Software Project Documentation

for

SupraMart - Digital Supermarket Management System

Version: 1.0

Prepared by: Group Name - MORZ

Members -

Kithupa Gajanayake	753383670V
Nimuthu Paranawithana	200418100551
Yasitha Nadeeshan	200434702670
Dehemi Fabbio	200619302440
Dinan Joseph	200431302810
Shenu Aturupana	200578102473
Sanduni Sewwandi	200259900582

Instructor: Mr.Anjana Dilhara

Course: H7E1 04

Batch: RHINO

Date: 17/08/2025

Acknowledgement

We, the members of **Team MORZ**, would like to express our deepest gratitude to everyone who supported and guided us throughout the successful completion of our final project, **"SupraMart – Supermarket Management System."**

First and foremost, we are sincerely thankful to our lecturers and academic mentors for their valuable advice, continuous encouragement and constructive feedback throughout the development of this project. Their guidance helped us understand the core principles of software engineering, system design, and team collaboration.

We also extend our heartfelt thanks to our university and department for providing the resources, learning environment and support systems necessary to plan, design and implement this project successfully.

Special thanks go to all the **members of Team MORZ** for their hard work, dedication, and strong team spirit during each phase of the project. Each member brought unique strengths to the team, and together we were able to build a fully functioning and scalable desktop application that addresses real-world supermarket operations.

Lastly, we appreciate the time and feedback from those who tested or reviewed the system during development. Their insights were crucial in refining SupraMart into a more user-friendly and reliable solution.

Thank you all for making this journey a successful one.

– MORZ Team

Abstract

SupraMart is a role-based desktop supermarket management system developed as our final project for Software Application Development (SAD). It's designed to streamline and automate the essential day-to-day operations of a supermarket, making tasks faster, more accurate and easier to manage. The system supports multiple stakeholders Cashiers, Inventory Managers, Administrators and Auditors each with clearly defined roles and access to their own dedicated interfaces.

Cashiers are responsible for front-end billing operations like barcode scanning, cart updates, applying discounts, processing payments and handling returns. Inventory Managers handle product-related data such as adding new items, updating stock levels, setting reorder points and tracking expiry dates. Administrators manage the overall system by creating user accounts, assigning permissions and performing backup and restore operations. Auditors are focused on generating reports and analyzing performance across different branches.

SupraMart includes 40 custom-built interfaces, covering everything from login and dashboards to billing, inventory management, reporting and user control panels. The main modules include Billing, Inventory, User Management and Reporting. Our system design started with the ER diagram and expanded through the creation of the SRS document, followed by sequence diagrams to guide development.

Built with scalability and user-friendliness in mind, SupraMart aims to improve operational accuracy, reduce manual errors and offer a smart, efficient experience for supermarket management.

Table of Contents

1. Introduction

1.1 Background and Need for the System

In today's fast-paced retail environment, supermarkets require robust digital solutions to manage high volumes of transactions, inventory data and staff activities. Traditional methods such as manual billing, handwritten stock records and non-integrated software lead to inefficiencies, human errors, stock mismanagement and suboptimal customer service.

With the increasing demand for automation and data-driven decision-making, there is a growing need for a centralized, digital supermarket management system that streamlines day-to-day operations, enhances accuracy and improves overall efficiency.

SupraMart aims to address these challenges by offering a fully integrated desktop-based platform that covers billing, inventory, user role management and reporting delivered through a user-friendly and intuitive interface.

1.2 Problem Statement

Many supermarkets continue to rely on outdated or partially automated systems that do not provide complete, end-to-end functionality for different roles within the organization. The key challenges faced include:

- Frequent manual errors in billing and inventory tracking
- Absence of real-time inventory updates
- Inefficiency in applying discounts or processing returns
- Lack of role-based access control for staff
- Difficulty in generating detailed reports for audits and analysis

- No mechanisms for reliable data backup and restoration

These limitations result in lost revenue, stock wastage, poor audit trails and overall operational inefficiency. A unified system that integrates all these features is essential for modern supermarket operations.

1.3 Objectives

The primary objectives of the SupraMart system are to:

- Develop a robust digital supermarket management system with 40 functional user interfaces
- Support role-based access for Admins, Cashiers, Inventory Managers and Auditors
- Enable barcode-based billing and real-time cart management
- Facilitate inventory tracking, low-stock alerts, and expiry management
- Generate detailed and exportable reports for audits and performance reviews
- Provide secure login, data backup, and system restoration capabilities
- Improve both customer experience and back-end operational workflow within the supermarket
- Enhance scalability and future-readiness for expansion into cloud integration or mobile applications

1.4 Scope of the Project

SupraMart is designed as a comprehensive desktop-based solution tailored for small to medium-sized supermarkets. It focuses on automating and managing key operational areas:

- Billing System: Barcode scanning, cart modification, discount application and payment processing
- Inventory Management: Product addition/editing, stock monitoring, expiry alerts and reorder level settings
- User & Role Management: Admin-controlled user registration, permission assignment and role-based access
- Reporting & Auditing: Interfaces for auditors to generate, review and export branch and sales reports
- Backup & Restore: Secure, Admin-only access to perform database backups and restoration
- Interface Coverage: A total of 40 well-designed interfaces to support role-specific workflows and features

The system currently excludes features such as online ordering, supplier integration and customer-facing mobile applications. However, it is built with scalability in mind, allowing for future enhancements in these areas.

1.5 Project Overview

SupraMart is a digital supermarket management system designed with a strong focus on modularity, scalability and user-friendliness. The system is structured to support four main stakeholders: Cashier, Inventory Manager, Admin and Auditor each assigned specific responsibilities with controlled access based on their role.

With a total of 40 custom-built interfaces, the system covers all essential supermarket operations ranging from login and billing to inventory management, user control, report

generation and data backup/restore. Every screen is crafted to match the exact needs of its user type, helping ensure smooth daily operations.

