Software Requirement Specification

Retail Wizard

Team Number: 22

Members:

202351173, Dev Saraswat

202351002, Aditya Raj

202351130, Shivam Kumar

202352320, Lavudya Siddhartha

202352311, Dream Vijay Sahare

Outline

List of Acronyms/Abbreviations	
SRS	Software Requirement Specification
DFD	Data Flow Diagram
SIG	System Integrity Goals
UI	User Interface
UPI	United Payments Interface

Bibliography

IEEE 830-1998 Software Requirements Specification Standard.

IEEE SA - IEEE 830-1998

Software Requirements Specification (SRS)

1. **Introduction**

1.1 Purpose

Retail Wizard is a comprehensive retail management system designed to assist shopkeepers in efficiently managing inventory, accounts, payments, and invoices while enhancing the overall customer experience. The system automates various manual tasks, improving accuracy and operational efficiency.

1.2 Scope

The system provides the following functionalities:

- **User Authentication:** Secure login for shopkeepers and customers.
- **Inventory Management:** Stock tracking, categorization, and reporting.
- **Accounts Management:** Recording transactions and generating financial reports.
- **Payment Processing:** Handling cash and UPI transactions, applying discounts, and calculating taxes.
- **Invoice Management:** Generating and securely storing invoices.

1.3 Overview

This document outlines the system's functional and non-functional requirements, design constraints, data flow models, and security goals.

2. Overall Description

2.1 Product Perspective

Retail Wizard is a standalone application integrated with a database to manage inventory, accounts, and financial transactions. It supports both online and offline operations.

2.2 Product Functions

- **Login System:** Secure authentication with role-based access control.

- **Inventory Management:** Adding, updating, deleting products, and tracking stock levels.
- **Accounts Management:** Generating sales, purchase, and stock reports.
- **Payment Processing:** Supporting cash and UPI payments, applying discounts, and calculating taxes.
- **Invoice Management:** Creating invoices for customers and suppliers.

2.3 User Characteristics

- **Shopkeepers:** Primary users responsible for inventory and financial management.
- **Customers:** End-users who purchase products and complete transactions.

2.4 Constraints

- Compliance with financial regulations and data security standards.
- Internet dependency for UPI transactions and cloud storage.

2.5 Assumptions and Dependencies

- Users have basic knowledge of digital transactions.
- Reliable internet connectivity is required for UPI payments.

3. Specific Requirements

3.1 Functional Requirements

3.1.1 User Authentication

- Secure login and registration for users.
- Role-based access control for shopkeepers and customers.

3.1.2 Inventory Management

- Adding, updating, and deleting products.
- Categorizing stock and setting restock alerts.
- Generating stock performance reports.

3.1.3 Accounts Management

- Recording all sales and purchase transactions.
- Generating financial reports:

- **Sales Reports:** Summarizing revenue from sales.
- **Purchase Reports:** Tracking procurement spending.
- **Stock Reports:** Analyzing inventory turnover.

3.1.4 Payment Processing

- Supports cash and UPI payments.
- Automatically calculates applicable taxes.
- Generates digital and printed receipts.

3.1.5 Invoice Management

- Generates and stores invoices for transactions.
- Allows invoice printing and digital access.

3.2 Non-Functional Requirements

- **Performance:** Fast processing and optimized database queries.
- **Usability:** Intuitive UI with clear error handling.
- **Security:** Data encryption, backup, and role-based authentication.
- **Automation:** Auto-generated reports and alerts.

