**A PROPOSED OFFERING OF**

**INVENTORY AND SALES**

**SYSEM FOR PANDAYAN SCHOOL SUPPLIE’S STORE**

A Project Proposal Presented to the

Faculty of Datamex College of Saint Adeline, Inc.

In Partial Fulfillment of the Requirements for the

Degree of Bachelor of Science in Information Technology

**Prepared By:**

Baquilar, Lovely B.

Ebona, Sheila D.

Tapitan, Reylyn B.

Tamparong, Meldrin

October 2025

**DESIGN DOCUMENT**

**INTRODUCTION**

The purpose of this document is to describe the system design of Inventory and Sales System for Pandayan’s School Supplies Store. This project is part of our academic requirement, but more importantly, it’s a practical solution to a real-world problem we observed in our local community.

The store currently uses manual methods to track inventory and record sales, which often leads to errors, delays, and confusion especially during busy school days. Through this system, we aim to automate those processes and provide a more organized, efficient, and user-friendly way to manage products and transactions.

Our system is designed as a desktop-based using Visual Basic and SSMS Access, making it accessible even without internet connection. It allows the store owner (Manager) to log products, monitor stock levels, and report sales. All within a single platform. We focused on simplicity and functionality, ensuring that users with basic computer skills can operate it smoothly.

This document includes the system’s architecture, database design, user interface layout, component breakdown, and security features. It also outlines how we plan to deploy and maintain the system after development. As students, we made sure to apply what we’ve learned in class from coding to proper documentation while keeping the system relevant and usable for its intended environment.

**SYSTEM ARCHITECTURE**

**Architecture Type**

* Desktop-based, Double-user system.

**High-level Components**

* Client-side (Desktop Interface for Manager and Cashier).
* Built using Visual Basic and SSMS Access.
* Allows user to log in, manage inventory, record sales, and history.

**Server-side (System Logic & Operations)**

* Let the Manager add new products to the system, stock updates, and transaction logging.
* Create total sales records.

**Database (SSMS Access)**

* Store product info, stock levels, user accounts, and sales records.
* Supports queries for reports and data retrieval.

**Deployment**

**Communication Protocols and Interfaces**

* The system uses direct database queries to perform CRUD operations (Create, Update, Delete).
* All interactions between the user and the database are handled internally through Visual Basic forms and modules.
* Input validation and error handling are built into the interface to ensue data accuracy and prevent system crashes.

**Database design**

* Products: (Product\_ID, categories, quantity, Unit\_price, Product\_name, Selling\_price )
* Sales: (ruler, crayons, pens, paper).
* User: (user ID, username, password, Full\_name).

**Relationship**

* One product can appear in many sales.
* One admin can log multiple transactions.

**Database Tables (based on SSMS Schema)**

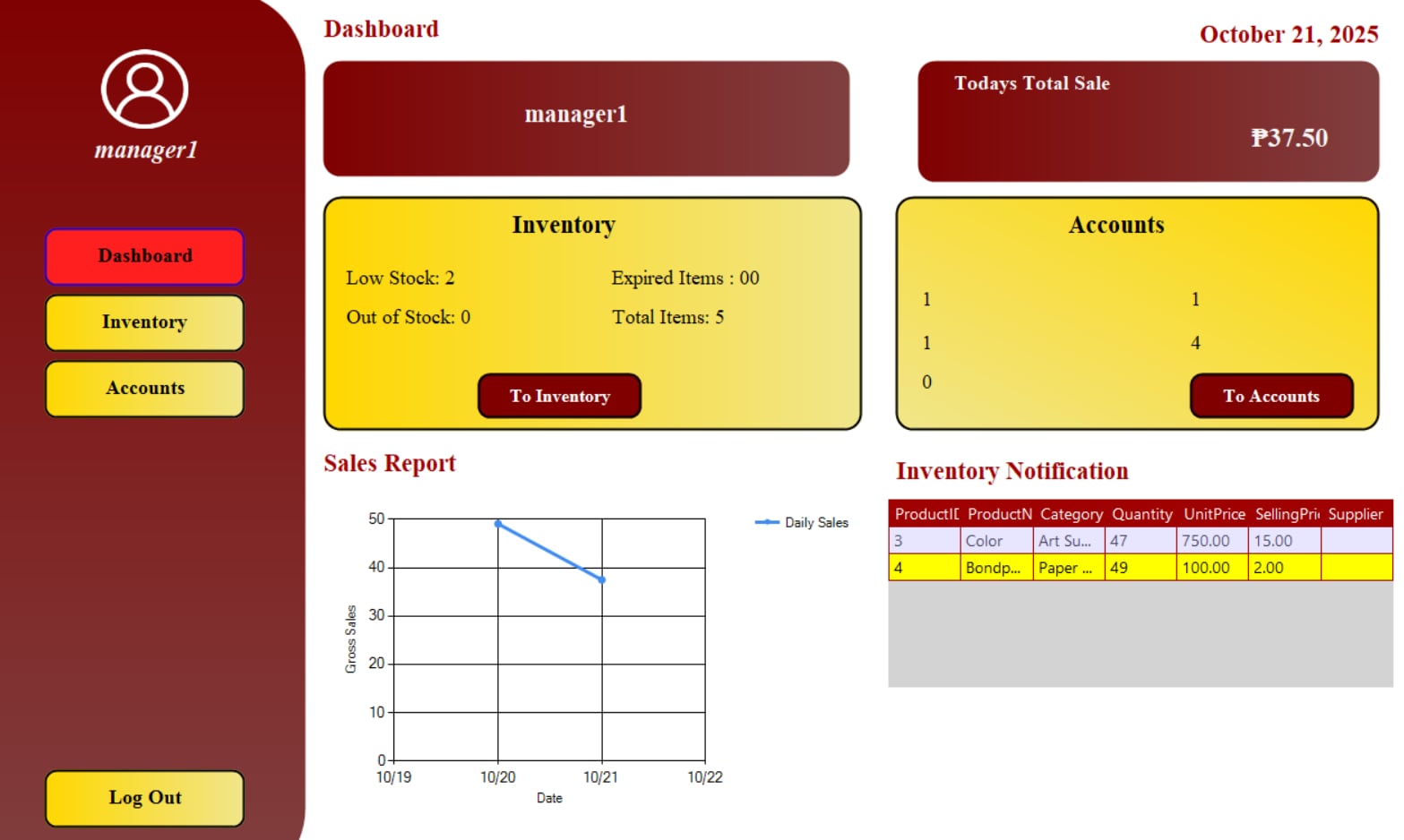
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Produc**  **t Id** | **Categorie**  **s** | **Produc t Name** | **Quantit**  **y** | **Pric e** | **Retaile d Price** | **Mar**  **k Up** | **Barcod**  **e** | **User**  **Nam**  **e** |
| **42** | Art Supplies | Color | 25 | 45 | 6 | 8 | NULL | NULL |
| **43** | Writing  Materials | Ballpen | 1 | 10 | 15 | 5 | NULL | NULL |
| **44** | Art Supplies | Crayon | 1 | 70 | 15 | 2 | NULL | NULL |