The project follows modern software engineering practices, incorporating ER diagram modeling, SRS documentation and UML diagrams to plan and design effectively. Testing phases are also included to ensure functionality and reliability before deployment.

By unifying all operations into one platform, SupraMart aims to reduce manual workload, increase billing accuracy and offer meaningful data insights for better decision-making, creating a smarter, more efficient retail experience for supermarkets.

---

2. Literature Review/Existing System

2.1 Comparison with Existing Supermarket Systems and POS Applications

Many supermarket management systems and POS (Point-of-Sale) applications are already available in the market. However, most of them either come with limited features or require costly subscriptions to unlock advanced tools. Below is a comparison between SupraMart and existing solutions:

Feature	Traditional POS Systems	Modern Retail Software	SupraMart
Barcode Scanning	Basic scanning, no real-time updates	Advanced scanning with inventory sync	Real-time scanning with instant cart updates
Inventory Management	Manual stock tracking, no alerts	Basic tracking with alerts	Live updates, AI-powered alerts, expiry tracking

User Roles & Permissions	Limited customization	Role-based access control	Detailed permissions for each stakeholder
Reporting & Analytics	Basic sales reports, no export	Custom reports with PDF export	Multi-format export: PDF, Excel, CSV
Backup & Restore	Manual backups only	Cloud backup (premium feature)	Automated daily backup, local & cloud
Multi-Branch Support	Not available	Enterprise version only	Built-in branch comparison tools
Offline Functionality	Fails without internet	Partial offline access	Fully offline-capable with sync on reconnect
Customer Management	Basic details only	Loyalty points & CRM	Optional feedback system and purchase history

Key Observations:

- Most POS systems prioritize billing, neglecting inventory and reporting tools
- Modern software often hides critical features behind paid plans
- Very few platforms offer a truly unified system that includes billing, inventory, user roles, and reporting under one solution

2.2 Limitations of Current Systems

a) Lack of Real-Time Inventory Sync

- Stock updates are delayed or require manual input, increasing chances of overselling
- Small-scale POS systems often depend on physical checks for accuracy

b) Poor User Role Management

- Limited roles usually just Admin and Cashier
- No separate access control for roles like Inventory Manager or Auditor

c) Limited Reporting Capabilities

- Only basic sales summaries are available
- Export formats are restricted, often to PDF only

d) No Backup & Restore Functionality

- Data loss risks are high in the absence of auto-backups
- Some systems rely on third-party cloud services with added costs

e) No Offline Mode

- Internet dependency causes interruptions during outages
- Unsynced transactions may be lost permanently

f) Poor Handling of Discounts & Returns

- Manual price overrides are needed for discounts
- Returns and exchanges are not efficiently tracked or logged

2.3 How SupraMart Improves on Existing Systems

a) Unified Platform for All Stakeholders

- Cashiers benefit from fast barcode scanning, real-time cart updates and smoother billing
- Inventory Managers can manage stock efficiently, receive low-stock alerts and handle bulk imports
- Admins have full control over user accounts, data backups and security settings
- Auditors get access to detailed reports, can compare across branches and export in multiple formats

b) Real-Time Inventory Management

- Instant stock updates after transactions
- Automated alerts prevent stock-outs and expired product issues

c) Advanced Security & Permissions

- Role-Based Access Control for every user type
- Sensitive actions like backup/restore are protected from unauthorized access

d) Comprehensive Reporting

- Detailed sales trends, inventory movement and profit analysis
- Reports can be exported as PDF, Excel, or CSV for deeper insights

e) Offline-First Design

- System works even without internet
- Syncs all data automatically once the connection is restored

f) Automated Backups

- Daily backups stored locally or optionally in the cloud
- One-click restore reduces downtime and data recovery effort

g) Improved Discount & Return Handling

- Predefined discount rules (member-based, seasonal offers, coupons)
- Easy and trackable returns/exchanges with audit logging

---

3. System Requirements

3.1 Functional Requirements:

Login Module

- FR1: System shall allow login for Admin, Cashier, Auditor, and Inventory Manager
- FR2: Login must be secured and password-protected

Dashboard

- FR3: Role-based dashboard should load upon successful login

Billing (Cashier)

- FR4: Scan product using barcode scanner to fetch product details
- FR5: Add, remove, or edit items in the cart
- FR6: Apply discounts either manually or using a code
- FR7: Select and process payment via cash or card
- FR8: Print receipt after successful transaction
- FR9: Handle product returns and exchanges with reason logging

Inventory Management

- FR10: Add new products with barcode, name, price, quantity, and expiry date
- FR11: Edit or delete product details
- FR12: Track and highlight low stock items
- FR13: Set and monitor reorder levels per product
- FR14: Alert and manage expired or soon-to-expire products

User Management (Admin)

- FR15: Create, edit, and delete user accounts
- FR16: Assign specific roles (Admin, Cashier, Auditor, Inventory Manager) and permissions

Backup & Restore (Admin)

- FR17: Manually trigger database backups or schedule automatic backups
- FR18: Restore system database from a selected backup file

Reporting (Auditor, Admin)

- FR19: Generate sales reports with filters (daily, weekly, monthly)
- FR20: Generate inventory reports showing stock levels and product details
- FR21: Export all reports to PDF and Excel formats

- FR22: View and compare multi-branch performance (if multi-branch module is implemented)

Customer Management

- FR23: Register customers to a loyalty or membership system
- FR24: Search, view, and manage registered customer profiles

3.2 Non-functional Requirements:

Performance

- NFR1: System must respond to user actions within 2 seconds
- NFR2: Must support at least 10 concurrent users and 100+ product scans without delay

Usability

- NFR3: The interface should be clean and intuitive, with tooltips and labels
- NFR4: Input fields should be validated with clear error messages

Reliability

- NFR5: Daily automatic backups should be maintained to avoid data loss
- NFR6: System should handle recovery gracefully during failure scenarios

Security

- NFR7: All passwords must be stored using encryption
- NFR8: Role-based access control should restrict module visibility and functions

Maintainability

- NFR9: System should be designed in a modular fashion to allow easy updates or feature additions

3.3 Hardware Requirements:

- Barcode Scanner

Used by cashiers to quickly scan product barcodes during billing, enabling faster checkout and reducing manual entry errors.