4. Data Flow Diagrams (DFD)

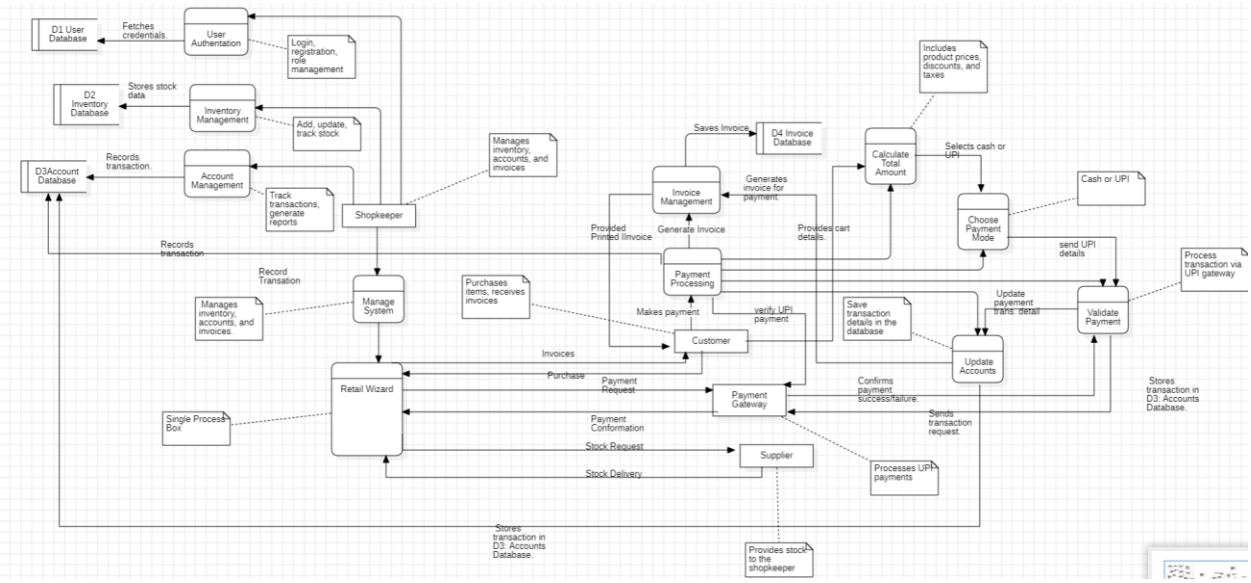

Fig. 1: DFD Diagram of our project

4.1 DFD Level 0 (Context Diagram)

External Entities:

- **Shopkeeper:** Manages inventory and financial operations.
- **Customer:** Purchases products and processes payments.
- **Payment Gateway:** Handles UPI transactions.
- **Supplier:** Provides inventory to the shopkeeper.

Data Flow:

1. Shopkeeper → Retail Wizard: Manages inventory and accounts.
2. Customer → Retail Wizard: Selects and purchases products.
3. Retail Wizard → Customer: Generates invoices.
4. Retail Wizard → Payment Gateway: Processes UPI transactions.
5. Retail Wizard → Supplier: Sends stock requests.
6. Supplier → Retail Wizard: Delivers products and invoices.

4.2 DFD Level 1 (Major Modules Breakdown)

Processes in DFD Level 1

1. **User Authentication:** Login, role management
2. **Inventory Management:** Stock tracking and categorization
3. **Accounts Management:** Transaction recording and reporting
4. **Payment Processing:** Handling cash, UPI, and tax calculations
5. **Invoice Management:** Generating and storing invoices

Data Stores in DFD Level 1

- **D1:** User Database (Stores login credentials)
- **D2:** Inventory Database (Stores stock details)
- **D3:** Accounts Database (Stores transactions and reports)
- **D4:** Invoice Database (Stores invoices)

4.3 DFD Level 2 (Detailed Process Breakdown)

Example: Payment Processing (4.0)

Sub-Processes:

4.3.1 Calculate Total Amount: Includes price, tax, and discounts.

4.3.2 Choose Payment Mode: Cash or UPI.

4.3.3 Validate Payment: Verify transaction via UPI gateway.

4.3.4 Update Accounts: Record payment details.

Data Flow:

1. Customer → (4.1) Calculate Total Amount: Provides cart details.
2. (4.1) Calculate Total Amount → (4.2) Choose Payment Mode: Selects cash or UPI.
3. (4.2) Choose Payment Mode → (4.3) Validate Payment: Sends UPI details.
4. (4.3) Validate Payment → Payment Gateway: Processes transaction.
5. Payment Gateway → (4.3) Validate Payment: Confirms payment.
6. (4.3) Validate Payment → (4.4) Update Accounts: Records transaction.
7. (4.4) Update Accounts → (5.0) Invoice Management: Generates invoice.