*Table 1: Table Name: Products*

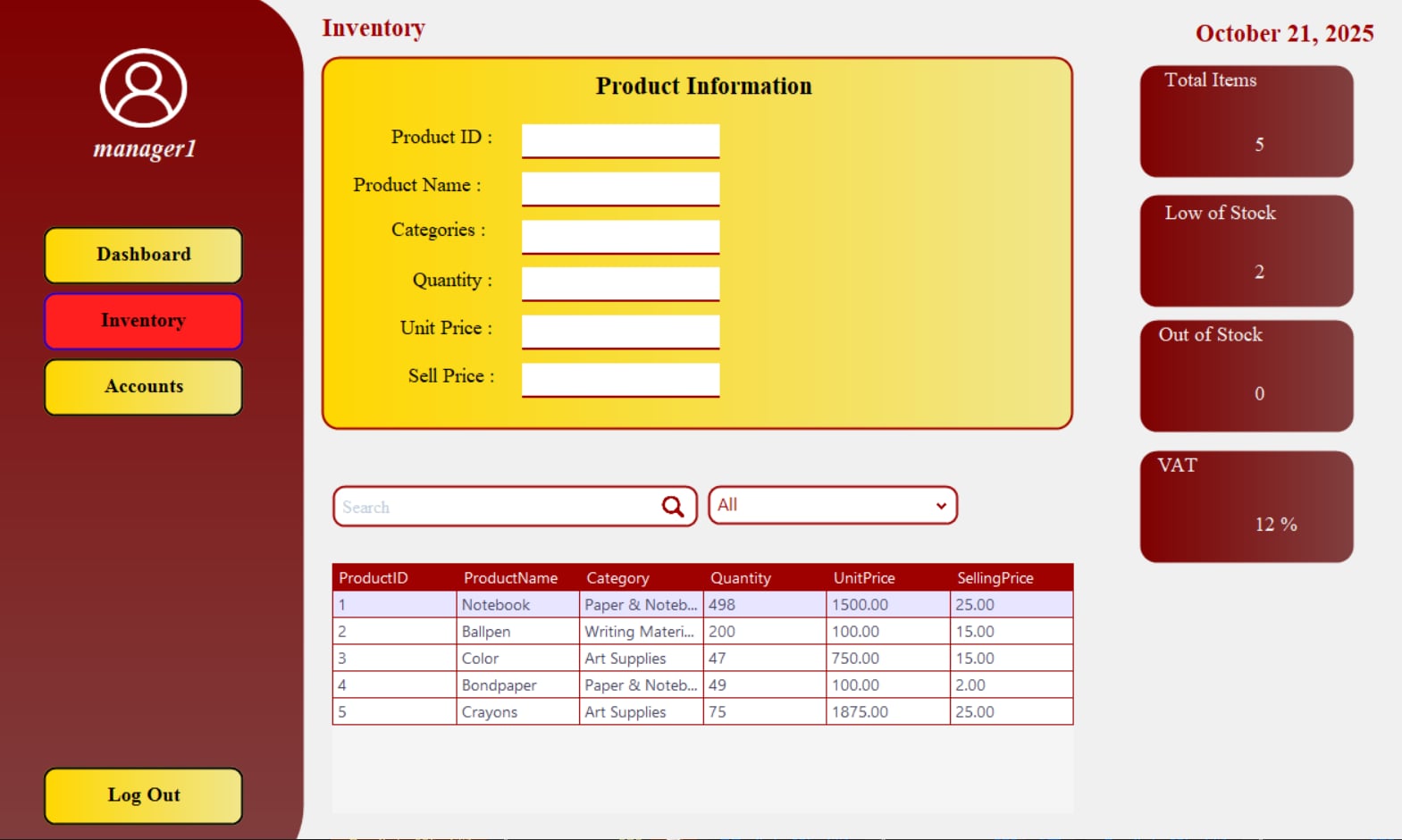
**USER INTERFACE DESIGN**

**Wireframe **

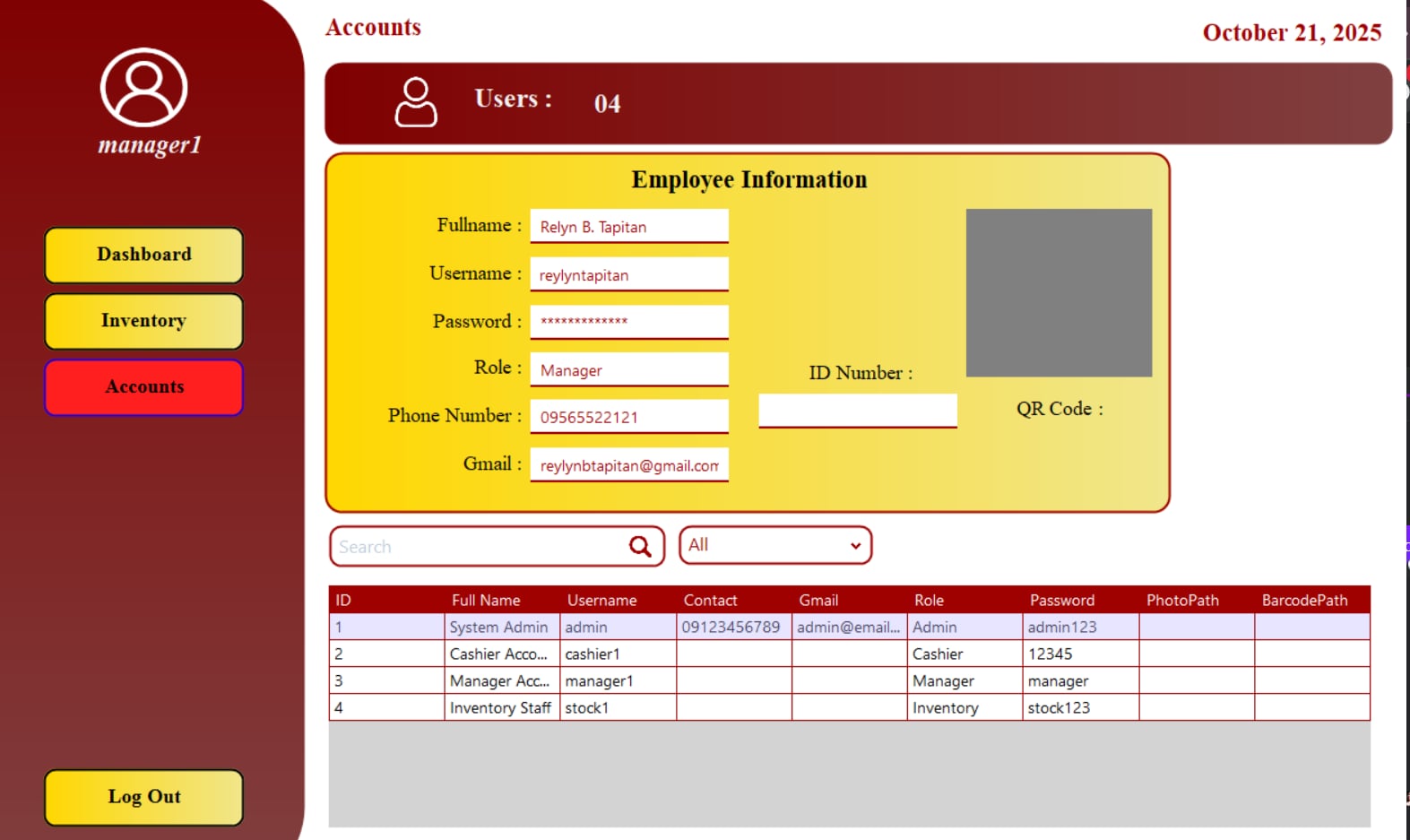
*Image 1: Manager Login Page*

****

*Image 2: Dashboard*



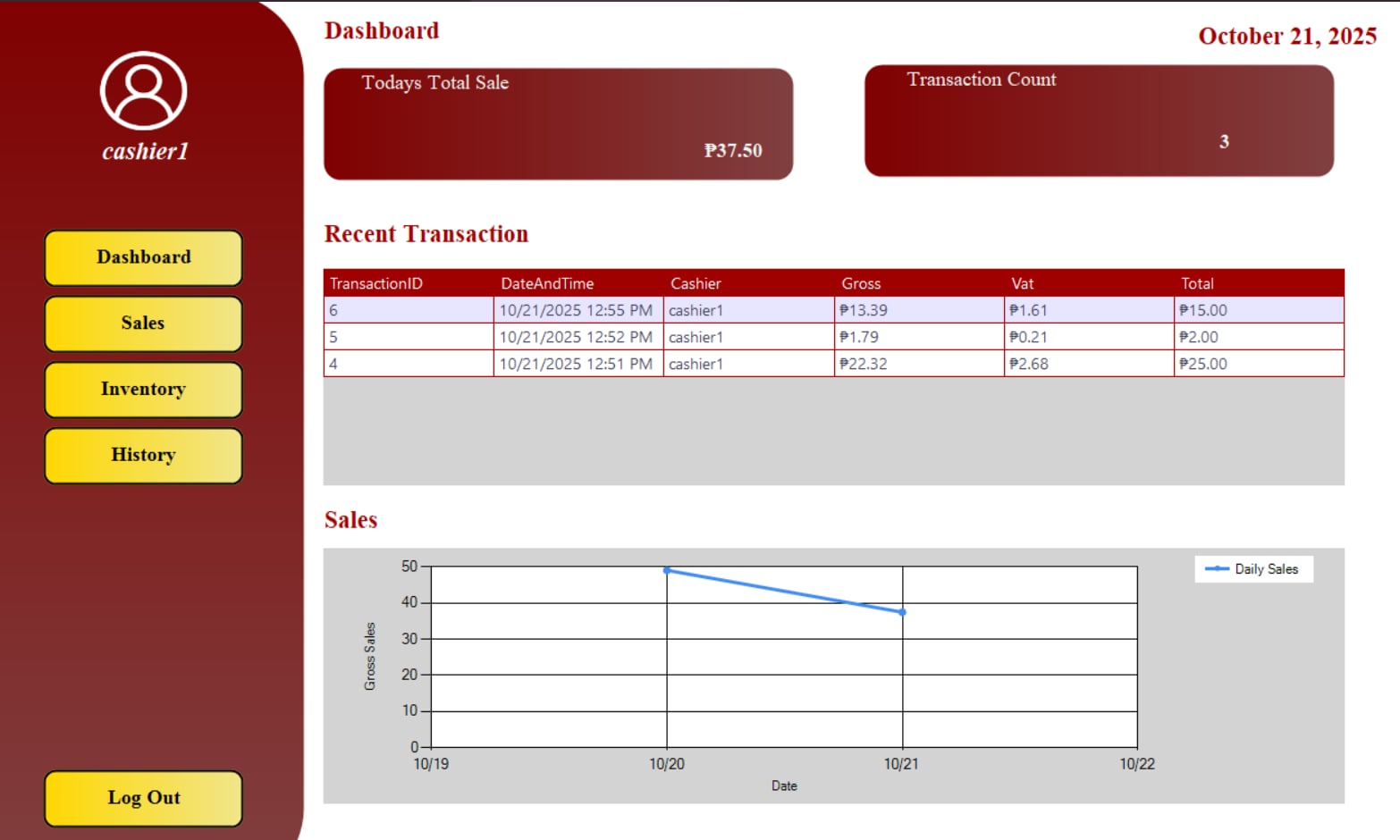
*Image 3: Inventory*



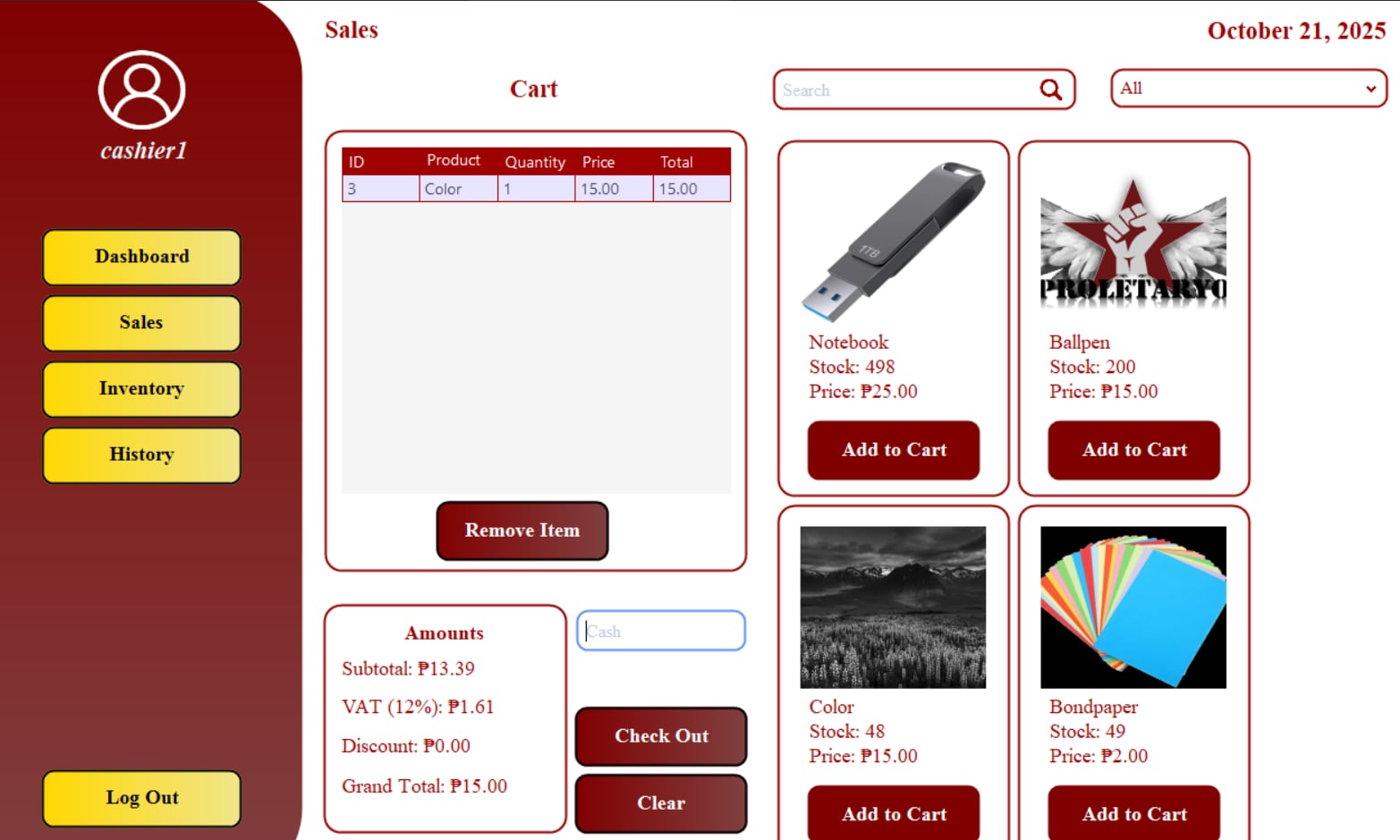
*Image 3: Accounts*

**Cashier Log in Form**

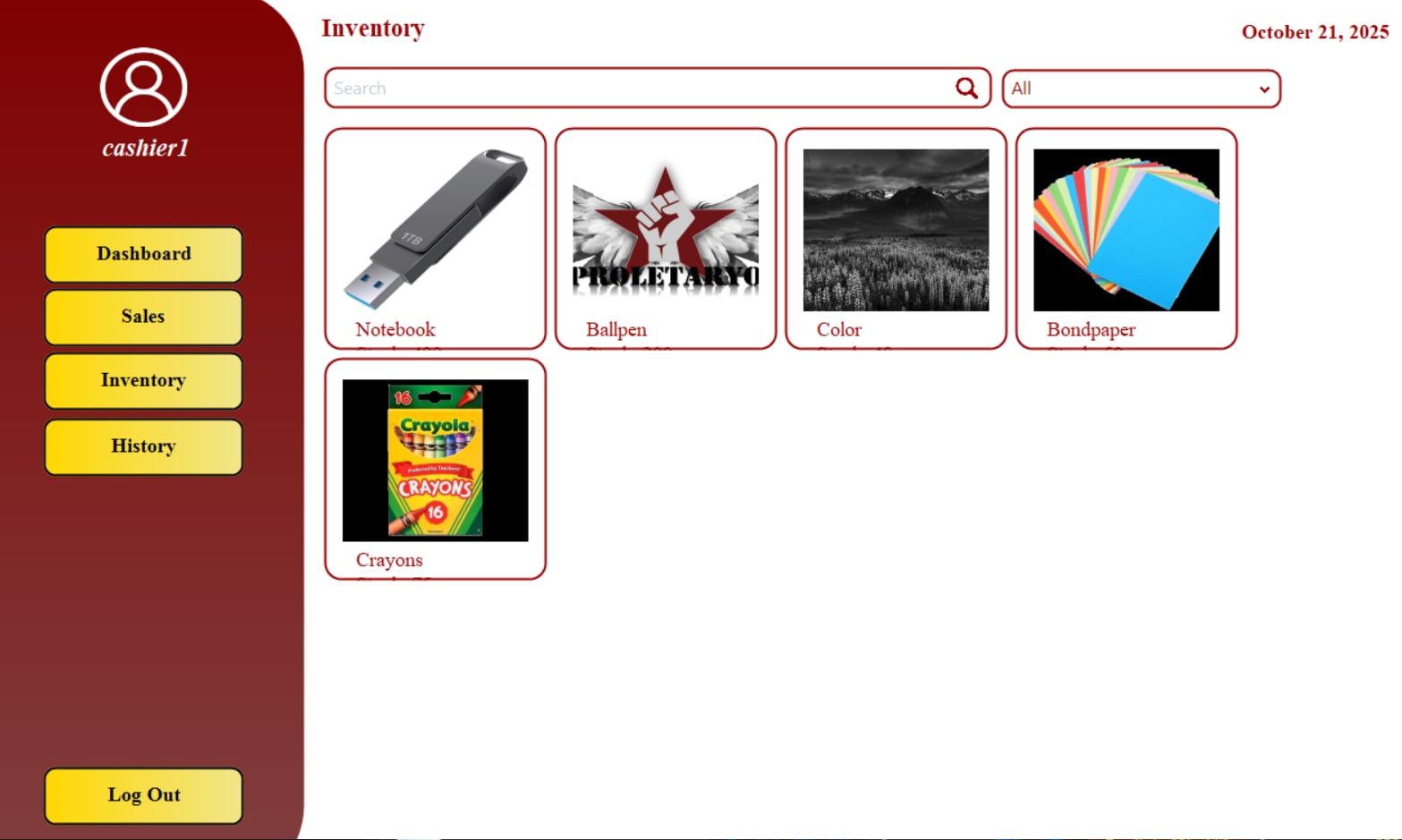
*Image 5: Log in Form*

****

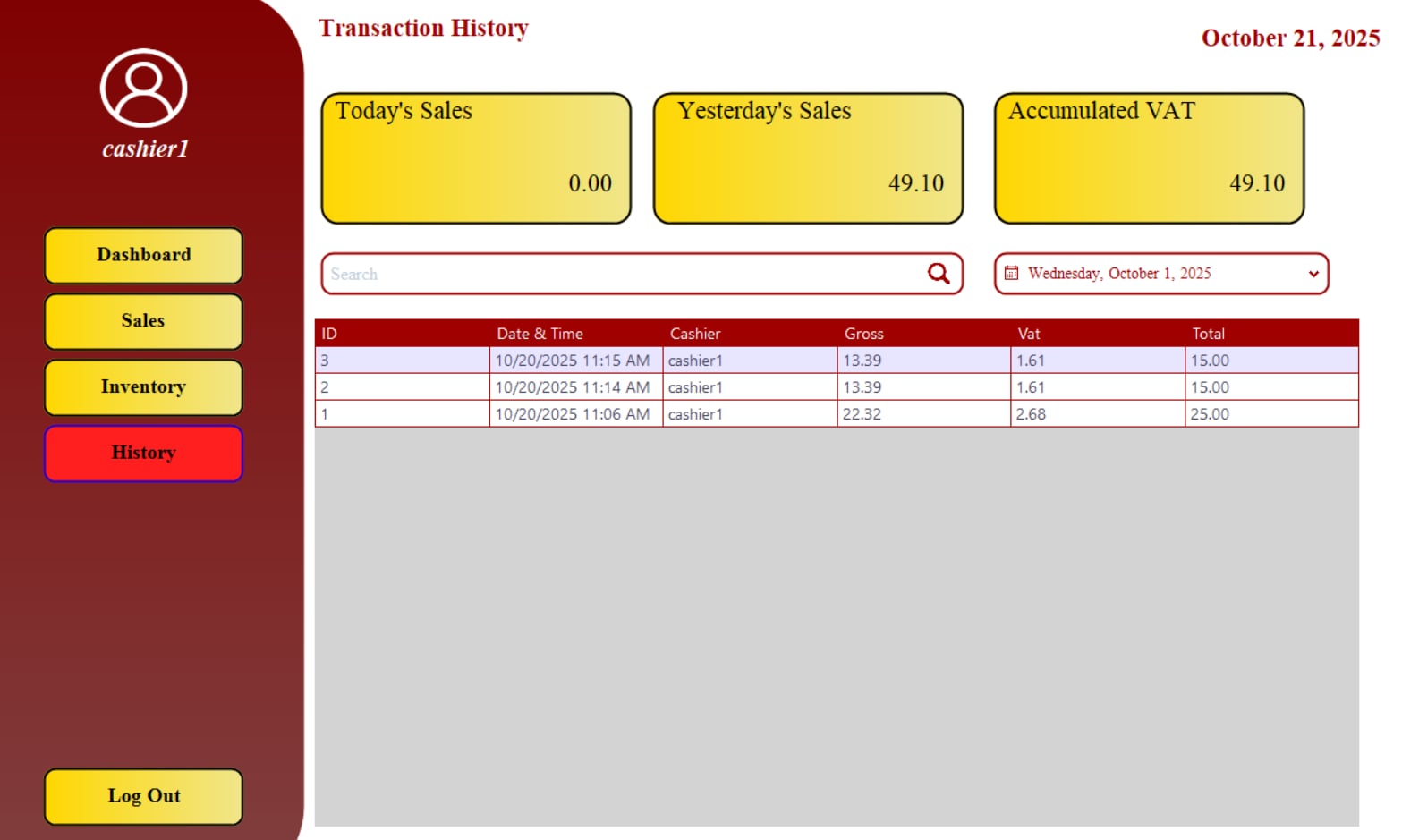
