**Project Documentation**

**Store Manager: Keep track of Inventory.**

**1. Introduction**

**Project Title:** Store Manager: Keep track of Inventory.

**Team ID:** [NM2025TMID42696]

**Team Leader**: [PRIYA.K.& [priyak2007pt@gmail.com](mailto:priyak2007pt@gmail.com)]

**Team Members:**

[RAMYA.G. & [ramyagopi9566@gmail.com](mailto:ramyagopi9566@gmail.com)]

[RAKSHITHA.K & [rakshitharakshitha6462@gmail.com](mailto:rakshitharakshitha6462@gmail.com)]

[PRIYADHARSHINI.V & [pd148920@gmail.com](mailto:pd148920@gmail.com)]

**2. Project Overview**

**Purpose:**

The Store Manager application helps businesses efficiently track inventory levels, manage stock, and monitor sales activities in real time.

**Features:**

 Add, edit, and delete products

 Real-time inventory tracking

 Low-stock alerts

 Product categorization

 Sales and revenue reports

 Role-based access (Admin/Staff)

**3. Architecture:**

 **Frontend:** React.js with Bootstrap / Material UI

 **Backend:** Node.js and Express.js

 **Database:** MongoDB to store product, category, and sales 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

**7. API Documentation**

**User:**

 POST /api/user/register

 POST /api/user/login

**Products** 

POST /api/products/create

 GET /api/products/:id

 PUT /api/products/:id

 DELETE /api/products/:id

**Sales:**

 POST /api/sales/add

 GET /api/sales/report

**8. Authentication:**

 JWT-based authentication for secure login

 Middleware protects private routes

**9. User Interface:**

* Dashboard (overview of stock and sales)
* Products Page (CRUD operations)
* Sales Page
* Reports Page
* Admin Panel

**10. Testing:**

 Manual testing during milestones

 Tools: Postman, Chrome Dev Tools

**11. Screenshots or Demo link:**

https://drive.google.com/file/d/148ecA0N5ZpvfX0e3zZ4caOHPm9X4ce4f/view?usp=drive\_link

**12. Future Enhancements:**

 Barcode scanning support

 Export inventory data (CSV/PDF)

 Multi-store support

 Billing system integration