Software Requirements Specification (SRS) for Inventory Management System

1. Introduction

1.1 Purpose

The purpose of this document is to define the software requirements for the Inventory Management System. It specifies the system's intended features, performance needs, and design constraints. This document serves as a guideline for developers, testers, and stakeholders to ensure proper implementation and validation.

1.2 Scope

The Inventory Management System is a web-based application that allows organizations to track, manage, and control stock levels across warehouses and retail outlets. The system enables users to add, update, and remove items, monitor stock levels in real time, generate purchase and sales orders, and provide reporting for decision-making.

Core functions include stock management, supplier and customer management, purchase and sales tracking, and reporting. It is intended for use by storekeepers, sales staff, and managers to streamline operations and reduce errors.

1.3 Definitions and Acronyms

- **System**: The Inventory Management System.
- **User**: A storekeeper, sales staff, or administrator using the system.
- **Item/Product**: A good or material being managed in the inventory.
- **Stock**: The quantity of an item available in the system.
- **Supplier**: A vendor providing products to the business.
- **Customer**: A buyer purchasing items from the business.
- **UI**: User Interface.
- **API**: Application Programming Interface.

---

2. Overall Description

2.1 Product Perspective

The system is a centralized web-based solution with support for multi-user access. It will use a relational database (e.g., MySQL, PostgreSQL) for data storage and retrieval. It can integrate with barcode scanners, POS systems, and third-party ERP solutions through APIs.

2.2 Product Functions

The system provides the following major functionalities:

- **Stock Management**: Add, update, delete, and search items; set minimum stock levels; generate alerts.

- **Supplier Management**: Manage supplier information and purchase records.

- **Customer Management**: Manage customer data and sales history.

- **Purchase & Sales Management**: Record purchases from suppliers and sales to customers.

- **Reporting**: Generate stock reports, sales summaries, and purchase histories.

- **Security & Authentication**: Provide login, access control, and activity logging.

2.3 User Characteristics

- **Storekeeper**: Basic computer literacy; uses the system to update stock and manage suppliers.

- **Sales Staff**: Moderate technical knowledge; uses the system for creating sales orders and invoices.

- **Administrator/Manager**: Advanced user; manages system access, reviews reports, and oversees operations.

2.4 Design and Implementation Constraints

- **Technology**: Must be implemented as a web application with responsive UI.

- **Database**: Must ensure data consistency and support real-time updates.

- **Hardware Integration**: Should support barcode scanners for stock entry and checkout.

- **Performance**: The system should handle at least 1000 concurrent users with minimal latency.

- **Security**: Data must be encrypted and access must be role-based.

---

3. Specific Requirements

3.1 Stock Management Module

- **3.1.1 Add Item**: The system shall allow the user to add new items with details such as name, SKU, category, supplier, and quantity.

- **3.1.2 Update Item**: The system shall allow modification of existing item details.

- **3.1.3 Delete Item**: The user shall be able to delete items from the inventory.

- **3.1.4 Search Items**: The system shall provide search functionality by SKU, name, or category.

- **3.1.5 Stock Alerts**: The system shall generate low-stock alerts when quantities fall below the threshold.

3.2 Supplier Management Module

- **3.2.1 Add Supplier**: The system shall allow the user to add supplier details.

- **3.2.2 Update Supplier**: The system shall allow modification of supplier details.

- **3.2.3 Supplier History**: The system shall maintain a record of transactions with each supplier.

3.3 Customer Management Module

- **3.3.1 Add Customer**: The system shall allow the user to add customer details.

- **3.3.2 Update Customer**: The system shall allow modification of customer details.

- **3.3.3 Customer History**: The system shall maintain a sales history for each customer.

3.4 Purchase & Sales Management Module

- **3.4.1 Record Purchase**: The system shall allow the user to create purchase orders and update stock.

- **3.4.2 Record Sale**: The system shall allow the user to create sales invoices and reduce stock.