*Image 6: Bashboard*

****

*Image 7: Sales*

**

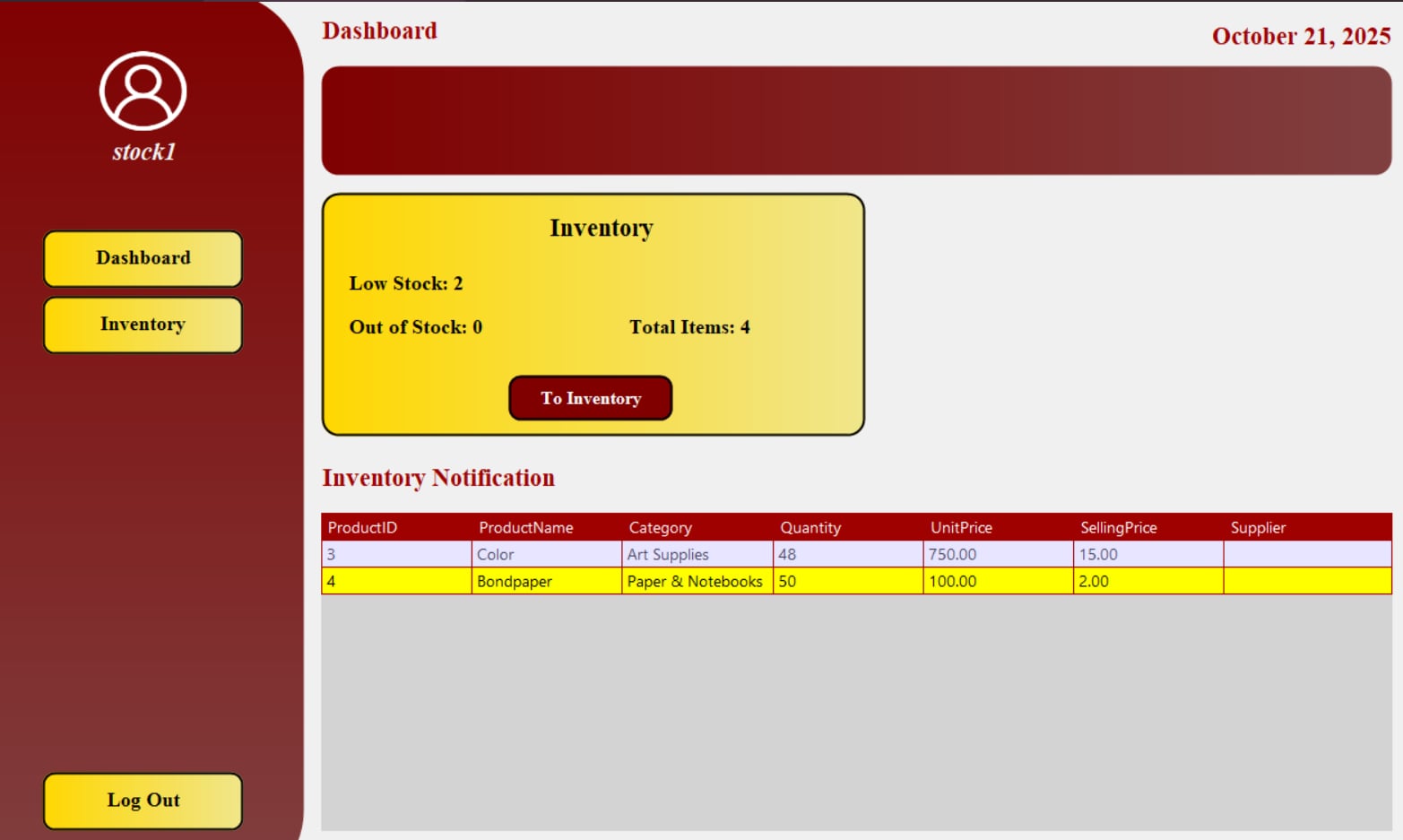
*Image 8: Inventory*

**

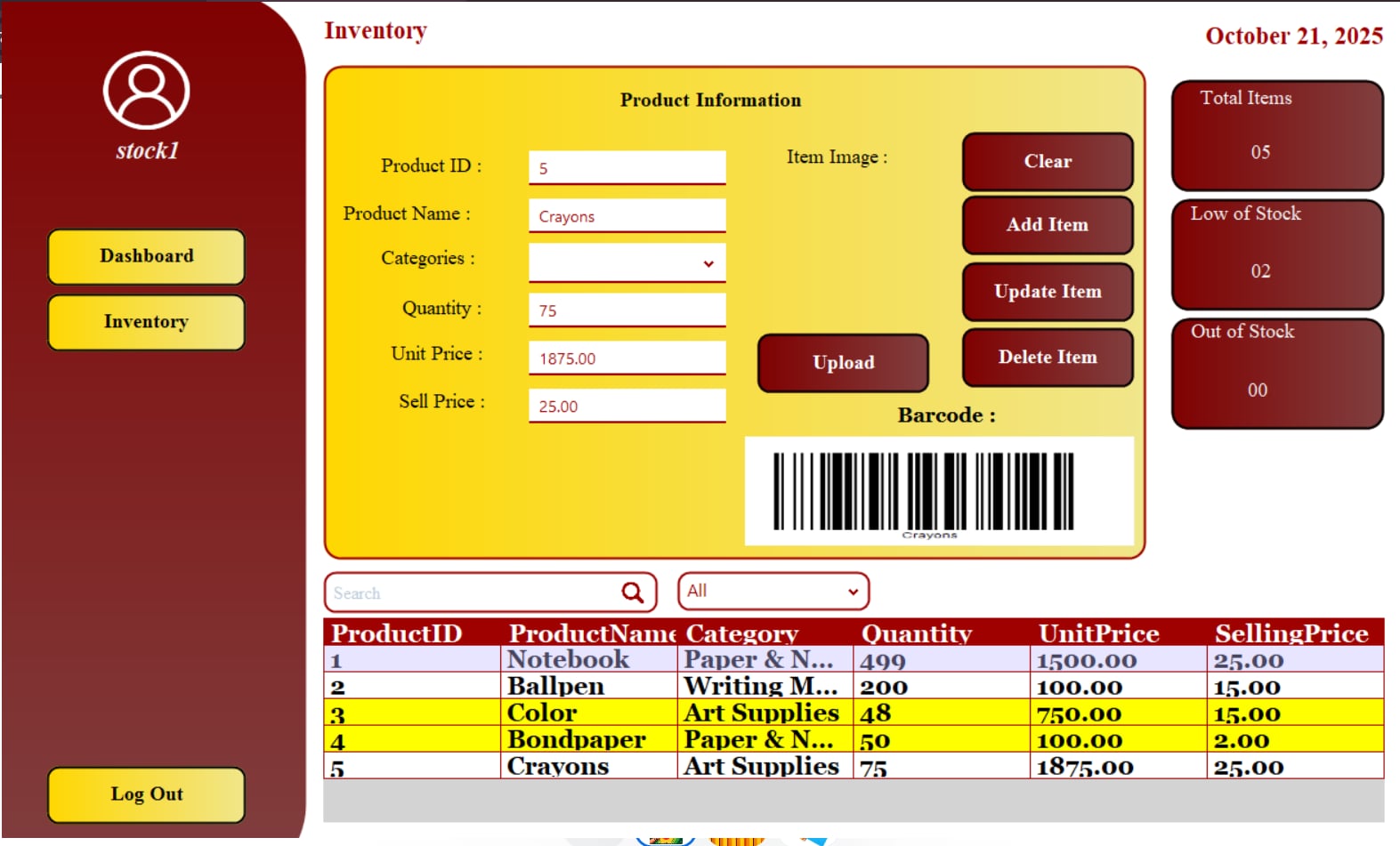
*Image 9: History*

**Inventory Account**

****

*Image 10: Inventory Acc*

*Image 11: Dashboard*

**

*Image 12: Inventory*

**COMPONENT DESIGN**

**Manager Side**

Log in Page - Secure login form for managers with role selection, username, and password input.

**Dashboard**

Displays current date and username. Shows daily sales summary, inventory status (low stock, out of stock, expired items), and total accounts. Includes a daily sales graph and a table listing inventory notifications. Sidebar buttons allow navigation to Inventory, Accounts, and Logout.

**Accounts Module**

Allows the manager to view and edit employee details including full name, username, password, role, contact number, and Gmail. Employee records are displayed in a searchable data grid view.

**Inventory Module**

Displays product details such as product ID, name, category, quantity, unit price, and selling price. Includes a search bar and inventory statistics (total items, low stock, out of stock).

**Cashier Side**

**Login Page** – Secure login form for cashiers with role selection, username, and password input.

**Dashboard** – Displays current date and username. Shows total sales for the day, transaction count, and a table of recent transactions with details like transaction ID, date, time, gross, VAT, and total. Includes a daily sales graph.

**Sales Module** – Allows the cashier to select products, add them to the cart, and process transactions. The cart table shows product ID, name, quantity, price, and total. VAT is auto-calculated, and cash input is provided.

**Transaction History** – Displays past transactions in a searchable table with filters by date. Shows summaries of today’s sales, yesterday’s sales, and accumulated VAT.

