make sure the system was easy to use, accurate and met the business needs.

### 5.2.1 System Testing

System testing was done on all nine modules to check if each part of the system worked correctly, both individually and together. These modules included Login, Manage Users and Suppliers, Manage Categories, Manage Products, Manage Sales, POS, Process Payment, Generate Reports and Dashboard. Each module was thoroughly tested for proper functionality, error handling and seamless integration with other system components. Any bugs or issues found were fixed to ensure the system was stable and reliable. The Manage Products module provides comprehensive control over the product inventory. Administrators can add, update, and delete product records including critical details like name, price, category and expiration date. The module supports barcode scanning for efficient product handling, displays product information clearly and enables advanced search capabilities by category or name. It also ensures real-time stock updates after each sale.

The Point of Sale (POS) module is the front-facing interface used by cashiers to process transactions. It performs essential tasks such as calculating total prices, applying discounts, updating inventory in real-time and retrieving product data via barcode scanning. The module also supports multiple payment methods including cash and QR code payments and handles refund processing by storing returned product data securely. Table 3 and Table 4 are the test cases designed for the Manage Products module and the POS module.

**Table 3** *Manage Products module test case*

| Test Case ID | Description | Expected Result | Actual Result | Status |
|---|---|---|---|---|
| TC005-01 | The system shall allow administrators to create, update and delete product details. | Products can be added, edited or removed successfully | As expected | Pass |
| TC005-02 | The system shall display product details. | The product list shows all product data correctly | As expected | Pass |

**Table 3** *Manage Products module test case (Continued)*

| Test Case ID | Description | Expected Result | Actual Result | Status |
|---|---|---|---|---|
| TC005-03 | The system shall allow barcode scanning to add products. | The scanned product is added to the system or cart | As expected | Pass |
| TC005-04 | The system shall allow administrators to search for products using category or product names. | Search returns relevant products based on keywords | As expected | Pass |
| TC005-05 | The system shall deduct the quantity after a product is sold. | Stock quantity decreases accordingly after a sale | As expected | Pass |

**Table 4** *POS module test case*

| Test Case ID | Description | Expected Result | Actual Result | Status |
|---|---|---|---|---|
| TC007-01 | The system shall calculate the total price. | The total is calculated correctly based on quantity and unit price | As expected | Pass |
| TC007-02 | The system shall apply the discount if needed. | Discount is applied based on rules (e.g., expiry, promotion) | As expected | Pass |
| TC007-03 | The system shall update the inventory level after each transaction. | Product stock decreases after a sale | As expected | Pass |
| TC007-04 | The system shall promote barcode scanning of products to retrieve product information. | Scanning a barcode auto-fills product data in the cart | As expected | Pass |
| TC007-05 | The system shall accept payments via cash and QR codes. | System processes both payment types correctly | As expected | Pass |
| TC007-06 | The system shall store the refund products data. | Refunded items are stored in the database with relevant details | As expected | Pass |

### 5.2.2 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) was the final step where the actual users of the system tried it out to confirm it met their needs. They tested the system using real-life tasks and their feedback helped identify any last-minute changes or improvements. The user acceptance which also acted as beta testing for Maizan Solution's Point-of-Sales System was carried out for two user groups, which are administrators and cashiers. Detailed findings will be presented in subsections.

#### 5.2.2.1 User Acceptance Testing for Administrator

During the testing phase, three administrators tested the Point-of-Sales System developed for Maizan Solution and shared their feedback about the user interface and overall experience. All of them said that logging into the dashboard was very easy. They found the dashboard to be well-designed and informative. The sales transaction page was smooth and easy to follow. They also mentioned that managing users, suppliers, categories and products was straightforward. Moreover, they found the system eased their minds as they could just scan the barcode to know the real-time data of the products. In addition, the system allows reports to be generated easily in both PDF and CSV formats.