- Receipt Printer

  Prints itemized receipts for customers after each transaction.

3.4 Software Requirements:

- Java Runtime Environment

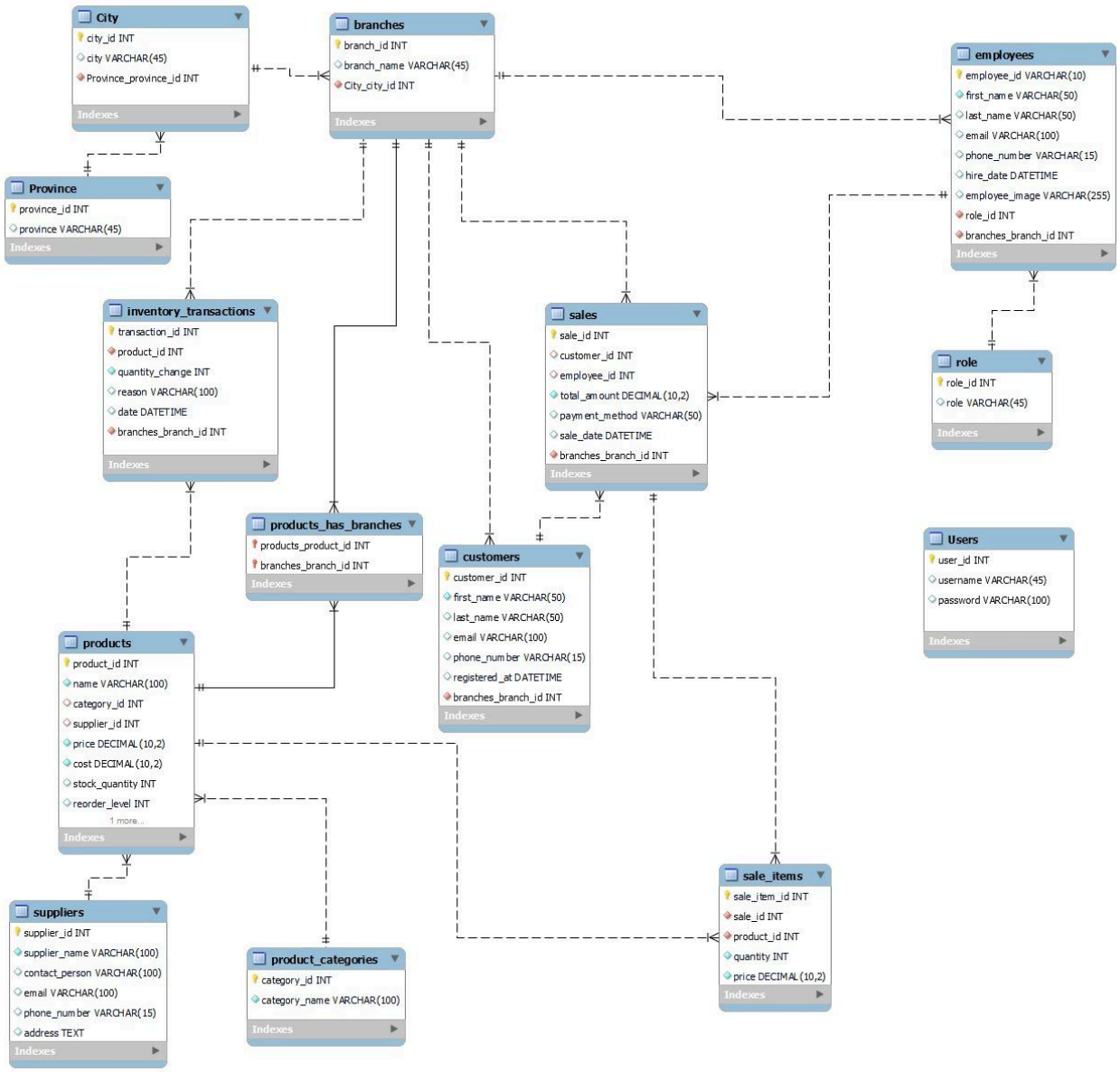
  Required to execute the application, especially if the GUI is developed using Java Swing

- MySQL Connector (JDBC)

  A driver that enables the application to communicate with the MySQL database.