5. System Integrity Goals (SIG)

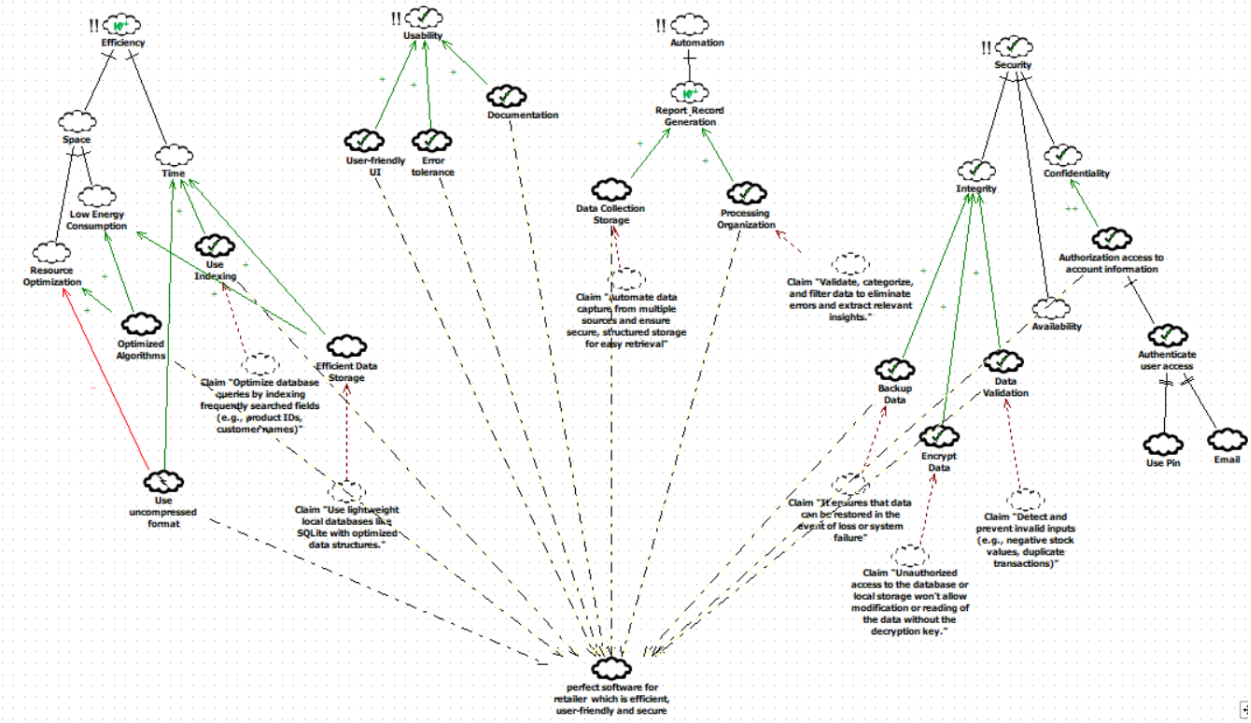

Fig 2: SIG Diagram

Core Soft Goals:

1. **Efficiency:** Optimizing system performance and memory consumption.
2. **Usability:** Creating an intuitive UI and ensuring error tolerance.
3. **Automation:** Auto-generated reports and alerts for better efficiency.
4. **Security:** Implementing encryption, authentication, and data validation.

