

## Store Manager – Frontend Documentation

### Introduction

Project Title: Store Manager

Team Members: P.Priyadharshini, U.S Sadhana, R.A Yuvashree, A.Samrin Fathima

### Project Overview

Purpose: The Store Manager project is designed to provide an efficient interface for managing products, inventory, and sales.

The goal is to deliver a user-friendly and responsive frontend that enables store admins to easily handle day-to-day operations.

### Features:

1

Dashboard for quick insights into sales and inventory

2

Product management (add, update, delete, search products)

3

Inventory tracking with stock alerts

4

User authentication (login/logout)

5

Responsive design for desktop and mobile

### Architecture

Component Structure: App, Navbar, Dashboard, ProductList, ProductForm, Inventory, Auth

### Components

State Management: Context API (or Redux) for global state, React hooks for local state

Routing: React Router with paths for Dashboard, Products, Inventory, and Authentication

### Setup Instructions

Prerequisites: Node.js, npm/yarn

### Installation:

Clone repository → Navigate to directory → Install dependencies → Configure .env file

## Folder Structure

src/ contains assets, components, pages, context, hooks, utils, App.jsx, index.js

## Running the Application

Use `npm start` to run the frontend locally at <http://localhost:3000>

## Component Documentation

Key Components: Dashboard, ProductList, ProductForm, Inventory, Auth Components

Reusable Components: Button, Modal, FormInput

## State Management

Global state with Context API or Redux. Local state with React hooks.

## Styling

Tailwind CSS (or chosen CSS framework). Custom CSS modules for specific components.

## Testing

Unit testing with Jest/RTL. Integration testing for component interactions. E2E testing with Cypress (optional).

## Known Issues

1

UI may need optimization for mobile

2

Product search filter requires refinement

3

API error handling not fully implemented

## Future Enhancements

1

Add role-based access control

2

Implement analytics with charts

3

Improve UI with animations

Export product/inventory data as Excel/PDF