4. System Design

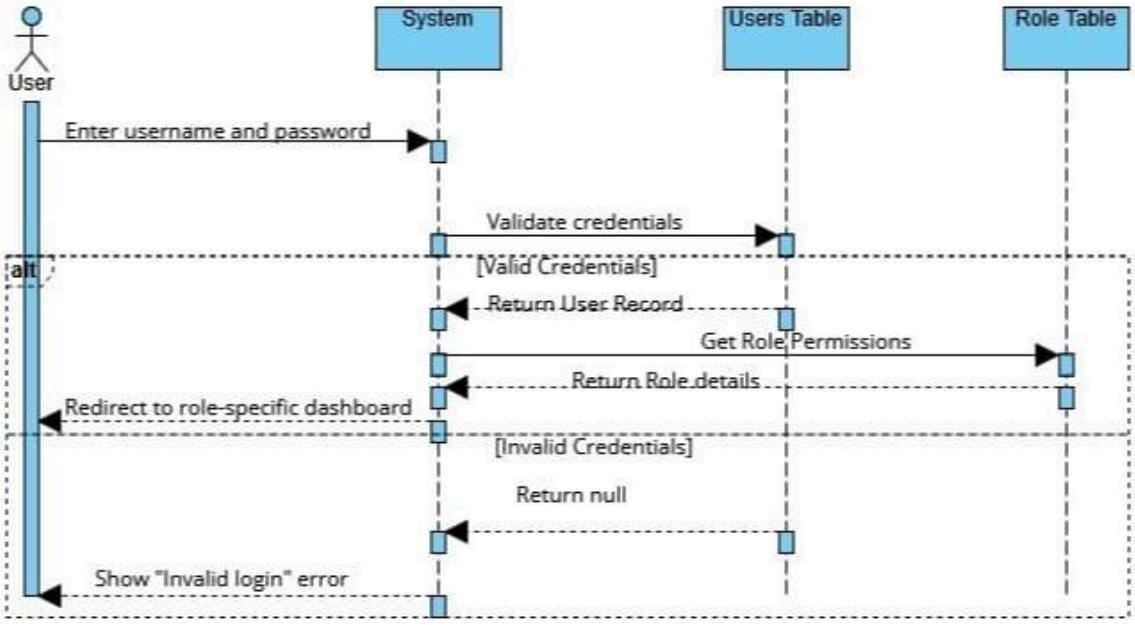
4.1 ER Diagram



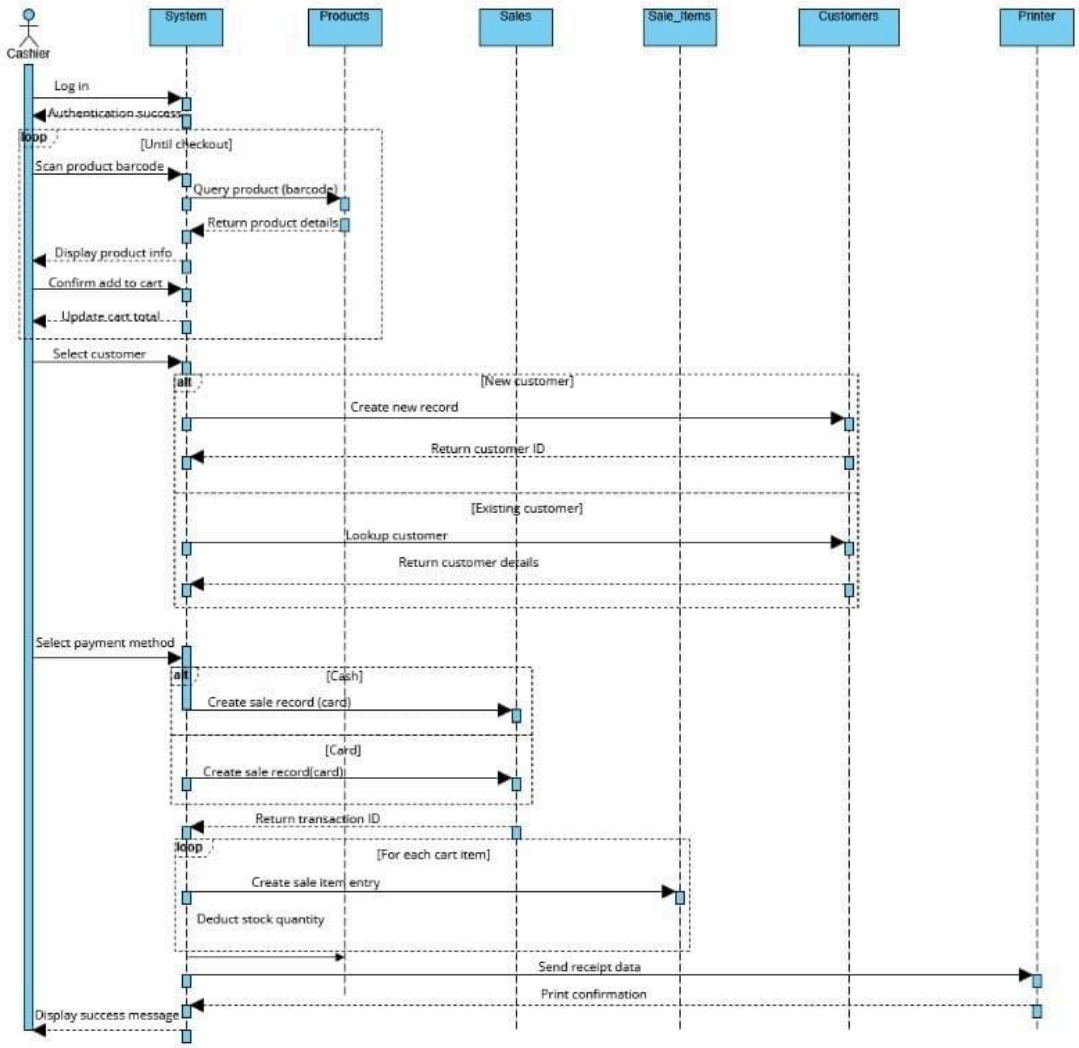
4.2 UML Diagrams:

