**Frontend Development With React.js – Project Documentarion**

**Introduction**

**1.Project Title:**

* Store Manager – Keep Track of Inventory System

**2.Team Members:**

1.PAVITHRA M

2.POOVESHWARI G

3.RADHIKA S

4.NITHYA SRI B

* Team members Roles:
  + 1. PAVITHRA M: Coding
    2. POOVESHWARI G: Coding Development
    3. RADHIKA S: Document
    4. NITHYA SRI B: Demo Video

**Project Overview**

**Purpose:**  
This project helps store managers efficiently track inventory, manage stock levels, monitor product inflows and outflows, and generates reports.

**Goals:**

* Reduce stockouts and overstocking.
* Provide real-time updates on inventory.
* Enhance decision-making through data visualization.

**Features:**

* Add, update, and delete products.
* View current stock levels.
* Automatic alerts for low stock.
* Sales and purchase tracking.
* Search and filter products.
* Generate inventory reports.

**Architecture**

**Component Structure:**

* **Dashboard** – Overview of inventory status.
* **Product Management** – CRUD operations on products.
* **Stock Alerts** – Notifications for low stock.
* **Reports** – Sales and inventory reports.

**State Management:**

* Redux (or Context API) for global inventory data.

**Routing:**

* React Router for navigation between Dashboard, Products, Reports, and Settings.

**Setup Instructions**

**Prerequisites:**

* Node.js (v16+)
* npm or yarn
* Git

**Installation:**

1. Clone repo: git clone <repo-url>
2. Navigate: cd store-manager
3. Install deps: npm install
4. Run app: npm start

**Folder Structure**

* **/components** – Reusable UI components.
* **/pages** – Dashboard, Products, Reports, Login.
* **/assets** – Images, icons, styles.
* **/redux** – Store, reducers, actions.

**Running the Application**

npm start – Runs the app on localhost:3000.

**Component Documentation**

* **ProductCard** – Displays product details.
* **InventoryTable** – Shows stock levels.
* **ReportChart** – Visualizes sales data.
* **AlertBadge** – Low stock notifications.

**State Management**

* **Global State:** Inventory data, sales records.
* **Local State:** UI interactions like modal open/close, search filters.

**User Interface**

* Dashboard with key stats.
* Product management forms.
* Charts for sales/inventory trends.

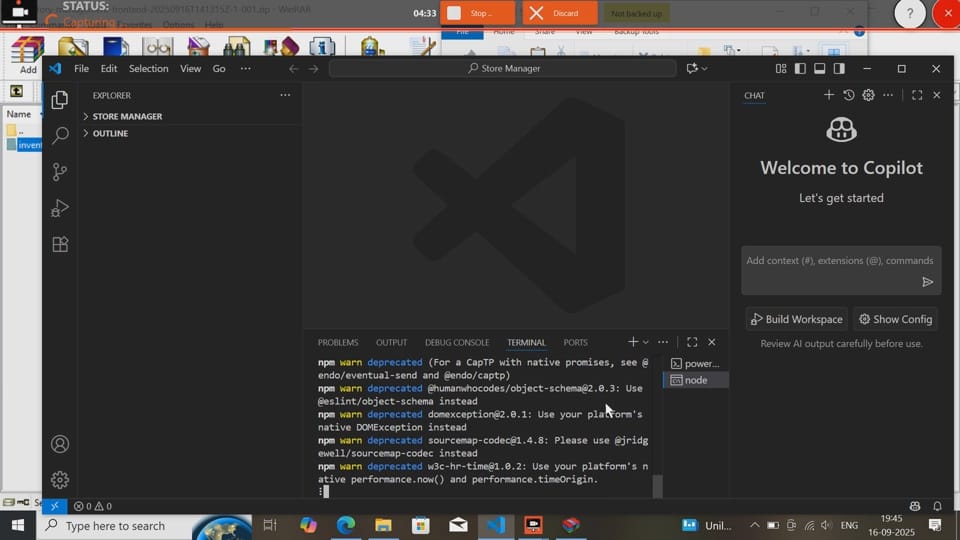
**Styling**

* Tailwind CSS for design.
* Custom theme for branding.

**Testing**

* Jest + React Testing Library for unit tests.
* Cypress for end-to-end tests.

**Screenshot or demo:**



**Known Issues**

* Limited offline functionality.
* Report export only in CSV.

**Future Enhancements**

* Barcode scanning integration.
* Mobile app version.
* AI-based demand forecasting.