STORE MANAGER: KEEP TRACKING OF INVENTORY

Project Documentation

1.Introduction:

➤ **PROJECT TITLE: Store manager keep tracking of inventory**

➤**TEAM ID: NM2025TMID43252**

➤**TEAM SIZE: 3**

➤**TEAM LEADER: KAYALVIZHI P**

➤**ROLE: CODING AND DEVELOPMENT**

➤**TEAM MEMBER: VIMALA S**

➤**ROLE: CODING AND DEVELOPMENT**

➤**TEAM MEMBER: SWARNA D**

➤**ROLE: CODING AND DEVELOPMENT**

1. Project Overview

The Store Manager app is a web-based system that allows store managers to track, update, and manage inventory in real-time.

2. Purpose and Features

Its main purpose is to streamline inventory management with features like adding, editing, and deleting products, viewing stock levels, and tracking restock dates.

3. Architecture

The application follows a modular architecture using React for the frontend, Redux for state management, and optionally Node.js and MongoDB for backend and database support.

4. Component Structure

Components are divided into reusable UI elements (like ProductCard), container components (like InventoryList), and page components (like DashboardPage).

5. State Management and Routing

State is managed globally using Redux Toolkit, while routing is handled by React Router for smooth navigation between pages like Dashboard and Inventory.

6. Setup Instructions

To set up the project, clone the repository, run npm install to install dependencies, and start the development server with npm start.

7. Folder Structure

The project is organized into folders such as/client for frontend code, /redux for state logic, /components for UI, and /pages for views.

8. Client

The client folder contains all the React application files including components, pages, styles, Redux store, and routing configurations.

9. Utilities

Utility functions, such as API request handlers or date formatters, are stored in a utilities or utils folder to keep the code clean and reusable.

10. Code

The codebase is written in JavaScript (or TypeScript), follows best practices with modular design, and is structured for scalability and maintainability.