1. Introduction :
   * + Project Title: Store Manager
     + Team Id : NM2025TMID48665
     + Team Leader : Shruthi Priya.S & [shruthipriyas07@gmail.com](mailto:shruthipriyas07@gmail.com)
     + Team Members:
     + Shruthi Priya.S
     + Email id: [shruthipriyas07@gmail.com](mailto:shruthipriyas07@gmail.com)
     + Narmadha.V
     + Email id: narmadhavenkatesh406@gmail.com
     + Keerthana.S
     + Email id: [skrowdybaby2813@gmail.com](mailto:skrowdybaby2813@gmail.com)
     + Devadharshini.M
     + Email id: <2007deva06@gmail.com>

2.Project overview:

* Purpose: A Store manager purpose is to over see the daily operations of a retail store, ensuring it runs smoothly and efficiently.
* Features

Leader ship

Communication

Problem solving

Time management

Adaptability

Analytical skills

Customer focus

Inventory management

3.Architecture:

Frontend:

* Customer interaction

Sales flow management

Backend:

* Data bases
* Servers
* Stock management
* Finance

Data base:

* Mongo DB Stores user data
* Project information
* Application
* Chat message

Setup Instructions:

* Prerequisites:
* Node. Js
* MongoDB
* Git
* Resct. Js
* Express. Js

Installation steps:

#Clone the repository git clone

#Install client dependencies cd client npm install

#Install server dependencies cd../server npm install

5.Folder Structure:

* Since “store manager “ can be built with different technologies (node.Js,Django,java spring boot )the folder structure depends on the tech stack.

6.Running the Application:

* Frontend:
* Instruction that explain how to start the system.
* Steps to run both the backend (server) and frontend(UI).

7.API Documentation

* User :
* An API documentation explains how the backend endpoints for things like store manager, user, products, orders, etc., can be use.
* Project:
* POST / auth /login = Login user
* POST /products = Add new product
* GST /orders = Fetch all orders
* POST /chat /send = Send chat message
* Let me break them down clearly with definitions , roles ,and how they connect.
* Chats:
* /api/chat/send
* /api/chat/:userId

8.Authentication:

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

9.User Interface:

* Landing page
* Freelancer dashboard
* Admin panel
* Project details page

10.Testing:

* Manual Testing during milestones
* Tools: Postman ,chrome Dev Tools