There were 12 questions analyzed, divided into two parts which are Part A for the system's functionality, Part B for the system's satisfactory usability and Part C for the system's interface design. These criteria were determined on a scale of 1 to 5, where 1 strongly disagree, 2 disagree, 3 neutral, 4 agree and 5 strongly agree. This consistent positive feedback from the admin users highlights the user-friendliness and reliable functionalities of the Point-of-Sales System for Maizan Solution features indicating a strong performance and high satisfaction.

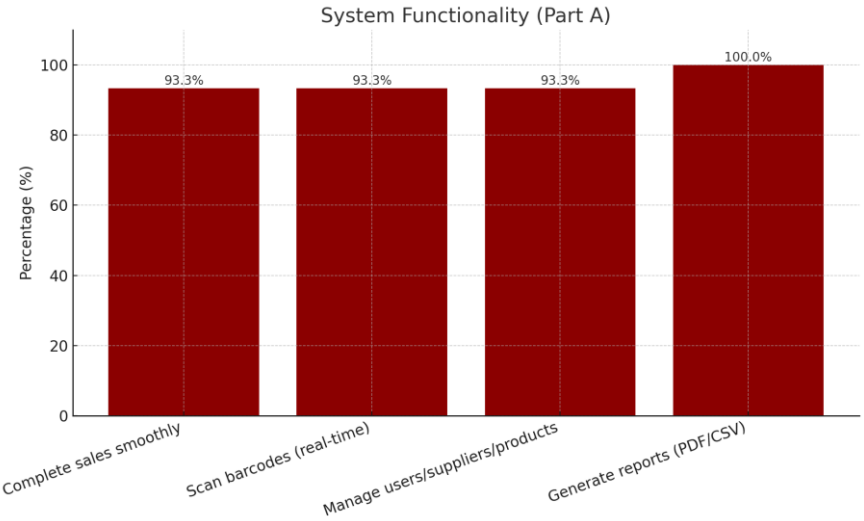



**Fig. 15** *UAT for Administrator: System Functionality*

Fig. 15 displays the results for Part A, which focuses on the system's functionality such as how well it performs tasks and reduces errors. The percentage for complete sales smoothly, scanning product barcodes and users and products management is 93.3%, while generating reports in PDF/CSV is 100%.

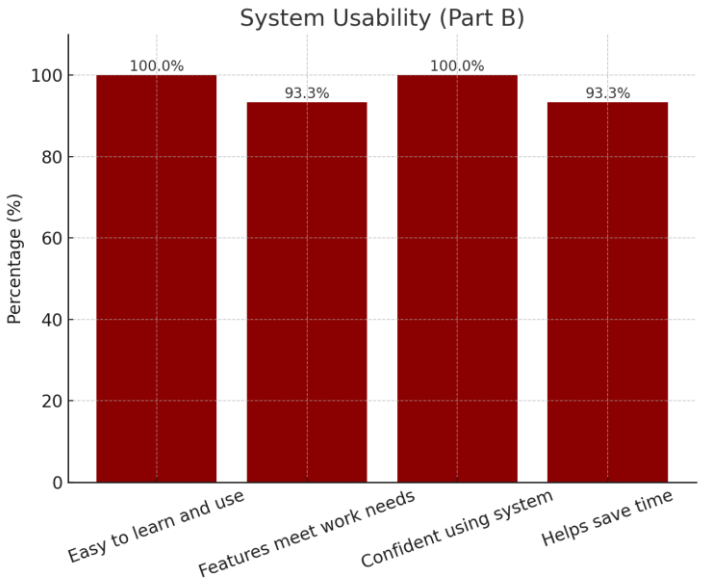



**Fig. 16** *UAT for Administrator: System Usability*

Fig. 16 shows the results for Part B, which measures user satisfaction and how easy the system is to use. The percentage for easy to learn and use as well as confidence in using the system is 100%. In addition, the percentage for features that meet work needs and help save time is 93.3%.

