Project Documentation

**1. Introduction**

• Project Title: Store Manager

• Team ID: Not Applicable

• Team Leader: Sham (mgc7sham24@gmail.com)

• Team Members:

– Samraj (mgc7samraj24@gmail.com)

– Rajiv Adhitya (mgc7adhitya24@gmail.com)

– Rajesh (mgc7rxjesh24@gmail.com)

**2. Project Overview**

• Purpose: The Store Manager project is designed to simplify store operations. It enables users to purchase items, manage inventory, view sales records, and add new products efficiently.

• Features:

– Home Page

– Product Catalog

– Cart Management

– Inventory Management

– Sales Record

– Add New Product

**3. Architecture**

• Frontend: React.js with Tailwind CSS for UI design and styling.

• Backend: Not implemented (frontend-only project).

• Database: Not implemented (data stored temporarily in state/context).

**4. Setup Instructions**

• Prerequisites:

– Node.js

– npm (Node Package Manager)

– Visual Studio Code

• Installation Steps:

1. Clone the repository.

2. Navigate to the project folder.

3. Run `npm install` to install dependencies.

4. Run `npm start` to launch the application.

**5. Folder Structure**

Store-Manager/  
 ├── node\_modules/  
 ├── public/  
 ├── src/  
 │ ├── components/  
 │ │ ├── Cart/  
 │ │ │ ├── Cart.jsx  
 │ │ │ └── CartItem.jsx  
 │ │ ├── Inventory/  
 │ │ │ └── Inventory.jsx  
 │ │ ├── Product/  
 │ │ │ ├── Product.jsx  
 │ │ │ ├── ProductCatalog.jsx  
 │ │ │ └── ProductList.jsx  
 │ │ ├── Sales/  
 │ │ │ ├── Sales.jsx  
 │ │ │ └── SaleRecord.jsx  
 │ │ ├── AddProduct.jsx  
 │ │ └── NavBar.jsx  
 │ │  
 │ ├── context/  
 │ ├── hooks/  
 │ ├── pages/  
 │ ├── App.css  
 │ ├── App.js  
 │ ├── App.test.js  
 │ ├── index.css  
 │ ├── index.js  
 │ ├── Layout.jsx  
 │ ├── logo.svg  
 │ ├── reportWebVitals.js  
 │ └── setupTests.js  
 │  
 ├── .gitignore  
 ├── package-lock.json  
 ├── package.json  
 ├── README.md  
 └── tailwind.config.js

**6. Running the Application**

• Frontend: Run `npm start` inside the project folder.

• Access: Visit [http://localhost:3000](http://localhost:3000/)

**7. API Documentation**

Not Implemented. (This is a frontend-only project.)

**8. Authentication**

Not Implemented.

**9. User Interface**

The following screenshots show the key UI features of the application:













**10. Testing**

• Testing Strategy: Manual testing of features such as product addition, cart management, inventory updates, and sales record verification.

• Tools: Browser DevTools (Chrome), Visual Studio Code.

**11. Screenshots or Demo**

Screenshots have been included in Section 9 (User Interface).

**12. Known Issues**

• No backend integration, so data is not persistent.

• Limited error handling and validation.

**13. Future Enhancements**

• Add backend support with a database (e.g., MongoDB, MySQL).

• Implement authentication and user roles.

• Improve error handling and form validations.

• Add reporting and analytics for sales.

• Enhance UI with animations and advanced styling.