

SOFTWARE REQUIREMENT SPECIFICATION

Name:	Helan Stefani B
Roll Number:	7376222IT153
Seat Number:	101
Project Id:	21
Project Title:	Inventory System

1. TOOLS AND TECHNOLOGIES:

MERN STACK

COMPONENTS	TECH STACK
Frontend	React
Backend	Node.js with Express.js
Database	MongoDB
API	OpenAI

2. PROBLEM STATEMENT:

Many businesses struggle with manual or outdated inventory management systems. This leads to several problems:

- **Inaccurate Stock Levels:** Manual tracking is prone to errors, resulting in inaccurate stock information. This can lead to stockouts, lost sales, and overstocking.
- **Inefficient Inventory Control:** Time is wasted on manual processes like stock counting and data entry. This reduces productivity and limits the ability to focus on other areas.
- **Poor Visibility and Reporting:** Lack of centralized data make it difficult to track trends, identify low stock items, and make informed decisions about purchasing.
- **Missed Sales Opportunities:** Stockouts can lead to lost sales and customer dissatisfaction.

The inventory system is a web application designed to manage purchases, sales, stock availability, and generate reports. It will cater to users with appropriate access levels.

3.PURPOSE:

The purpose of this project is to develop a comprehensive web application for managing inventory effectively and efficiently. The inventory system will address the challenges faced by businesses due to manual or outdated inventory management processes. By providing a centralized platform with intuitive functionalities, the system aims to streamline purchase and sales processes, enhance stock visibility, facilitate informed decision-making through detailed reporting, and ensure timely alerts and reminders for critical events.

KEY OBJECTIVES:

- 1. Streamlined Purchase and Sales Management:** The system will offer dedicated portals for recording purchase and sales transactions. Users can easily input relevant details such as item information, quantity, price, supplier/customer details, and transaction dates.
- 2. Real-time Stock Availability:** Users will have access to live stock data, allowing them to monitor current stock levels and receive alerts for low stock items. This feature ensures that businesses can maintain optimal inventory levels and avoid stockouts.
- 3. Comprehensive Reporting:** The system will generate comprehensive reports on inventory data, including consolidated, yearly, monthly, weekly, and daily reports. These reports will provide insights into sales performance, stock levels, and trends, enabling users to make data-driven decisions.
- 4. Reminders and Alerts:** Users will receive timely reminders and alerts for critical events such as low stock levels or pending purchase orders. This feature helps in proactive inventory management and prevents disruptions in operations.
- 5. Integration with Messaging Services:** The system will integrate with messaging services such as WhatsApp to facilitate seamless communication and collaboration. Users can share generated reports directly with relevant stakeholders for better transparency and decision-making.

4. SCOPE:

The inventory management system will encompass functionalities for:

- Recording purchase transactions including item details, supplier information, and date of purchase.
- Logging sales transactions with item details, customer information, and date of sale.
- Real-time monitoring of stock availability and low stock alerts.
- Generating various reports on sales, purchases, and stock levels.

- Sending reminders/alerts for critical events such as low stock levels.
- Integrating with messaging services for automated report sharing.

5. FUNCTIONAL REQUIREMENTS:

USER MANAGEMENT:

- **Secure Login:** Users can authenticate themselves using a username/email and password combination.
- **Logout:** Users can easily terminate their session and exit the application.

INVENTORY MANAGEMENT:

Purchase Entry Portal

1. **Create Purchase Orders:** Users can initiate new purchases by adding item details like name, description, quantity, price, and supplier information.
2. **Verify Purchase Information:** The system automatically validates entered purchase details to ensure accuracy and completeness, preventing errors.
3. **Automatic Stock Updates:** Upon successful purchase confirmation, the system automatically updates the inventory stock level for the purchased items.

Sales Entry Portal

1. **Process Sales Transactions:** Users can record new sales by entering item details, quantity, price, and customer information.
2. **Real-time Stock Validation:** The system verifies entered sales information, ensuring accuracy and sufficient stock availability to prevent overselling.
3. **Inventory Level Adjustment:** After a successful sale, the system automatically reduces the inventory stock level for the sold items.

Stock Management

1. **Live Stock Visibility:** Users can access real-time data showcasing the current stock availability for each item in the inventory.

Reporting System

1. **Generate Comprehensive Reports:** Users can initiate the creation of reports on inventory data, offering insights at different timeframes (consolidated, yearly, monthly, weekly, daily).
2. **Flexible Data Filtering:** The system allows users to filter report data based on the chosen report type for a more focused analysis.
3. **Downloadable Reports:** Users have the option to download generated reports in user-friendly formats like PDF or CSV for further analysis or sharing.

Reminders and Alerts

1. **Low Stock Notifications:** The system proactively alerts users when stock levels for specific items reach the pre-defined reorder level, preventing stockouts.
2. **Data Sharing via Messaging Services:** Users can share generated reports directly through integrated messaging services like WhatsApp, facilitating communication and collaboration.

SUPPLIER MANAGEMENT:

- Implementation of a supplier management module where users can add, edit, and deactivate supplier information.
- Enabling users to track supplier performance metrics such as delivery time, quality of goods, and communication responsiveness.