Applied Information Technology and Computer Science Vol. 7 No. 1 (2026) p. 579-595

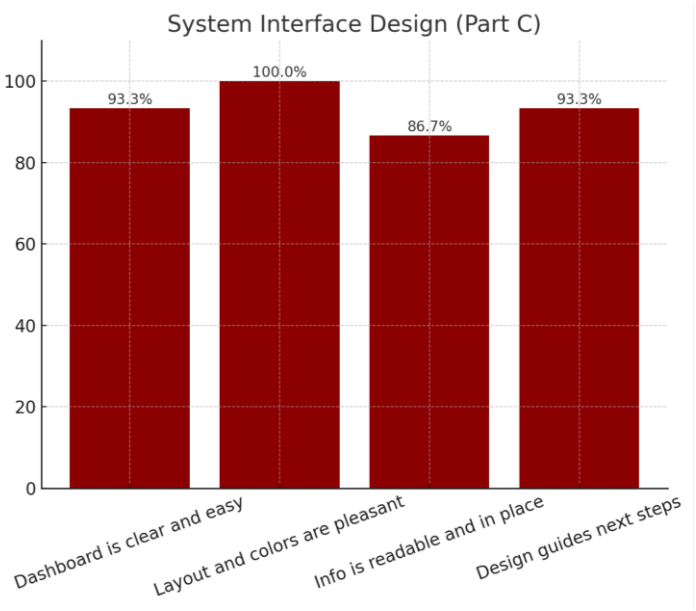



**Fig. 17** *UAT for Administrator: System Interface Design*

Fig. 17 presents the results for Part C, which evaluates the design and layout of the system's interface. The percentage for layout and colors is 100%, while information readability and in place are 86.7%. Both the clear dashboard and design guide the next steps are 93.3%.

### 5.2.2.1 User Acceptance Testing for Cashier

During the testing phase, five cashier users tested the Point-of-Sales (POS) system developed for Maizan Solution to ensure it met their daily operational needs. They found the POS interface simple and easy to navigate especially when processing sales using the barcode scanner which helped speed up transactions and reduce errors. The cashiers were also able to handle product returns smoothly and they appreciated the flexibility of the payment options which included both cash and QR code payments. In addition, they found it helpful that they could clock in and out through the system, making attendance tracking more convenient. They also highlighted how easy it was to generate daily inventory reports in both PDF and CSV formats. Overall, their feedback showed that the system was user-friendly, efficient and well-suited to support their day-to-day tasks.

There were 12 questions analyzed, divided into two parts which are Part A for the system's functionality, Part B for the system's satisfactory usability and Part C for the system's interface design. These criteria were determined on a scale of 1 to 5, where 1 strongly disagree, 2 disagree, 3 neutral, 4 agree and 5 strongly agree. This consistent positive feedback from the admin users highlights the user-friendliness and reliable functionalities of the Point-of-Sales System for Maizan Solution features, indicating a strong performance and high satisfaction.

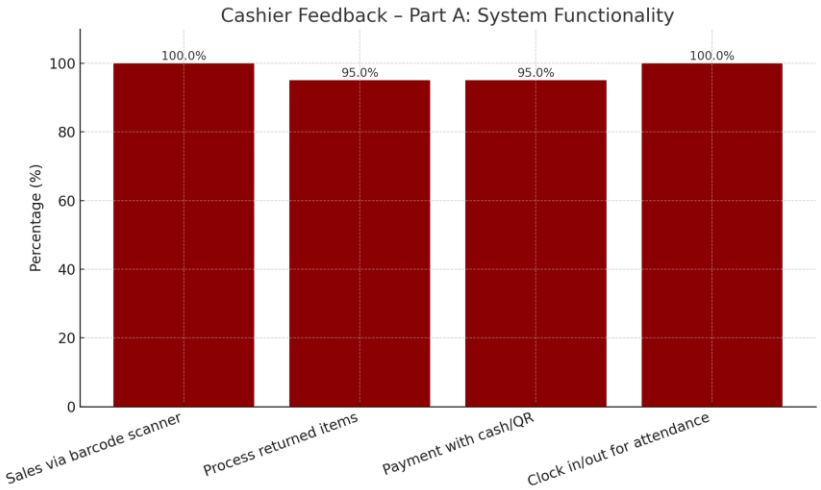



**Fig. 18** *UAT for Cashier: System Functionalities*

Fig. 18 shows the results for Part A, which focuses on how well the system works for cashiers, including tasks like scanning barcodes, handling returns and processing payments. Both sales via barcode scanner and attendance clocking get 100% for performance. Moreover, payment with cash and QR code, as well as processing return items, gets 95% for performance.