6. Entity Relationship Diagram (ER Diagram)

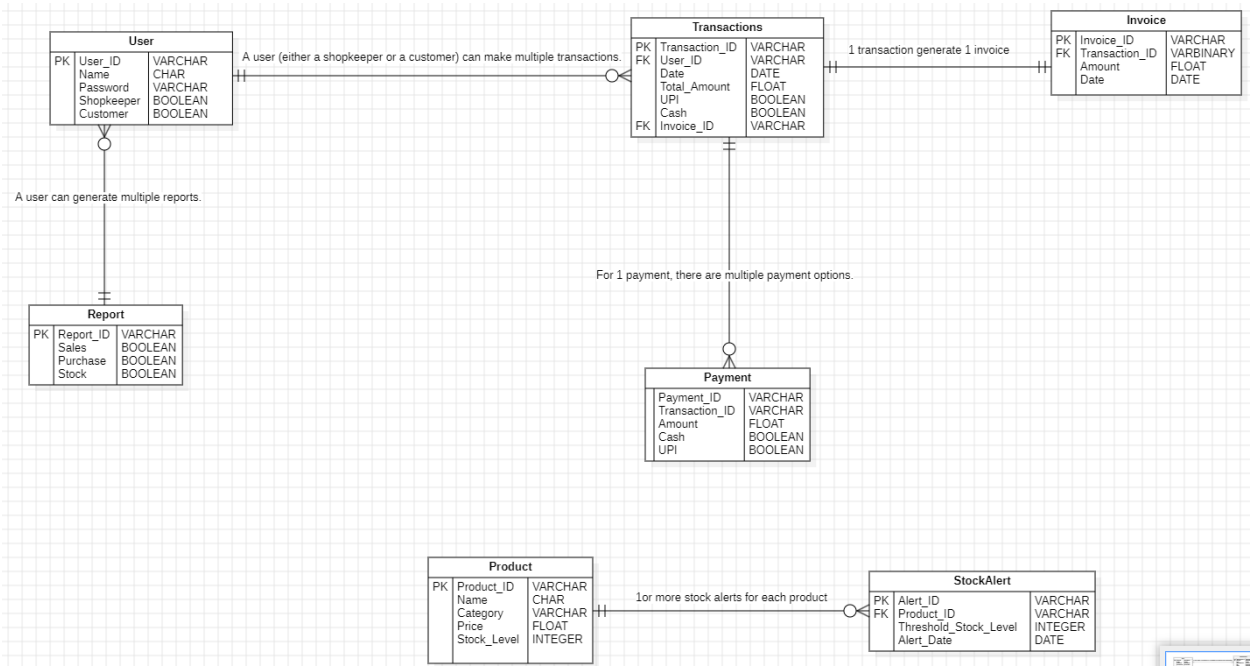

Fig 3: ER Diagram

This ER diagram represents a **Retail Transaction Management System**, involving users (shopkeepers/customers), transactions, invoices, payments, reports, products, and stock alerts.

- **User**: Represents shopkeepers and customers who can make multiple transactions and generate reports.
- **Transactions**: Stores transaction details like total amount, payment method, and invoice ID.
- **Invoice**: Each transaction generates a single invoice.
- **Payment**: A transaction can have multiple payment methods (cash, UPI, etc.).
- **Report**: Users can generate different types of reports (sales, purchase, stock).
- **Product**: Stores product details, including stock levels.
- **StockAlert**: Notifies when the stock level falls below a threshold.

Primary Key (PK) and Foreign Key (FK) Selection Justification

- **User Table**
  - **PK: User_ID** → Unique identifier for each user.
- **Transactions Table**

    - **PK: Transaction_ID** → Unique for each transaction.
    - **FK: User_ID** → Each transaction is linked to a user.
- **Invoice Table**
    - **PK: Invoice_ID** → Uniquely identifies each invoice.
    - **FK: Transaction_ID** → Links an invoice to a transaction (one-to-one relationship).
- **Payment Table**
    - **PK: Payment_ID** → Unique for each payment instance.
    - **FK: Transaction_ID** → A payment is linked to a transaction (one-to-many relationship).
- **Report Table**
    - **PK: Report_ID** → Unique for each report.
- **Product Table**
    - **PK: Product_ID** → Uniquely identifies each product.
- **StockAlert Table**
    - **PK: Alert_ID** → Unique for each stock alert.
    - **FK: Product_ID** → Each stock alert is related to a product.

  By choosing unique attributes for **PKs**, we ensure data integrity and uniqueness. The **FKs** establish relationships between tables, enabling efficient data retrieval and enforcement of business rules.

