/

**Project Documentation**

**1.Introduction:**

**Project Title:** Store Manager – Keep Track of Inventory

**Team ID:** NM2025TMID42634

**Team Leader:**

B. Surthilaya **– surthilaya256@gmail.com**

**Team Members:**

V. Sathya **– sathya792006@gmail.com**

S. Atchaya Sri **– atchayasri281@gmail.com**

F. Seema **– seemafairy52@gmail.com**

**2. Project Overview:**

**Purpose:**

Store Manager is designed to help businesses efficiently manage inventory by tracking stock, sales, and updates in real-time.

**Features:**

Add, update, and delete products

Track stock levels and sales

Generate reports for inventory status

User authentication for secure access

Admin control panel

**3. Architecture:**

**Frontend:** React.js with Bootstrap and Material UI

**Backend:** Node.js and Express.js managing server logic and APIs

**Database:** MongoDB stores product details, transactions, and user data

**4. Setup Instructions:**

**Prerequisites:**

Node.js

MongoDB

Git

React.js

Express.js

Mongoose

Visual Studio Code

**Installation Steps:**

# Clone the repository

git clone <repository-url>

# Install client dependencies

cd client

npm install

# Install server dependencies

cd ../server

npm install

**5. Folder Structure:**

Store-Manager/

│-- client/ # React frontend

│ ├── components/

│ └── pages/

│

│-- server/ # Node.js backend

│ ├── routes/

│ ├── models/

│ └── controllers/

**6. Running the Application:**

**Frontend:**

cd client

npm start

**Backend:**

cd server

npm start

Access: Visit http://localhost:3000

**7. API Documentation:**

**User:**

POST /api/user/register

POST /api/user/login

**Inventory:**

POST /api/products/add

PUT /api/products/update/:id

DELETE /api/products/delete/:id

GET /api/products/:id

**Reports:**

GET /api/reports/stock

GET /api/reports/sales

**8. Authentication:**

JWT-based authentication for secure login

Middleware for protecting private routes

**9. User Interface:**

Landing Page

Inventory Dashboard

Product Management Page

Admin Panel

Reports Page

**10. Testing:**

**Type:** Manual testing during milestones

**Tools:** Postman, Chrome DevTools

**11. Screenshots or Demo:**

[Attach screenshots or demo links here]

**12. Known Issues:**

[List any current issues/bugs here]

**13. Future Enhancements:**

Barcode scanner integration

Low-stock alerts via email/SMS

Cloud backup and synchronization

Mobile application support