- **Sequence Diagram**

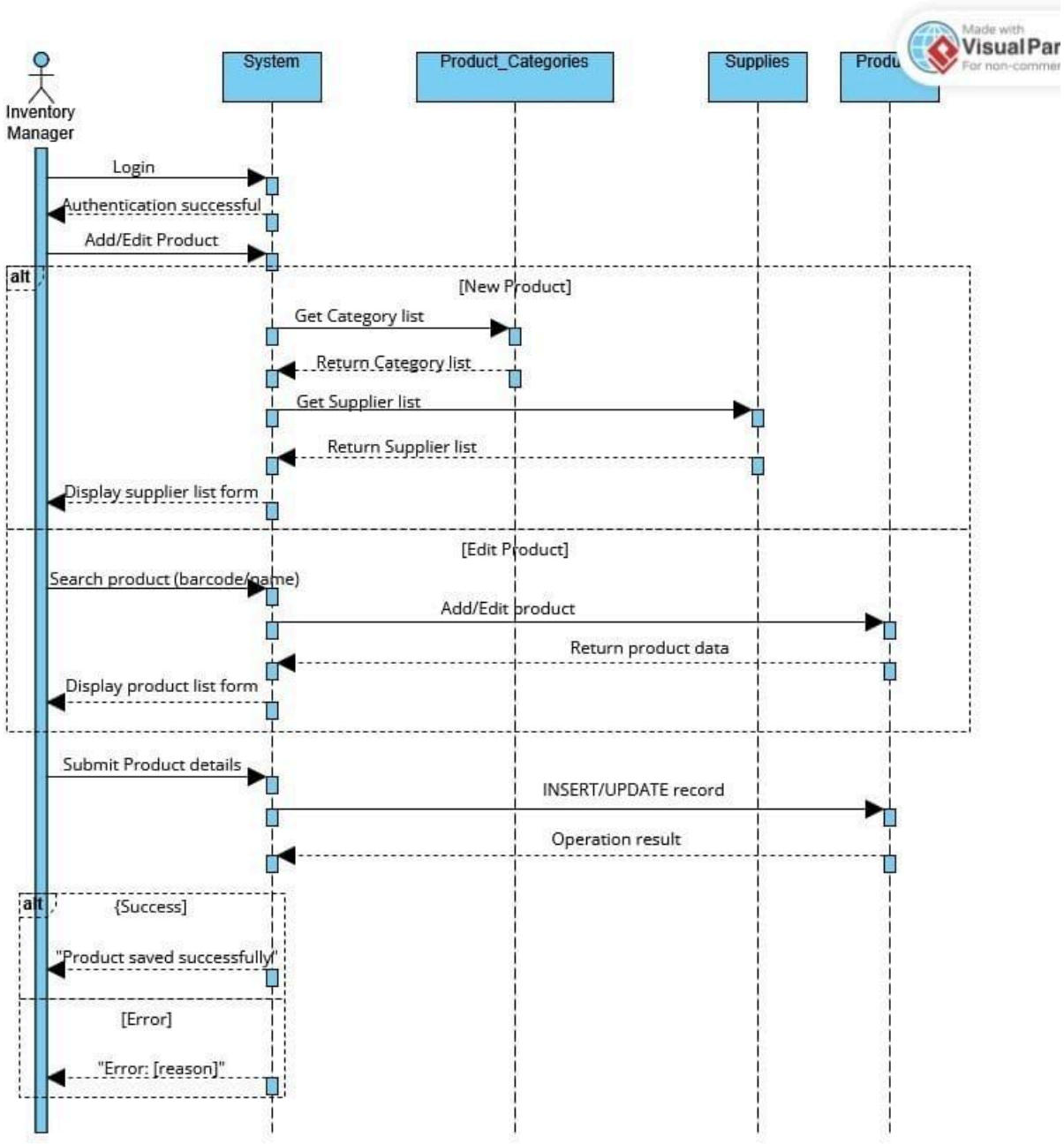
a) Login and Role Authentication



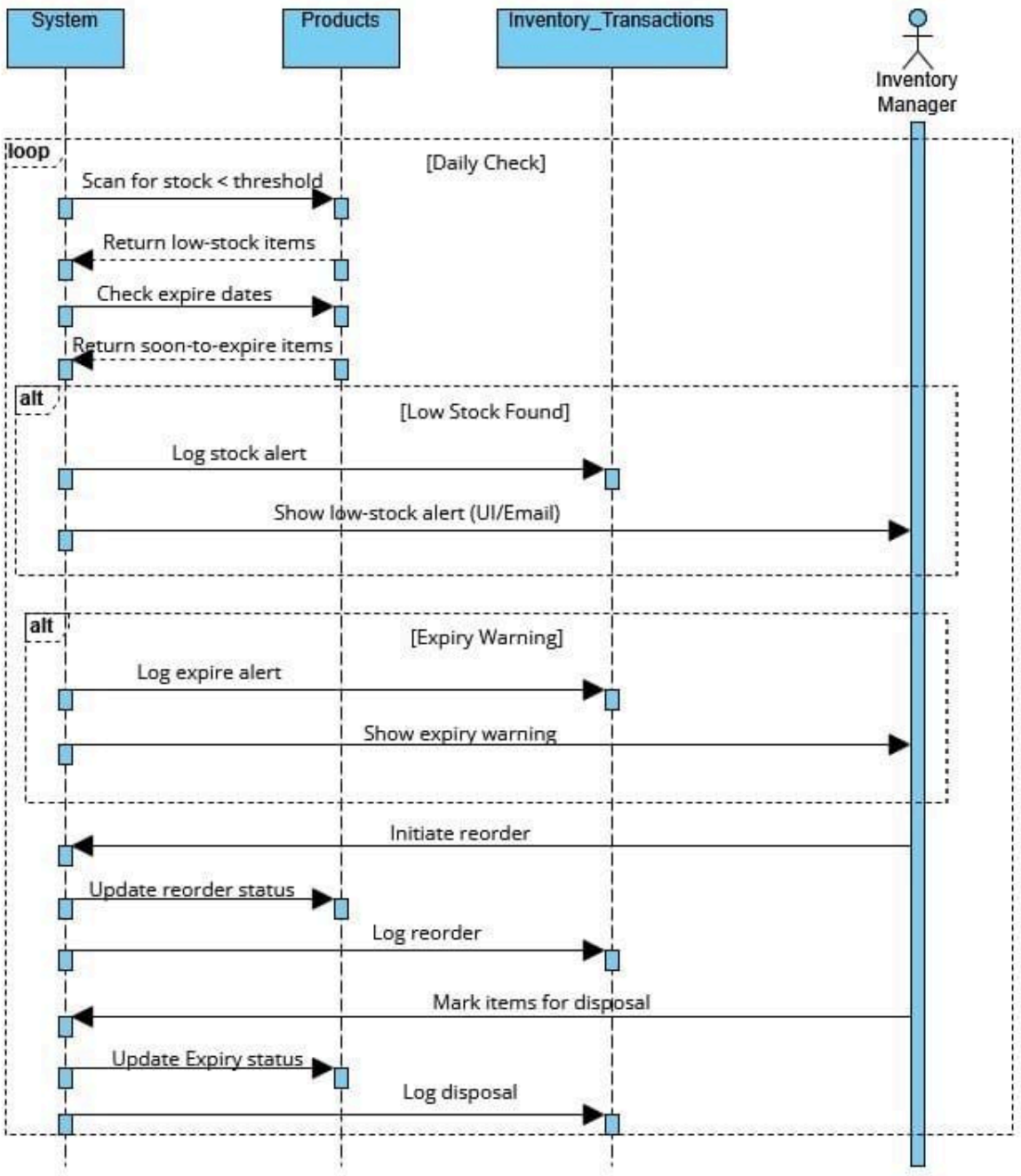
b) Billing & Checkout Process



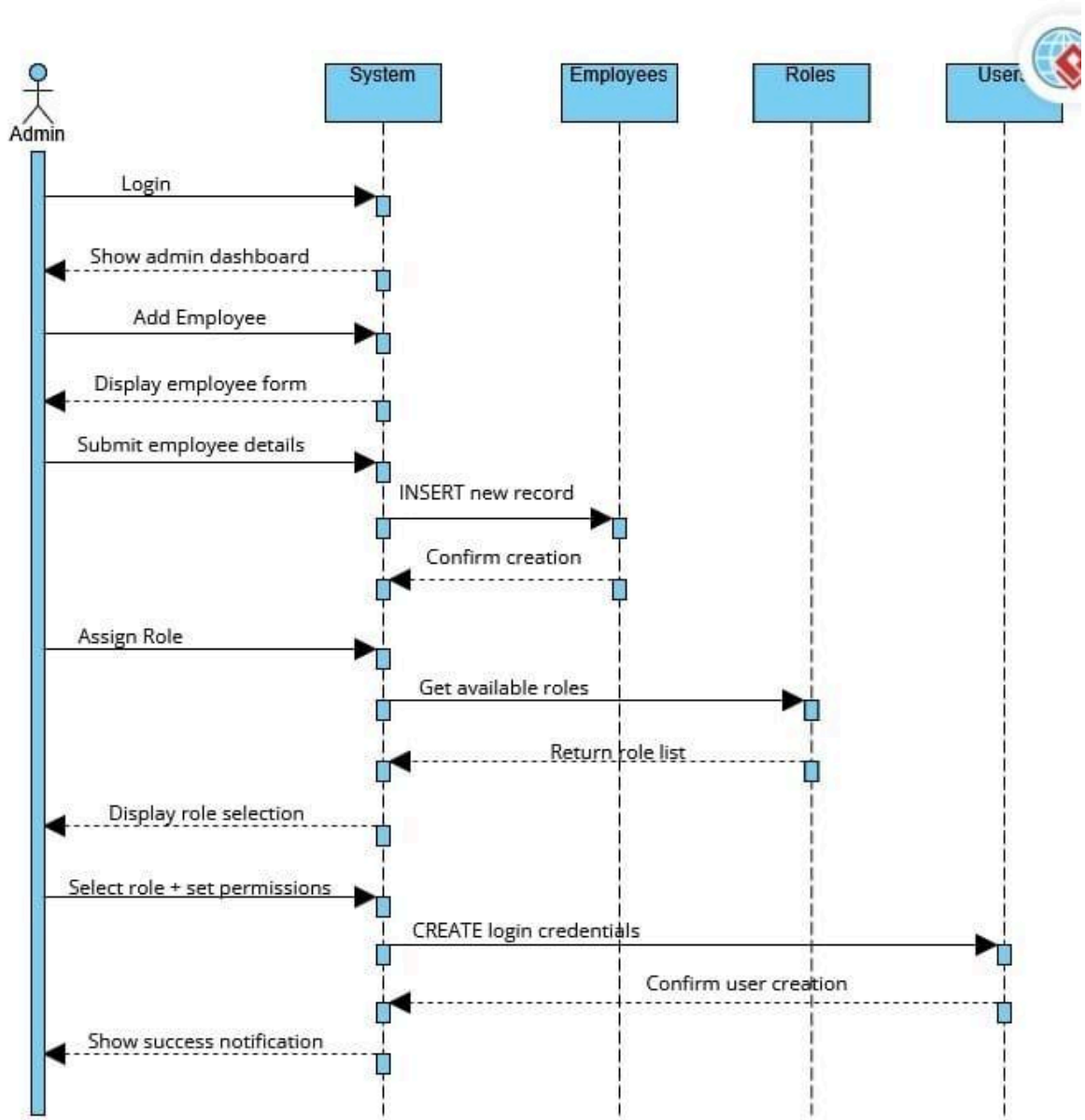
c) Inventory Management



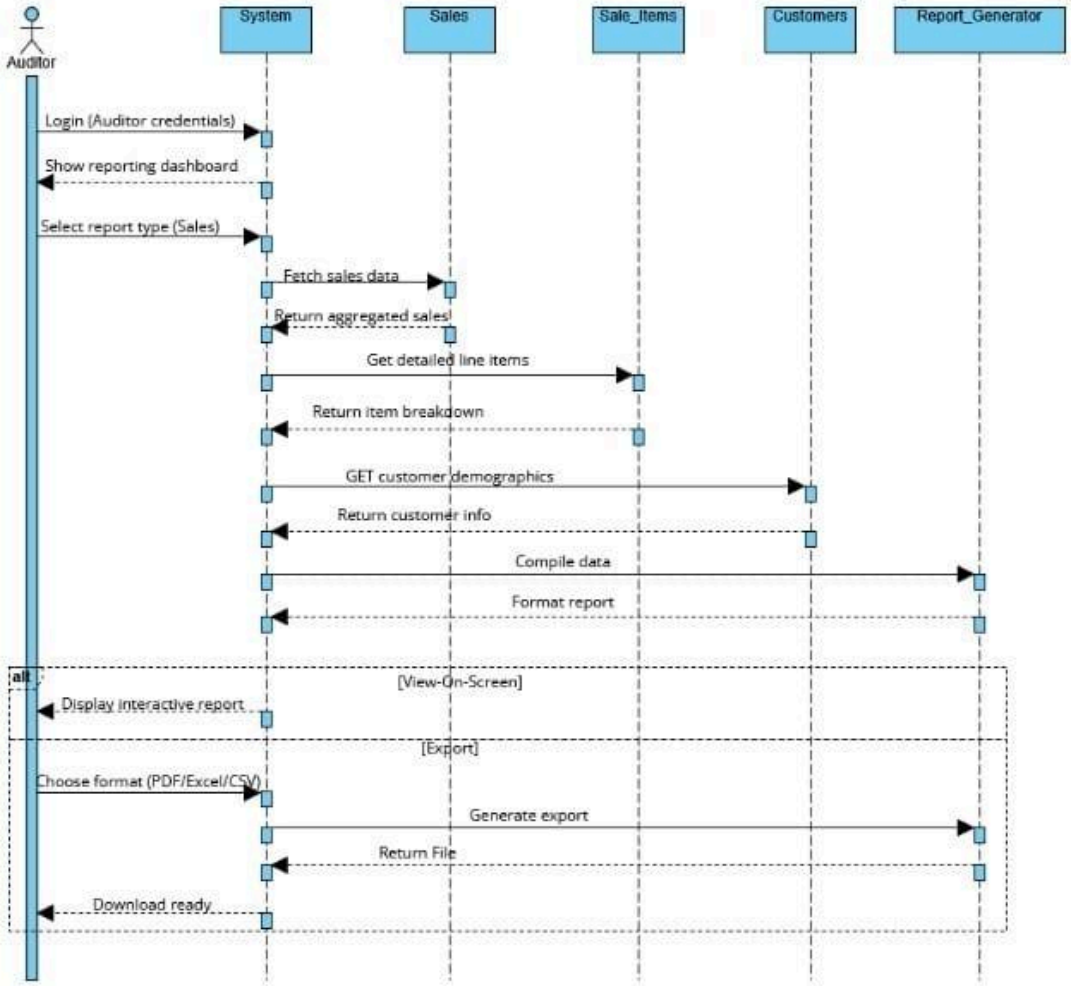
d) Stock Reorder/Expiry Management



e) User and Role Management



f) Generate Reports



5. Interfaces

- A detailed table of all 40 interfaces:

5.1 Authentication Interfaces (4)

5.2 Cashier Interfaces (8)

5.3 Inventory Management Interfaces (10)

5.4 Admin Interfaces (8)

5.5 Auditor Interfaces (6)

5.6 Shared/Utility Interfaces (4)

No.	Interface Name	Description	Role Access
1	Login Page	Login for all user roles	All
2	Logout Page	Logout functionality	All
3	Admin Dashboard	Admin control center	Admin
4	Cashier Dashboard	Quick access for billing and returns	Cashier
5	Inventory Dashboard	Stock overview	Inventory Manager
6	Auditor Dashboard	Reports and analytics view	Auditor
7	Billing Interface	Barcode scanning and cart updates	Cashier
8	Add to Cart	Add item to current transaction	Cashier
9	Edit Cart Item	Change quantity or remove item	Cashier
10	Discount Application	Apply automatic or manual discounts	Cashier
11	Payment Selection	Choose cash or card	Cashier
12	Generate Receipt	Print invoice after transaction	Cashier
