Project Documentation

Store Manager: Keep Track of Inventory

# Introduction

## Project Title: Store Manager: Keep Track of Inventory

* + **Team ID :** NM2025TMID37772
  + **Team Leader:** CHARUSMIRTHA M (charuharish1010@gmail.com)

## Team Members:

* + - DEEPIKA K G (g.deepika1911@gmail.com)
    - NAVYA V (vnnavya2007@gmail.com)
    - MANJARI P (Manjariponnurangam@gmail.com)

# Project Overview

* + **Purpose:** The Inventory Management System helps businesses manage stock efficiently by providing real-time product tracking, sales monitoring, and inventory updates.

## Features:

* + - Product management (add, update, delete items)
    - Inventory tracking with stock availability
    - Sales and billing management
    - Shopping cart functionality
    - Alert system for low-stock items
    - Search and filter options for products

# Architecture

* + **Frontend:** React.js with Bootstrap and Material UI
  + **Backend:** Node.js with Express.js
  + **Database:** MongoDB stores user data, project information, applications, and chat messages
  + **Other Tools:** Git, Visual Studio Code

# Setup Instructions

## Prerequisites:

* + - Node.js
    - MongoDB
    - Git
    - React.js
    - Express.js

## Installation Steps:

# Clone the repository git clone

# Install client dependencies cd client npm install

# Install server dependencies cd

../server npm install

# Folder Structure

Inventory-Management/

|-- client/ # React frontend

| |-- components/

| |-- pages/

|-- server/ # Node.js backend

| |-- routes/

| |-- models/

| |-- controllers/

# Running the Application

## Frontend:

cd client npm start •**Backend**

cd server

npm start

* + **Access:** Visit <https://localhost:3000>

# API Documentation

* **Products:**
  + POST /api/products/create
  + GET /api/products/:id
  + PUT /api/products/update/:id
  + DELETE /api/products/delete/:id
* **Cart:**
  + POST /api/cart/add
  + GET /api/cart
* **Sales:**
  + POST /api/sales/create
  + GET /api/sales/history

# Authentication

* + JWT-based authentication for secure login
  + Middleware protects private routes

# User Interface

* **Home Page** – Navigation to Cart, Inventory, Sales
* **Inventory Page** – Product list with stock & price details
* **Cart Page** – Items added for checkout
* **Sales Page** – Sales reports and transaction details
* **Add Product Page** – Admin can add/manage products

# Testing

#  **Tools Used:** Postman, Chrome DevTools

#  Testing Approach: Manual testing for CRUD operations, API validations, and stock alerts

1. **Screenshots or Demo**

[**demo video (1) (1) (1).mp4**](file:///C:\Users\user\Downloads\demo%20video%20(1)%20(1)%20(1).mp4)

1. **Known Issues**
   * + Limited search filters
     + No multi user role support (only admin and user)
2. **Future Enhancements**

* Role-based access control (Admin, Staff, Customer)
* Integration with barcode scanner for faster inventory updates
* Advanced analytics and sales reports
* Cloud deployment