- **3.4.3 Return Handling**: The system shall support purchase and sales returns.
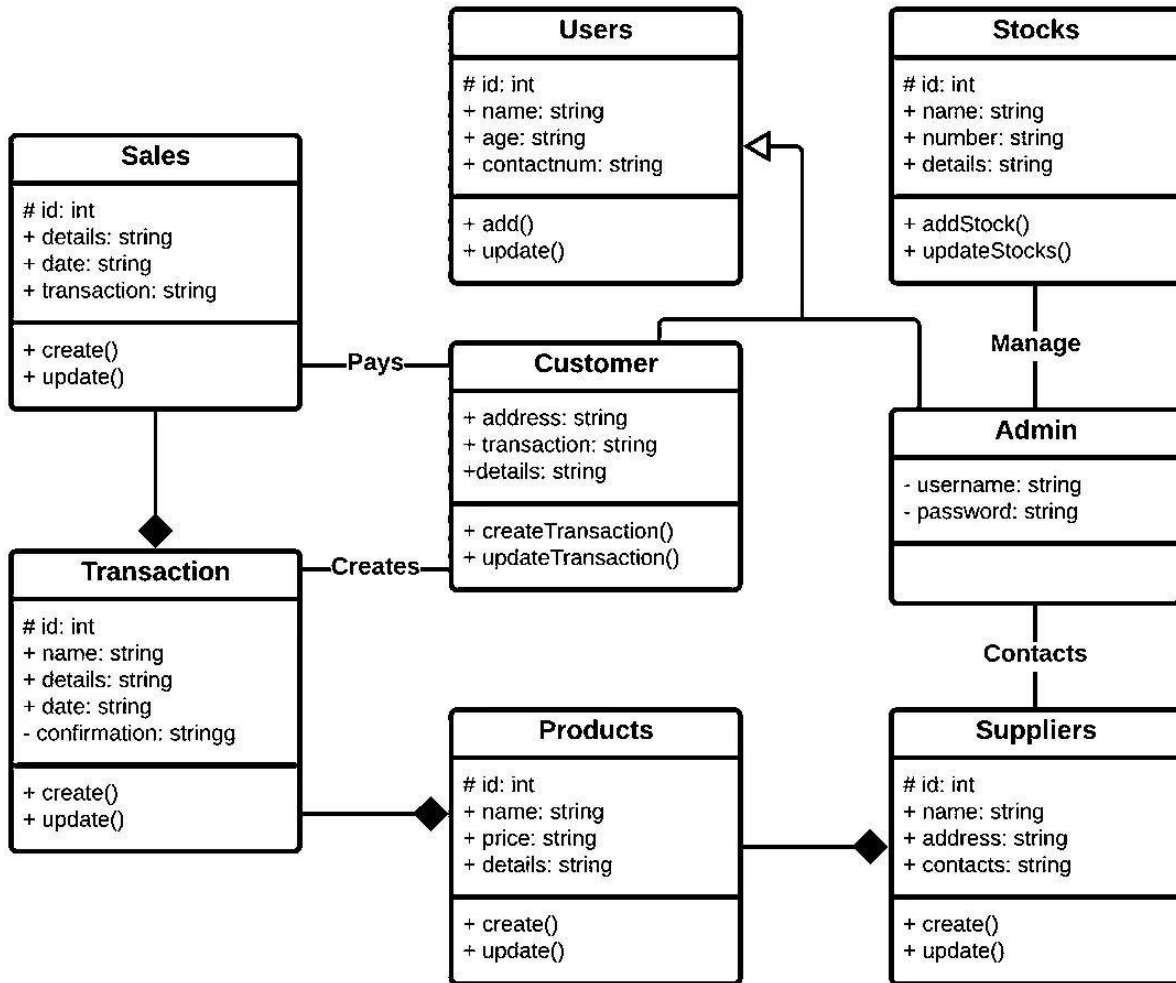
3.5 Reporting Module

- **3.5.1 Stock Reports**: The system shall generate reports of current stock levels.

- **3.5.2 Sales Reports**: The system shall provide daily, weekly, and monthly sales reports.

- **3.5.3 Purchase Reports**: The system shall generate reports of supplier purchases.