**Inventory Staff Side**

**Login Page**

Secure login form for inventory with role selection, username, and password input.

**Dashboard**

Displays current date and username. Shows inventory statistics such as total items, low stock, and out-of-stock counts. Includes a notification table listing products that need attention.

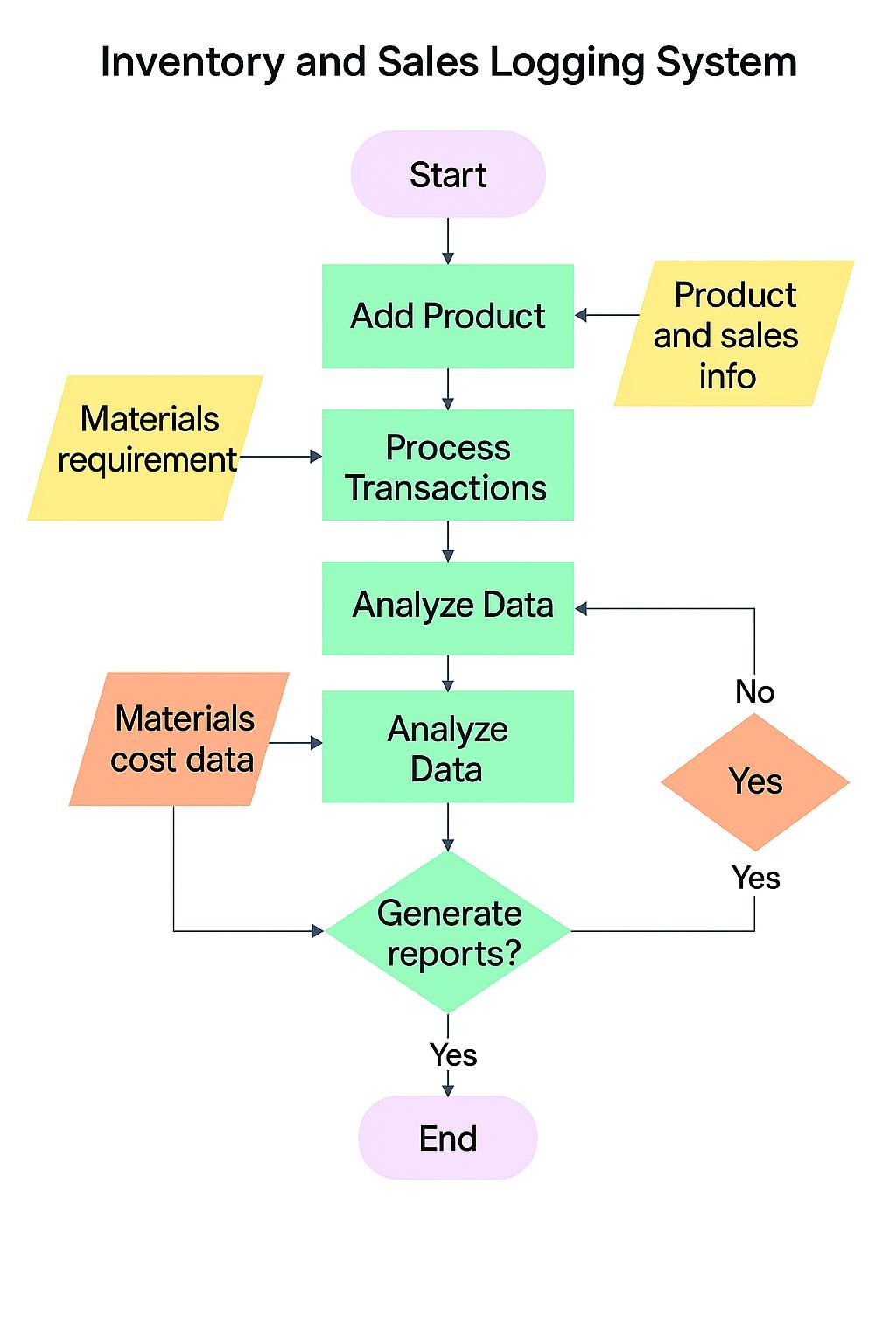
**Inventory Module**

Allows manager to add, update, delete, and clear product entries. The form includes fields for product ID, name, category, quantity, unit price, and selling price. Buttons are available for uploading images and generating barcodes.

**Inventory Table**

Displays all product details in a searchable data grid view. Helps manager monitor stock levels and maintain accurate records.

**DATA FLOW DIAGRAM**



*Image 6: Data Flow Diagram*

**Data Sources**

* Product and Sales Info - Details about each item (product ID, name, category, quantity, unit price, and selling price.)
* Materials Requirement - What supplies are needed to restock or fulfill orders.
* Materials Cost Data- Cost breakdown of each material or product used in transactions.

**Processing Logic**

1. Start

* System is activated or opened.

1. Add Product

* User inputs product details (product ID, name, category, quantity, unit price, and selling price.).
* System stores this in the product database.

1. Process Transactions

* System checks inventory, subtracts sold item.
* Uses “Materials Requirement” it flag low stock.

1. Analyze Data

* System reviews sales trends, inventory movements.
* Uses “Materials Cost Data” to calculate profit/loss.

1. Decision Point: Generate Reports? ▪ If YES

* System creates sales reports (inventory status).
* If NO
* Loop back to analyze more data or wait for next input.

1. End

* Process finishes or system is closed.

**Destination**

* Inventory updates stored in product inventory and sales log tables.
* Real-time stock status reflected in Manager Dashboard and Product Module.
* Restocking shown in Manage Stock.

**SECURITY DESIGN**

The Inventory and Sales System is designed to protect business data by using encrypted passwords on the login page. Access to the system is limited to the manager and cashier user, making sure that only the person in charge can manage inventory and sales records.

**Authentication and Authorization Mechanism:**

**Authentication**

* Admin log in using asterisk password.

**Authorization**

* Access to all system modules is exclusively granted to the manager account, ensuring that only authorized personnel can manage core functions such as dashboard, product management, and stock control.

## Data Encryption and Procedure Measures

* Passwords stored in the database are asterisk using encrypt to help protect user access.
* Database access is restricted to the authorized manager and cashier and inventory only.
* Session management is applied to prevent unauthorized use of the admin account.

**PERFORMANCE DESIGN**

**Performance Requirements and Objectives**

* The system must process basic transactions (adding products, logging sales) within 5minutes.
* The database can handle sales records smoothly and will keep working as long as the device has enough storage space.