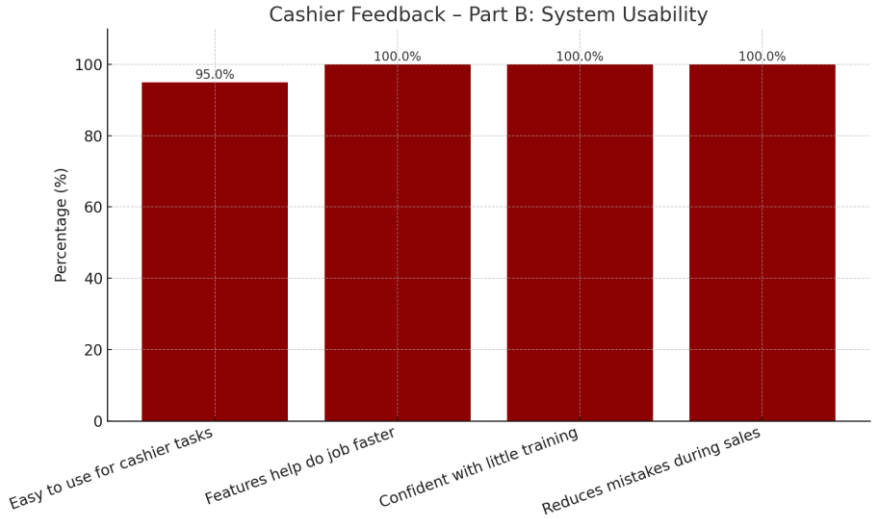



**Fig. 19** *UAT for Cashier: System Usability*

Fig. 19 shows the results for Part B, which looks at how satisfied the cashiers are with using the system in their daily work. The figure displays that users are satisfied with the features to help speed up the payment process, are easy to use and reduce mistakes during sales with a 100% rating for performance.

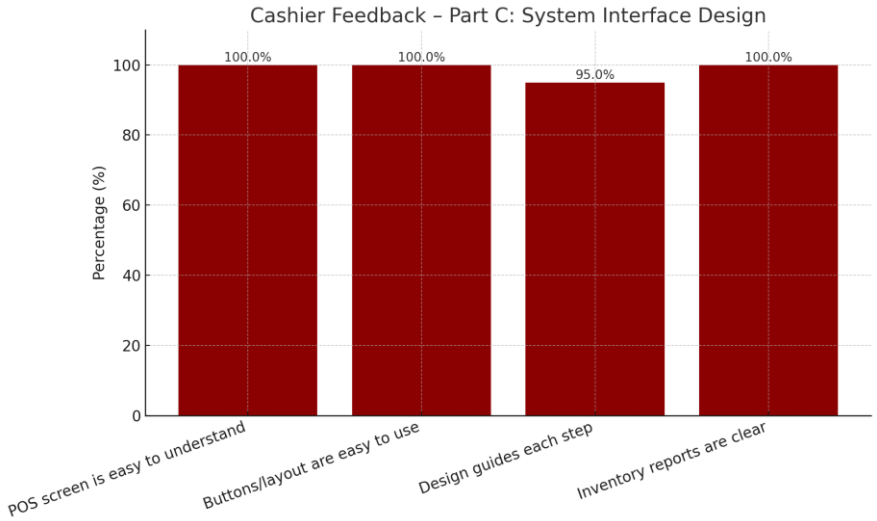



**Fig. 20** *UAT for Cashier: System Interface Design*

Fig. 20 shows the results for Part C, which looks at how easy the design and layout of the POS system are for cashiers to use. For system interface design performance, the POS page is easy to understand, inventory reports are clear and the layout is organized, earning 100% satisfaction. Furthermore, design guides for each step have a lower satisfaction percentage of 95%.