13	Return Item Form	Manage item returns	Cashier

14	Product List View	See all product records	Admin, Inventory Manager
15	Add New Product	Add product details to inventory	Admin, Inventory Manager
16	Edit Product	Update product info	Admin, Inventory Manager
17	Delete Product	Remove product from system	Admin, Inventory Manager
18	Low-Stock Alert	Notification for restocking	Inventory Manager
19	Expiry Alert	View soon-to-expire or expired products	Inventory Manager
20	Reorder Threshold Setup	Set minimum stock levels	Inventory Manager
21	Stock Adjustment	Manually update stock quantity	Inventory Manager
22	Product Category Manager	Manage categories	Admin, Inventory Manager
23	User List View	See registered users	Admin
24	Add User	Register new staff member	Admin
25	Edit User Info	Update staff info	Admin
26	Assign Roles	Role-based permission control	Admin
27	Delete User	Remove user account	Admin
28	Staff Activity Log	View user activity logs	Admin, Auditor
29	Sales Report Viewer	View sales summary	Admin, Auditor
30	Inventory Report Viewer	Track stock movement	Admin, Auditor
31	Export Reports	Export in Excel or PDF	Admin, Auditor
32	Branch Comparison Tool	Analyze multiple branches (if enabled)	Auditor
33	Daily Sales Summary	Track daily revenue	Admin, Auditor
34	Transaction Log	See past transaction details	Admin

35	Backup System Data	Take system backup	Admin
36	Restore System Data	Restore previous backups	Admin
37	System Settings	Configure business rules	Admin
38	Product Search Interface	Search bar for quick product find	All
39	Help & FAQ Page	Assistance and common queries	All
40	Customer Feedback Entry	Record customer complaints/suggestions (optional)	Cashier

6. System Modules

6.1 Authentication & User Roles

Functions Performed

- Provides secure login access to all user roles (Admin, Cashier, Inventory Manager, Auditor)
- Implements role-based access control to restrict features based on user role
- Manages sessions and prevents unauthorized access

Inputs

- Username
- Password
- (Role is determined after successful authentication)

Outputs

- Access granted or denied message
- Redirects user to the appropriate role-specific dashboard

Connected Interfaces

- Login Interface
- Dashboard Interface

Role Access

- All roles: Admin, Cashier, Inventory Manager, Auditor

6.2 Billing & POS

Functions Performed

- Scans items using barcode input
- Manages cart operations (add/remove items, update quantities)
- Applies discounts either manually or automatically
- Accepts payment via cash or card
- Generates and prints receipts
- Handles returns and product exchanges