- **3.5.4 Profitability Reports**: The system shall calculate profit margins based on purchase and sales data.
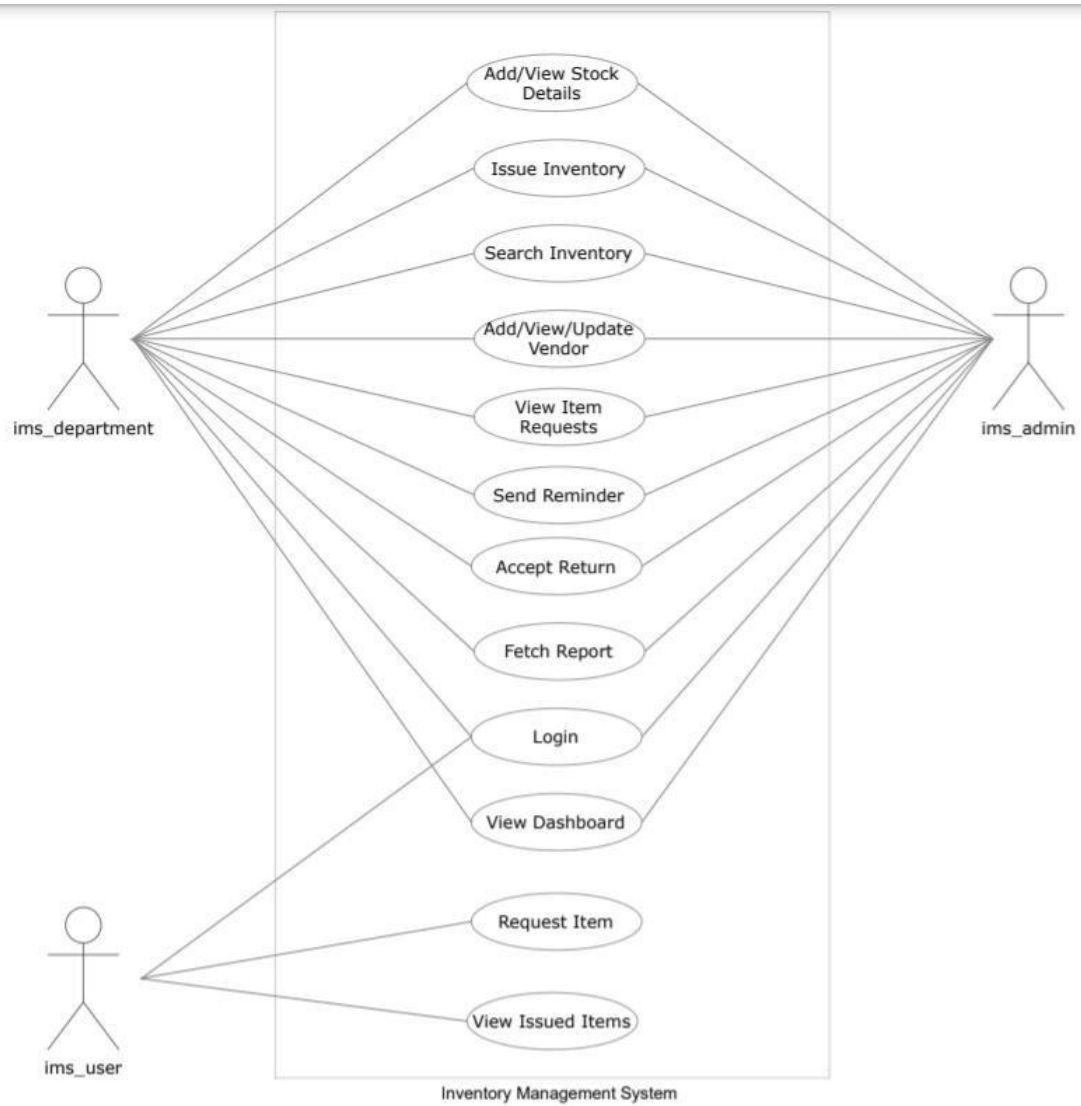
3.6 Security Module

- **3.6.1 Authentication**: The system shall require login credentials for access.

- **3.6.2 Authorization**: Access shall be role-based (storekeeper, sales staff, admin).