## 6. Conclusion

POS system for Maizan Solution applies Agile methodology and uses Laravel, Firebase, and machine learning algorithms like linear regression and random forest to deliver accurate stock forecasts, customizable discounting and improved inventory control. These features improve operational efficiency and customer experience by offering specific discounts and seamless transactions that ensure profitability and data-driven

decision-making. Three objectives have been achieved, which are to design a system based on an Object-Oriented Approach, to develop a web-based system and to perform user acceptance testing on the system.

Future works of this project will have collaboration with e-commerce platforms and third-party logistics providers as well as the use of advanced AI models to improve customer behaviour prediction and sales forecasting. These improvements would increase the system's scalability making it more suitable for larger retail venues and improving its adaptability in an evolving retail situation. This project is important for the retail industry because it helps store owners manage their daily tasks more easily, track inventory better and make smarter business decisions using real-time data.

## Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, for its support.

## Conflict of Interest

Authors declare that there is no conflict of interest regarding the publication of the paper.

## Author Contribution

The author confirms sole responsibility for the following: study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.

## References

[1] Jack, F., & Bommu, R. (2024). Unveiling the Potential: AI-Powered Dynamic Inventory Management in the USA. International Journal of Advanced Engineering Technologies and Innovations, 1(3), 241-261

[2] Hayes, K. (2024). POS systems: Your complete guide to success. Extend a Retail. https://www.extendaretail.com/resource/pos-systems-the-ultimate-guide/

[3] Star Micronics. (2024). What is artificial intelligence, and how does it impact your point-of-sale? Retrieved from https://starmicronics.com/blog/what-is-artificial-intelligence-and-how-does-it-impact-your-point-of-sale/

[4] Hashemi-Pour, C., & Wigmore, I. (2024). What are machine learning algorithms? 12 types explained. WhatIs. https://www.techtarget.com/whatis/definition/machine-learning-algorithm

[5] Date, C. J. (2003). *An Introduction to Database Systems* (8th ed.). Addison-Wesley.

[6] AutoCount Solutions Sdn. Bhd. (2025). *Point-of-Sale System Malaysia*. Retrieved from https://autocountsystem.com/pos-system-malaysia/

[7] IRS Technologies Sdn. Bhd. (2025). *Point-of-Sale System*. Retrieved from https://irs.com.my/possystem/

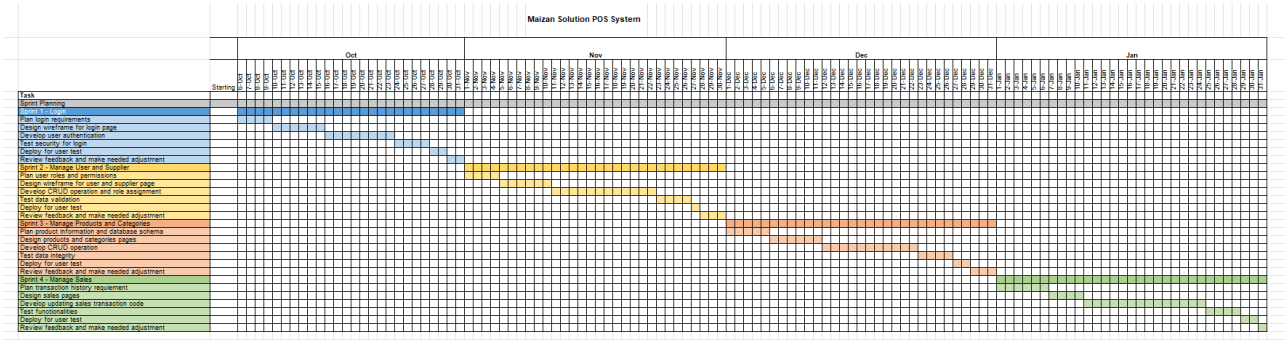
[8] Qube Apps Solutions Sdn Bhd. (2025). *QubePos – All-in-one restaurant & retail POS system*. Retrieved from https://qubepos.com/