7. Use Case Diagram

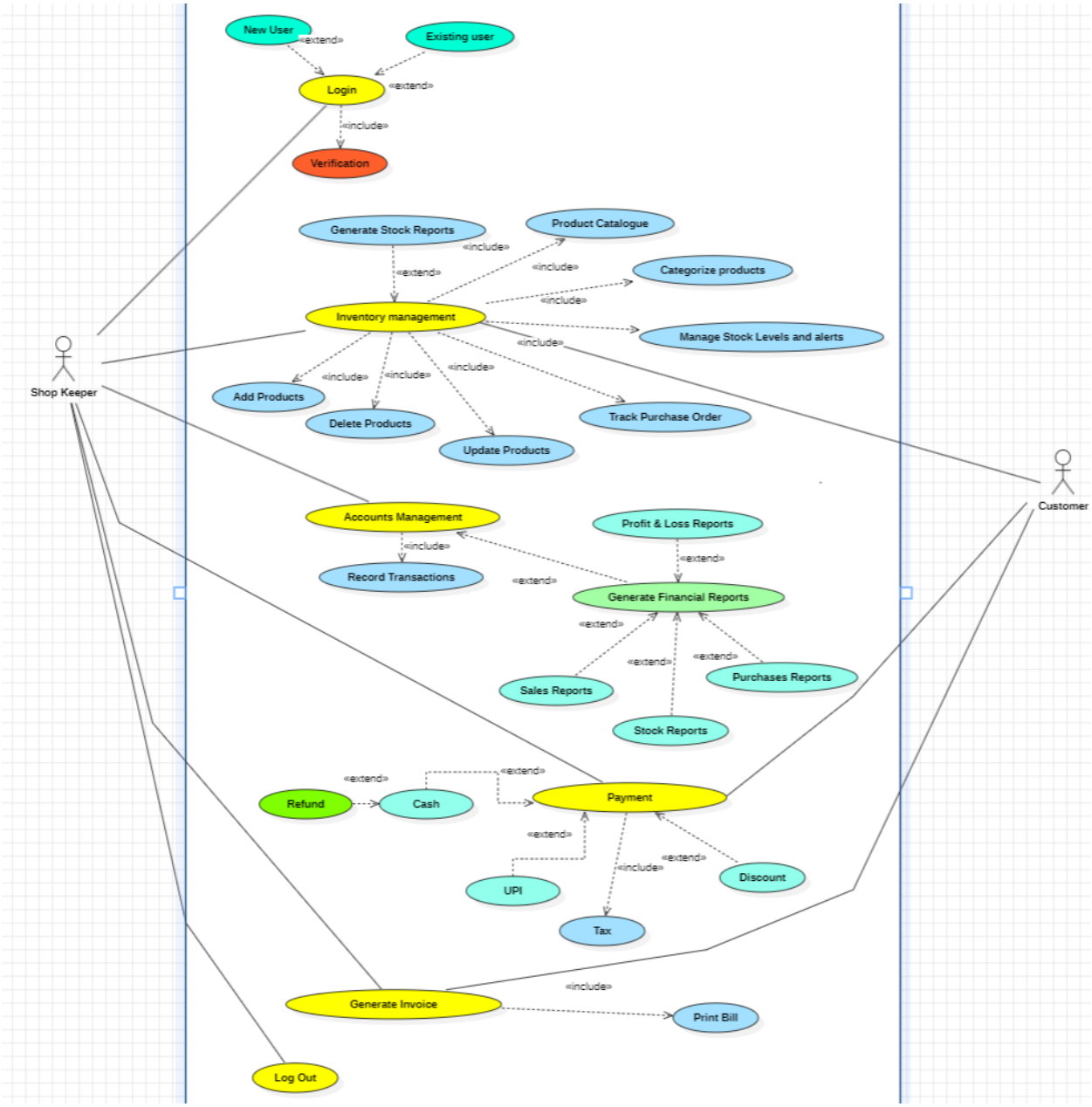

Fig 4: Use Case Diagram of our project

Following are the brief description of our use case in the diagram:

- Login – Allows new and existing users to access the system.

- Verification – Ensures the user's credentials are valid before granting access.
- Generate Stock Reports – Creates reports on stock availability and status.
- Inventory Management – Handles product-related operations such as adding, deleting, and updating products.
- Product Catalogue – Displays available products for reference.
- Categorize Products – Organizes products into different categories.
- Manage Stock Levels and Alerts – Tracks inventory levels and generates alerts for low stock.
- Track Purchase Order – Monitors and records incoming stock from suppliers.
- Add Products – Adds new items to the inventory.
- Delete Products – Removes unwanted or outdated items from the inventory.
- Update Products – Modifies product details such as price and description.
- Accounts Management – Manages financial transactions within the system.
- Record Transactions – Logs financial transactions for bookkeeping.
- Profit & Loss Reports – Analyzes financial performance by tracking income and expenses.
- Generate Financial Reports – Produces summaries of financial data including sales and purchases.
- Sales Reports – Details revenue generated from product sales.
- Purchases Reports – Tracks expenses incurred from purchasing inventory.
- Stock Reports – Summarizes stock usage and availability.
- Payment – Facilitates customer payments for purchases.
- Refund – Processes refunds for returned items.
- Cash – Handles cash-based transactions.
- UPI – Supports digital payments through UPI.
- Tax – Computes and applies tax to transactions.
- Discount – Applies discounts on purchases when applicable.

- Generate Invoice – Creates an invoice for completed transactions.
- Print Bill – Prints a physical copy of the transaction invoice.
- Log Out – Allows the user to exit the system securely.

8. Conclusion

The Retail Wizard SRS document outlines the system's functionalities, design constraints, and models essential for efficient inventory, accounts, and payment management. This document ensures compliance with IEEE standards and best practices in software engineering.