**Strategies for Optimizing System Performance**

* **Optimized Queries:** SQL queries in SSMS Access are organized to speed up searching and filtering of products.
* **Data Normalization:** Database is normalized to reduce unnecessary data and improve efficiency.
* **Efficient UI:** Forms are designed with filters and pagination to prevent loading large data tables all at once.
* **Resource Management:** System closes unused forms and clears variables to save memory.

**Performance Testing Plan**

* **Load Testing:** Simulate 50+ transactions in a day to measure response time.
* **Stress Testing:** Fill database with 500 product records and measure query execution speed.
* **Scalability Testing:** Monitor performance when adding more than 5

concurrent sales in rapid succession.

**Error handling and logging**

* Input a validation on all forms ensures that users cannot leave required fields blank or enter incorrect information.
* Error messages to guide admin instead of system crashes.
* Automatic rollback if database operations fail during a transaction.

**Logging Requirements and Specifications**

* Maintaining error logfilerecording system errors with timestamp, user, and action attempted.
* Keep a sales log for every successful transaction including transaction ID, product, quantity, date.
* Track inventory changes including stock-in, stock-out, and adjustment for accountability.

**Error Codes and Messages:**

* Invalid Login (incorrect username/password).
* Insufficient Stock (user tried to sell more than available).
* Invalid Input (blank fields or wrong data type).
* Invalid Input (blank fields or wrong data type).
* Unauthorized Access (user tries to open restricted modules).

**Third-Party Integration**

**List of Third-Party Services or APIs Integrated**

* Barcode - For creating product barcode in the Product Management module.
* SSMS Access Database Engine For database connectivity.
* Integration Points and Data Exchange Formats Barcode**:** Exports barcode images as PNG/JPG to be printed.
* Database Connectivity: System communicates with SSMS Access using SQL queries via.
* Data exchange remains in structured database format includ**e** tables and queries.

**Deployment plan**

**Overview of the Deployment Process**

* Install SSMS Access runtime and required libraries on target computer.
* Copy system executable (VB project .exe) to local machine.
* Configure database path and ensure it is accessible by the application.
* Test all modules (login, dashboard, inventory, accounts, sales, and history).

**Hardware and Software Requirements for Deployment**

**Hardware**

* Acer Aspire Lite AL14-31P-36BE (Laptop)

**Software**

* Visual Basic .Net (VB.NET).
* SQL Server (for database).
* Microsoft Office (for documentation).

**Configuration Management and Version Control Procedures**

* The system’s source code was developed by referencing publicly available YouTube tutorials and instructional videos.
* While no formal version control repository (e.g., GitHub or Bitbucket) was used, manual tracking of code changes was performed using local file backups and naming conventions.
* Version naming followed a simple format (e.g., v1.0, v1.1, v1.2) to distinguish between updates and improvements.
* Regular backups of both the system executable file and the database were created and stored locally to ensure recovery in case of system failure or accidental deletion.

**MAINTENANCE AND SUPPPORT**

**Guidelines for System Maintenance**

* Backup of the SSMS Access database file.
* System performance monitoring during peak seasons.

**Procedures for Handling Updates, Patches, and Bug Fixes:** Bugs reported by users will be logged and tracked.

* Patches developed and tested on a backup database before deployment.
* Updated executable files distributed to users along with installation instructions.

**Escalation Process for Resolving Issues:**

* User identifies issue and reports to support team (student developers).
* Developer team investigates log files and database records.
* If unresolved, system patch or database repair is created.
* If persistent, escalate to advisor or IT support for advanced debugging.

## REVISION HISTORY

|  |  |  |
| --- | --- | --- |
| August 10, 2025 | Introduction, Client  Information, Project scope. | Revision of the consistent format and parts of the document follow the same structure, style, and layout. |
| August 12, 2025 | Project Approach, project team, project timeline. | Revision of not consistent font, and figure table, fix the simply words. |
| August 16,2025 | Project Resources, Risk Management, Communication  Plan. | Revision of the Format, fix the words that are deeply, creating tables |
| August 22,2025 | Project Governance, Approval, Appendix. | Fixing the words are deeply. |

*Table 2: Revision History*

**APPENDIX**

How to keep accurate inventory records? MRPeasy. Reviewed by P. J. Scott. Retrieved [date you accessed the article], from [https://www.mrpeasy.com/blog/inventory-records/](https://www.mrpeasy.com/blog/inventory-records/?fbclid=IwZXh0bgNhZW0CMTAAYnJpZBExUkJDdHhvZmVQNXM1dmV0dgEetAsXN5CwbHpWHJZCBB1EydnQM5KIjjnsjFPbrPYS9FUrg515q6NW0vm5gvw_aem_61jvQXb3-DjOWFrsLXT7KQ)

NetSuite. (2025, July). What is inventory management? Benefits, types, & techniques. NetSuite. Retrieved [date you accessed the article], from [https://www.netsuite.com/portal/resource/articles/inventory-management/inventory-management.shtml](https://www.netsuite.com/portal/resource/articles/inventory-management/inventory-management.shtml?fbclid=IwZXh0bgNhZW0CMTAAYnJpZBExUkJDdHhvZmVQNXM1dmV0dgEeDsEHNIsddQDzZWYJ5op-VYqkLvfMBudkdS4cmWoHL6bAGTIh0btznhmvlvI_aem_0yTDadR_R4Z-dkS5Rgeg4A)

Lightspeed. (2023). Retail inventory management: The retailer’s ultimate guide to inventory management. Lightspeed. Retrieved [date you accessed the article], from [https://www.lightspeedhq.com/blog/the-retailers-ultimate-guide-to-inventory-management/](https://www.lightspeedhq.com/blog/the-retailers-ultimate-guide-to-inventory-management/?fbclid=IwZXh0bgNhZW0CMTAAYnJpZBExUkJDdHhvZmVQNXM1dmV0dgEexLTajeKEr_c18sRtcklvvdxLfmuVCr5-AjSPMC2FfMnky30-ae-JNf_E-2M_aem_xX64HKjj4_tTnRXZMLeCJQ)

Hayes, A. Ph. D., CFA. (n.d.). Inventory management: Definition, how it works, methods & examples. Investopedia. Retrieved [date you accessed the article], from [https://www.investopedia.com/terms/i/inventory-management.asp](https://www.investopedia.com/terms/i/inventory-management.asp?fbclid=IwZXh0bgNhZW0CMTAAYnJpZBExUkJDdHhvZmVQNXM1dmV0dgEexnYJc1WzYhfVCzqhRHTgc38yhOw-2fkVIyf2kFnBeNFvVRnWT2lezt32fFw_aem_QFg60pnYI3ngzG5ax7WTdg)