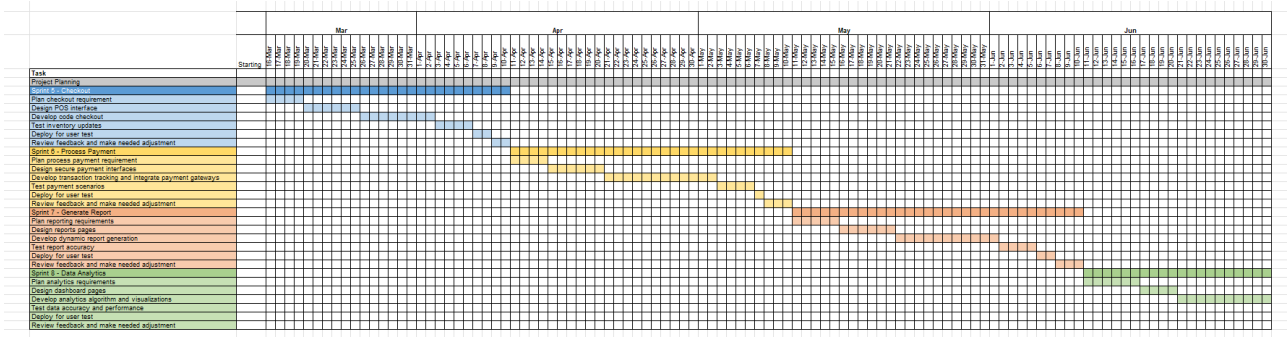
[9] Wangsa, K., Chugh, R., Karim, S., & Sandu, R. (2022). A comparative study between design thinking, agile, and design sprint methodologies. International Journal of Agile Systems and Management, 15(2), 225. https://doi.org/10.1504/ijasm.2022.124916

[10] Jethro Jeff. (2025). *["Agile Methodology"]*. Retrieved from https://jethrojeff.com/

[11] Rahman, M. M., & Ferdous, S. M. (2019). The benefits of MVC architecture in modern web development: A case study using Laravel. *International Journal of Computer Applications*, 178(7), 10–14.

## Appendix A


Maizan Solution POS System
Oct
Nov
Dec
Jan
Task
Sprint Planning
Sprint 1 - Login
Plan login requirements
Design wireframe for login page
Develop user authentication
Test security for login
Deploy for user test
Review feedback and make needed adjustment
Sprint 2 - Manage User and Supplier
Plan user roles and permissions
Design wireframe for user and supplier page
Develop CRUD operation and role assignment
Test data validation
Deploy for user test
Review feedback and make needed adjustment
Sprint 3 - Manage Products and Categories
Plan product information and database schema
Design products and categories pages
Develop CRUD operation
Test data integrity
Deploy for user test
Review feedback and make needed adjustment
Sprint 4 - Manage Sales
Plan transaction history requirement
Design sales pages
Develop updating sales transaction code
Test functionalities
Deploy for user test
Review feedback and make needed adjustment


Mar
Apr
May
Jun
Task
Project Planning
Sprint 5 - Checkout
Plan checkout requirement
Design POS interface
Develop code checkout
Test inventory updates
Deploy for user test
Review feedback and make needed adjustment
Sprint 6 - Process Payment
Plan process payment requirement
Design secure payment interfaces
Develop transaction tracking and integrate payment gateways
Test payment scenarios
Deploy for user test
Review feedback and make needed adjustment
Sprint 7 - Generate Report
Plan reporting requirements
Design reports pages
Develop dynamic report generation
Test report accuracy
Deploy for user test
Review feedback and make needed adjustment
Sprint 8 - Data Analytics
Plan analytics requirements
Design dashboard pages
Develop analytics algorithm and visualizations
Test data accuracy and performance
Deploy for user test
Review feedback and make needed adjustment