Inputs

- Product barcode
- Quantity
- Discount (if applicable)
- Selected payment method

Outputs

- Printed or digital receipt
- Updated inventory levels

- New transaction added to the log

Connected Interfaces

- Billing Interface (POS)
- Inventory Module (for stock adjustment)
- Reports Module (for sales tracking)

Role Access

- Cashier
- Admin (view/override access)

6.3 Inventory Management

Functions Performed

- Allows adding, editing and deleting product details
- Tracks inventory quantity, category, expiry dates and supplier information
- Monitors low-stock and expiry alerts
- Enables setting of minimum reorder levels

Inputs

- Product information (name, category, supplier, price, quantity, expiry)
- Minimum stock level

Outputs

- Updated inventory database

- Low-stock and expiry alerts

Connected Interfaces

- Add/Edit Product Interfaces
- Product List View
- Notification Panel

Role Access

- Inventory Manager
- Admin (full access)

6.4 Reporting & Auditing

Functions Performed

- Generates reports for daily sales, stock status, expired items and user activity
- Allows exporting reports in PDF, Excel, or CSV formats
- Supports report filtering by date range, branch, or user role

Inputs

- Date range
- Report type
- Optional filters (branch, user, etc.)

Outputs

- On-screen reports with exportable options

Connected Interfaces

- Reports Viewing Interface
- Print/Export Interface

Role Access

- Auditor
- Admin
- Inventory Manager (limited access to inventory reports)

6.5 Admin Controls

Functions Performed

- Manages user accounts (add, edit, delete)
- Assign roles and permissions
- Handles password resets

Inputs

- New user information
- Assigned role
- Existing user details (for updates or deletion)

Outputs

- Updated user list
- Confirmation messages

Connected Interfaces

- User Management Interface

- Authentication Module

Role Access

- Admin only

6.6 Backup & Restore

Functions Performed

- Performs manual backups of the entire system database
- Allows restoring the system to a previously saved backup state
- Prevents permanent data loss or corruption

Inputs

- Backup initiation command
- Selected backup file (for restore)

Outputs

- Confirmation of successful or failed operation
- Restored system state (data rollback)

Connected Interfaces

- Backup & Restore Panel
- Direct database connection

Role Access

- Admin only

6.7 Dashboard

Functions Performed

- Displays role-specific Key Performance Indicators (KPIs)
    - Cashier: Daily sales summary, active cart overview
    - Inventory Manager: Low-stock alerts, expiry notifications
    - Admin: System activity and user overview
    - Auditor: Sales trends and branch comparisons
- Provides quick access to relevant modules based on user role

Inputs

- User role (determined at login)
- Real-time backend data (inventory, sales, etc.)

Outputs

- Dynamic dashboard with widgets, graphs and shortcuts

Connected Interfaces

- Authentication Module
- Linked modules (Billing, Reports, Inventory)

Role Access

- All roles (dashboard content varies per role)

7. Implementation Details

- Technology stack

Layer	Technology Used
Frontend	Java Swing (for desktop GUI)
Backend	Java (JDBC)
Database	MySQL
Reporting	JasperReports
Build Tool	Maven or Gradle
IDE	IntelliJ IDEA
Version Control	Git (with GitHub or GitLab repository)

- Backend & Frontend Communication
- API or DB queries (if applicable)
- Code structure and directory layout (optional for academic projects)

8. Testing and Validation

- Unit Testing
- Integration Testing
- System Testing
- User Acceptance Testing (UAT)
- Sample test cases and screenshots of results

9. Results and Discussion

- Screenshots of major screens (login, billing, inventory)
- Performance results
- User feedback

10. Conclusion

This project successfully achieved its core objectives designing and developing a digital supermarket management system that streamlines operations, supports multiple user roles and ensures secure, efficient and real-time processing of daily supermarket activities.

SupraMart provides role-specific functionalities for cashiers, inventory managers, administrators and auditors. It offers barcode-based billing, real-time inventory tracking, user management, reporting and backup features, all within a user-friendly desktop environment. The system addresses the key problems faced by traditional supermarket setups and brings improved accuracy, transparency and control to business operations.

The development journey of SupraMart also enhanced our problem-solving skills as we encountered and addressed various technical and coordination challenges. From aligning user expectations with system design to debugging functionality during testing, each phase pushed us to think critically and adapt quickly.

From a learning perspective, this project gave us a deep understanding of software design, requirement analysis, user interface planning and teamwork.

11. References/Bibliography

Bibliography

We would like to specify the names of some websites which have helped us to complete this project.

- Oracle.(2024).*Java Swing Documentation*.Retrieved from https://docs.oracle.com/javase/8/docs/api/javax/swing/package-summary.html
- GeeksforGeeks.(2023).*Java JDBC Programs - Basic to Advanced*.https://www.geeksforgeeks.org/java/java-jdbc-programs/
- MySQL Documentation.(2023).*MySQL 8.0 Reference Manual*.https://dev.mysql.com/doc/
- GitHub.(2024).*Java POS System Projects*.Retrieved from https://github.com/
- perforce.(2025).*How to Write a Software Requirements Specification (SRS) Document*.https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document
- JasperReports Community.(2024).*JasperReports Documentation*.https://community.jaspersoft.com/

References

- Acharya,K. (2021) *SUPERMARKET MANAGEMENT SYSTEM PROJECT REPORT*.Available at:https://www.researchgate.net/publication/381469165_SUPERMARKET_MANAGEMENT_SYSTEM_PROJECT_REPORT (Accessed: 01 August 2025).
- Bitrus,E. (2014) *Design and Implementation of Computerized Supermarket Management System*.Available at:https://www.academia.edu/33171217/Design_and_Implementation_of_Computerized_Supermarket_Management_System (Accessed: 01 August 2025)

Project Timeline