6. NON-FUNCTIONAL REQUIREMENTS:

Security:

1. Secure user authentication and authorization.

Performance:

1. The system should be responsive and handle user requests efficiently.
2. Reports should be generated within a reasonable timeframe.

Usability:


1. User-friendly interface with clear navigation and intuitive functionalities.
2. Consistent design patterns throughout the application.


Maintainability:


1. The code should be well-documented and maintainable for future modifications.

7. ADDITIONAL CONSIDERATIONS:

 Define user roles and access permissions for different functionalities e.g., admin, manager, salesperson.

 Specify data validation rules for all input fields.

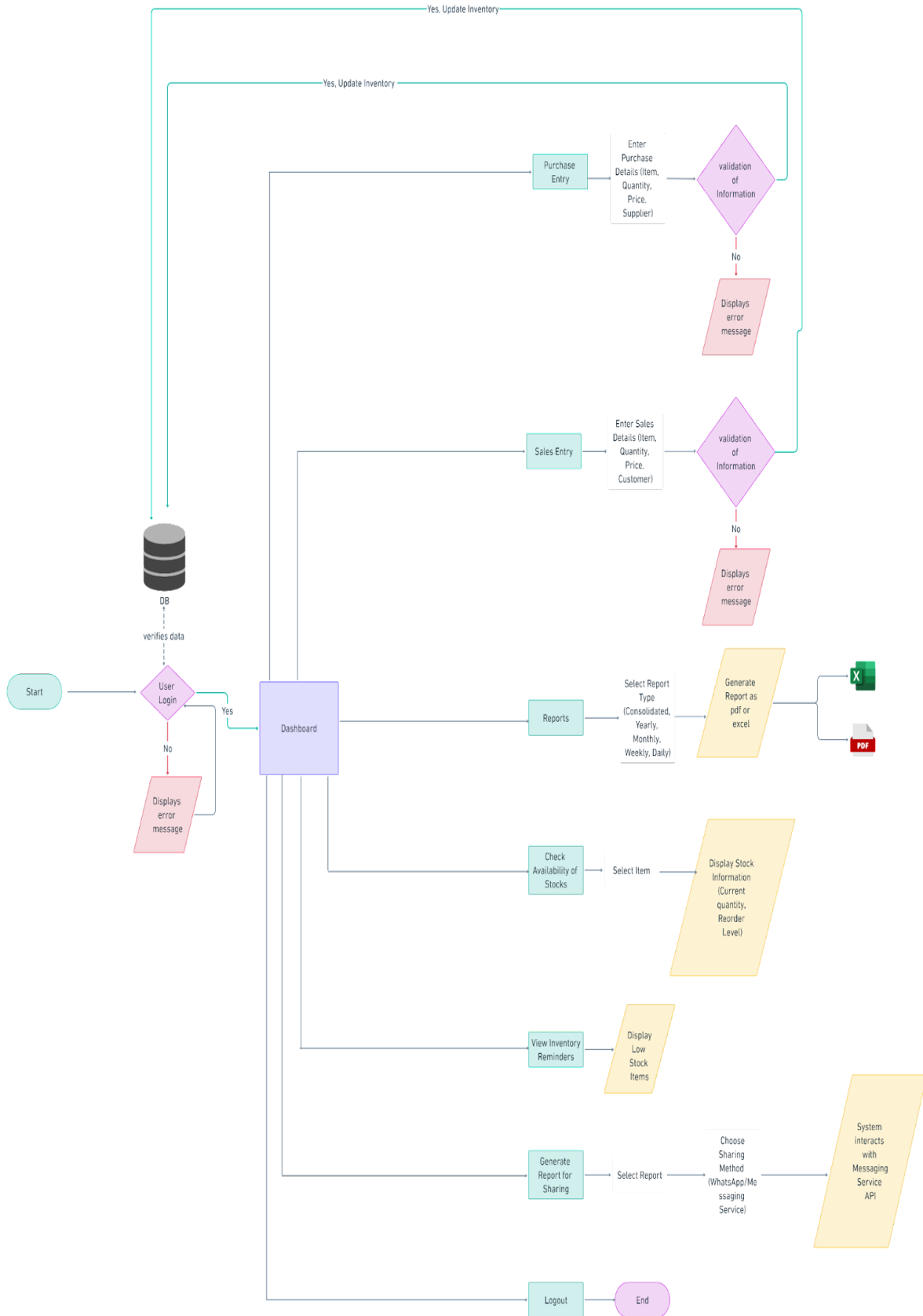
 Outline error handling procedures for invalid data or system errors.

 Consider backup and data recovery mechanisms.

Notification Settings:

- Allows users to customize notification preferences, such as receiving low stock alerts via email, SMS, or in-app notifications.
- Providing the ability to set thresholds for different types of notifications based on user preferences or roles.

FLOWCHART



PROTOTYPE OF THE PROJECT:



Welcome back!

Please enter your details

Email

Password

☐ Remember for 30 days

[Forgot password?](#)

Log In



Log In with Bitsathy Credentials

Don't have an account? [Register Here](#)