- **3.6.3 Audit Logs**: All activities shall be logged for accountability.
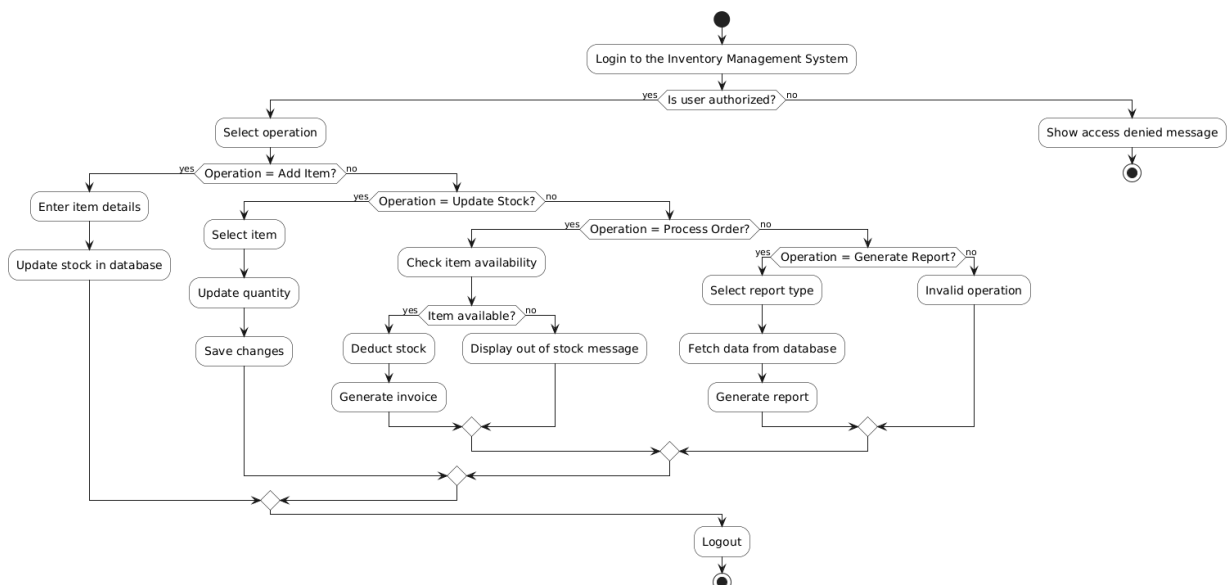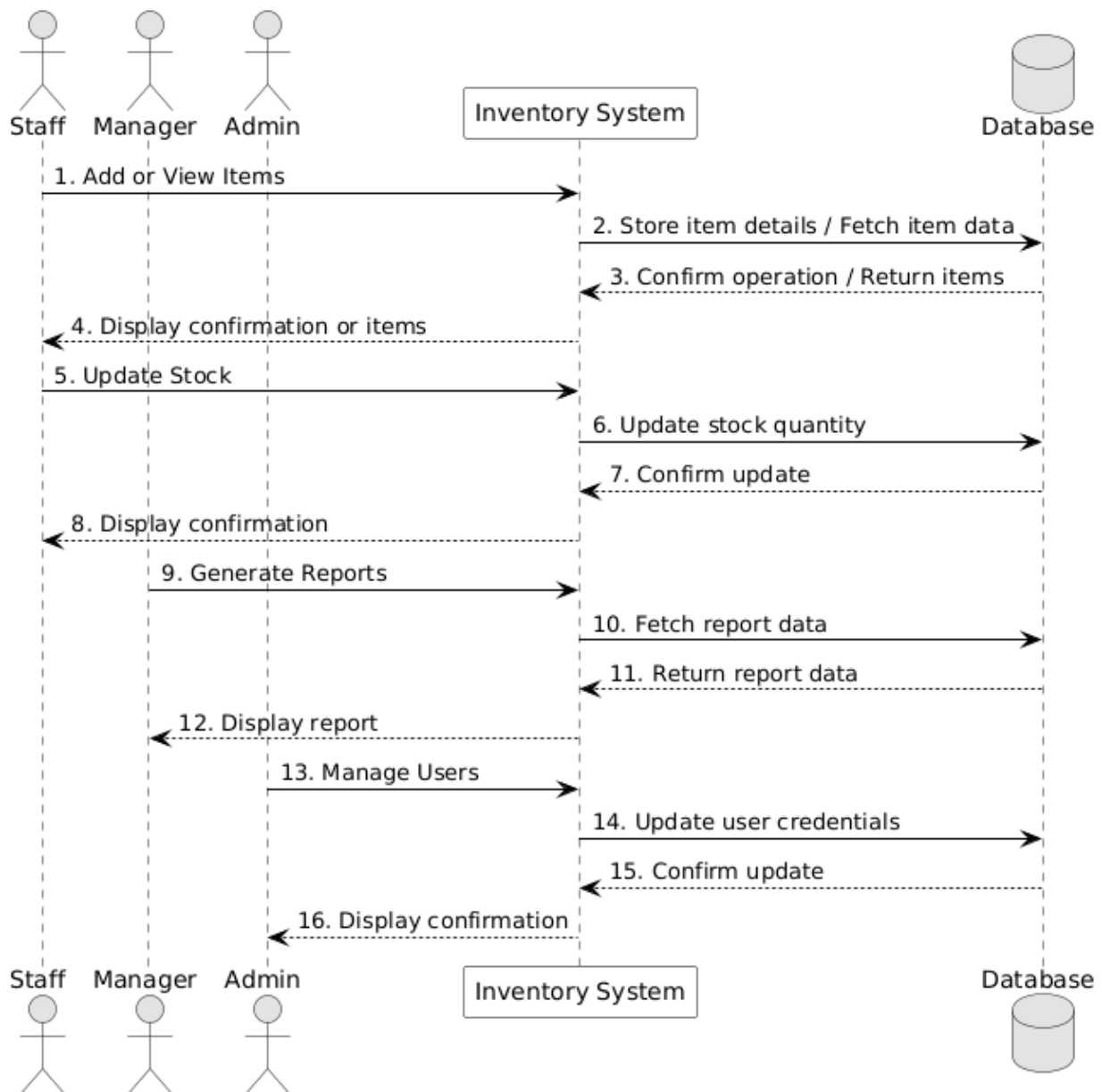
---

