PROJECT DOCUMENTATION

STORE MANAGER

- ## INTRODUCTION

Project title: Store Manager

Team ID: NM2025TMID39891

Team leader: Sandhya G & saizzsandy@gmail.com

Team members:

- SANDHYA G
- MONISHA V
- VINOTHINI V
- SHARMILA S

- ## PROJECT OVERVIEW

Purpose: *The purpose of this project is to effectively manage and oversee the daily operations of the store, ensuring smooth functioning, customer satisfaction, and achievement of sales and business goals. The store manager will be responsible for supervising staff, maintaining inventory, implementing company policies, and creating a positive shopping experience. This role ensures the store operates efficiently, profitably, and in alignment with organizational standards and objectives.*

Features:

- ✓ **Leadership & Team Management** – *Supervises staff, assigns duties, and motivates employees.*
- ✓ **Customer Service** – *Ensures customer satisfaction and handles complaints or queries.*
- ✓ **Sales & Profitability** – *Focuses on achieving sales targets and improving store performance.*
- ✓ **Inventory Control** – *Monitors stock levels, reduces wastage, and manages reordering.*
- ✓ **Staff Training & Development** – *Provides training to improve skills and efficiency.*
- ✓ **Operational Efficiency** – *Ensures smooth day-to-day operations of the store.*
- ✓ **Financial Management** – *Manages budgets, cash flow, and expense control.*
- ✓ **Compliance & Policies** – *Ensures the store follows company rules, safety, and legal regulations.*
- ✓ **Reporting & Analysis** – *Tracks sales data, prepares reports, and suggests improvements.*
- ✓ **Store Presentation** – *Maintains cleanliness, product display, and overall shopping environment.*

- ARCHITECTURE

1. Frontend (User Interface)

- ❖ **Technology Options**: *React.js, Angular, Vue.js (for web) / Flutter, React Native (for mobile)*

- **Features**:
  - *Dashboard for sales & performance overview*
  - *Product & inventory management screens*
  - *Employee & shift management interface*
  - *Customer billing & order management UI*
  - *Reports & analytics visualization*
  - ---
  - **2. Backend (Application Layer)**
- **Technology Options**: *Node.js (Express), Java (Spring Boot), Python (Django/Flask), .NET Core*
- **Responsibilities**:
  - *Handles business logic (sales, discounts, employee roles, etc.)*
  - *API development for frontend communication*
  - *Authentication & authorization (manager, staff, admin roles)*
  - *Order processing and transaction management*
  - *Notification system (low stock alerts, sales reports)*
  - ---
  - **3. Database (Data Layer)**
- **Options**:
  - **Relational**: *MySQL, PostgreSQL (for structured inventory, employees, sales records)*
  - **NoSQL**: *MongoDB (if handling unstructured data like customer reviews, logs)*
- **Data Stored**:
  - *Product catalog (ID, name, price, stock, supplier info)*
  - *Employee records (roles, shifts, performance)*

- *Customer data (billing, purchase history)*
- *Sales & financial transactions*
- *Reports & analytics logs*
- 📌 **In short**

- **Frontend** → *User interface (React/Angular/Flutter)*
- **Backend** → *Business logic + APIs (Node.js/Java/Python)*
- **Database** → *Store all records (MySQL/PostgreSQL/MongoDB)*

ARCHITECTURE NOTES

- ***Frontend****: User interface for managers and staff (React/Angular/HTML+CSS+JS).*
- ***Backend****: Handles business logic and API (Node.js, Django, Spring Boot, etc.).*
- ***Database****: Stores product, sales, and user data (MySQL, MongoDB, PostgreSQL).*
- ***API Layer****: RESTful APIs for communication between frontend & backend.*

DATABASE (important tables)

- ***Users****: user_id, username, password, role*
- ***Products****: product_id, name, category, price, stock*
- ***Sales****: sale_id, product_id, quantity, total_amount, date*
- ***Reports****: summary of sales, profit, inventory status*

ADAVANTAGES OF PROJECT

- *Reduces manual work.*
- *Increases accuracy in stock and billing.*
- *Easy sales tracking and reporting.*
- *Scalable for small to large stores.*

SECURITY NOTES

- *Use **Authentication & Authorization** for roles.*
- *Store passwords in **hashed format**.*
- *Prevent SQL Injection and XSS attacks.*

FUTURE ENHANCEMENTS (Good to Mention in Interviews)

- *Barcode/QR code scanning.*
- *Integration with online payment systems.*
- *Mobile app for store management.*
- *AI-based stock prediction.*

ARCHITECTURE NOTES

- ***Frontend**: User interface for managers and staff (React/Angular/HTML+CSS+JS).*
- ***Backend**: Handles business logic and API (Node.js, Django, Spring Boot, etc.).*

- ***Database**: Stores product, sales, and user data (MySQL, MongoDB, PostgreSQL).*
- ***API Layer**: RESTful APIs for communication between frontend & backend.*

The Store Manager Project improves efficiency by simplifying product, sales, and inventory management, making store operations faster, accurate, and more reliable.