4. System Diagrams

4.1 Class Diagram

Users

id: int
+ name: string
+ age: string
+ contactnum: string

+ add()
+ update()

Stocks

id: int
+ name: string
+ number: string
+ details: string

+ addStock()
+ updateStocks()

Sales

id: int
+ details: string
+ date: string
+ transaction: string

+ create()
+ update()

—Pays—

Customer

+ address: string
+ transaction: string
+details: string

+ createTransaction()
+ updateTransaction()

Manage

Admin

- username: string
- password: string

Transaction

id: int
+ name: string
+ details: string
+ date: string
- confirmation: stringg

+ create()
+ update()

—Creates—

Contacts

Products

id: int
+ name: string
+ price: string
+ details: string

+ create()
+ update()

Suppliers

id: int
+ name: string
+ address: string
+ contacts: string

+ create()
+ update()

4.2 Usecase Diagram
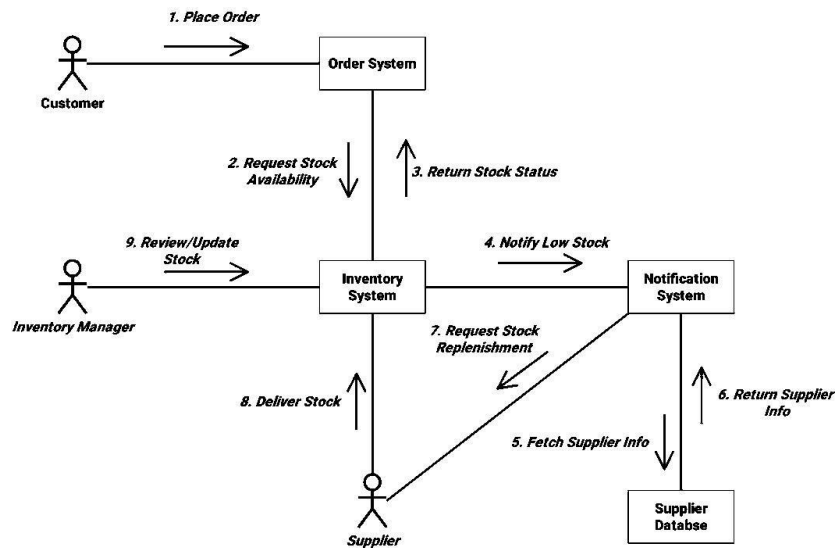

Inventory Management System
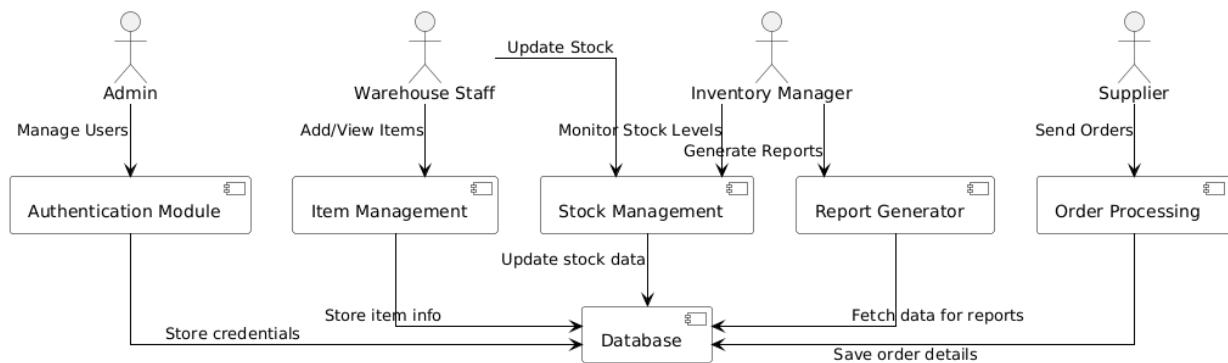
4.3 Activity Diagram

4.4.1 Sequence Diagram

4.4.2 Collaboration Diagram

4.5 Component/Deployment Diagram

Conclusion

This SRS outlines the requirements for the Inventory Management System, providing a clear understanding of its functionality, performance, and constraints. The system is designed to streamline inventory tracking, reduce errors, and improve overall efficiency. By following this specification, the development team can ensure the system meets user needs and supports effective inventory management.