

Project Proposal

On

Medicine Tracker

Guided By:-

Anuj Kumar

Created By:-

Aditya Chaudhary

AF04971828

ANP-D2404

WDR-1

Table Of Contents

1. Title of the Project
2. Introduction
3. Objective
4. Project Category
5. Analysis
 - Modules and Description
 - Database Design
 - ER Diagram
 - Data Flow Diagram
6. Complete Structure
 - Process Logical Diagram
7. Platform Used
 - Hardware Requirement
 - Software Requirement
8. Future Scope
9. Bibliography

Introduction

Medicine Tracker is a comprehensive digital health management system designed to simplify and streamline the way individuals handle their daily medications. In today's fast-paced lifestyle, people often struggle to remember dosage timings, keep track of remaining medicine stock, maintain prescription history, and manage multiple medicines prescribed for different conditions. This leads to missed doses, accidental overdosing, expired medicine consumption, and overall poor medication adherence. Medicine Tracker addresses all these challenges by providing a centralized, user-friendly platform where users can digitally organize and monitor every aspect of their medicine lifecycle.

The system allows users to store detailed medicine information, set up customizable dosage schedules, receive timely reminders, and track their adherence history. Each action is supported by secure authentication and a strict MySQL-based backend to ensure accuracy, reliability, and data integrity. Users can also upload and manage their prescriptions, maintain a complete log of when medicines were taken or skipped, and receive notifications for low stock and upcoming expiries. The application is particularly beneficial for individuals with chronic illnesses, elderly users who require assistance in managing their medicines, and families caring for multiple members.

Objectives

- Provide a dependable medicine management system for end users.
- Enable precise tracking of dosages and timing with configurable reminders.
- Maintain an auditable prescription and medication history.
- Track inventory levels and send low-stock notifications.
- Offer a clear, minimal UI so users of all ages can manage medicines easily.

Project Category

Software Application — Healthcare Management System
(Full-Stack Web Application)

Analysis

Modules and short descriptions:

- User Module — Registration, login, profile, role handling (patient, caregiver).
- Medicine Module — Add / edit / delete medicines (name, strength, form, manufacturer, instructions).
- Schedule Module — Create dosage schedules (time, frequency, days, start/end date).
- Reminder Engine — Trigger notifications at scheduled times and log taken / skipped doses.
- Stock Module — Track quantities, expiry dates, and generate low-stock alerts.
- Prescription History — Store uploaded prescriptions and link them to medicines and schedules.

DATABASE DESIGN

1. USERS:

Column	Data Type	Constraints
user_id	INT	PRIMARY KEY, AUTO_INCREMENT
name	VARCHAR(150)	NOT NULL
email	VARCHAR(255)	UNIQUE, NOT NULL
password_hash	VARCHAR(255)	NOT NULL
phone	VARCHAR(25)	NULL
role	VARCHAR(20)	DEFAULT 'user'
created_at	TIMESTAMP	DEFAULT CURRENT_TIMEST

2. MEDICINES

Column	Data Type	Constraints
medicine_id	INT	PRIMARY KEY, AUTO_INCREMENT
user_id	INT	FOREIGN KEY → users.user_id, NOT
name	VARCHAR(200)	NOT NULL
strength	VARCHAR(100)	NOT NULL
form	VARCHAR(50)	NOT NULL (e.g., tablet, syrup,
manufacturer	VARCHAR(150)	NULL
notes	TEXT	NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMEST

3. STOCK

Column	Data Type	Constraints
stock_id	INT	PRIMARY KEY, AUTO_INCREMENT
medicine_id	INT	FOREIGN KEY → medicines.medicin
quantity	INT	NOT NULL
unit	VARCHAR(30)	NOT NULL (e.g., tablets, ml)
expiry_date	DATE	NOT NULL
batch_no	VARCHAR(100)	NULL
last_updated	TIMESTAMP	DEFAULT CURRENT_TIMEST

4. DOSAGE_SCHEDULE

Column	Data Type	Constraints
schedule_id	INT	PRIMARY KEY, AUTO_INCREMENT
medicine_id	INT	FOREIGN KEY → medicines.medicin
dose_amount	DECIMAL(10,2)	NOT NULL
dose_unit	VARCHAR(50)	NOT NULL (e.g., mg, ml)
frequency_type	VARCHAR(50)	NOT NULL (e.g., daily, weekly,
times_json	JSON	NOT NULL (stores multiple timings
start_date	DATE	NOT NULL
end_date	DATE	NULL (if continuous
reminder_offset_m inutes	INT	DEFAULT 0

5. DOSE_LOG

Column	Data Type	Constraints
log_id	INT	PRIMARY KEY, AUTO_INCREMENT
schedule_id	INT	FOREIGN KEY → dosage_schedule.s
taken_at	TIMESTAMP	NULL (if missed/skipped)
status	ENUM	('taken', 'skipped', 'missed') NOT
note	TEXT	NULL

6. PRESCRIPTIONS

Column	Data Type	Constraints
prescription_id	INT	PRIMARY KEY, AUTO_INCREMENT
user_id	INT	FOREIGN KEY → users.user_id, NOT
file_path	VARCHAR(255)	NOT NULL
doctor_name	VARCHAR(150)	NULL
issued_date	DATE	NULL
notes	TEXT	NULL

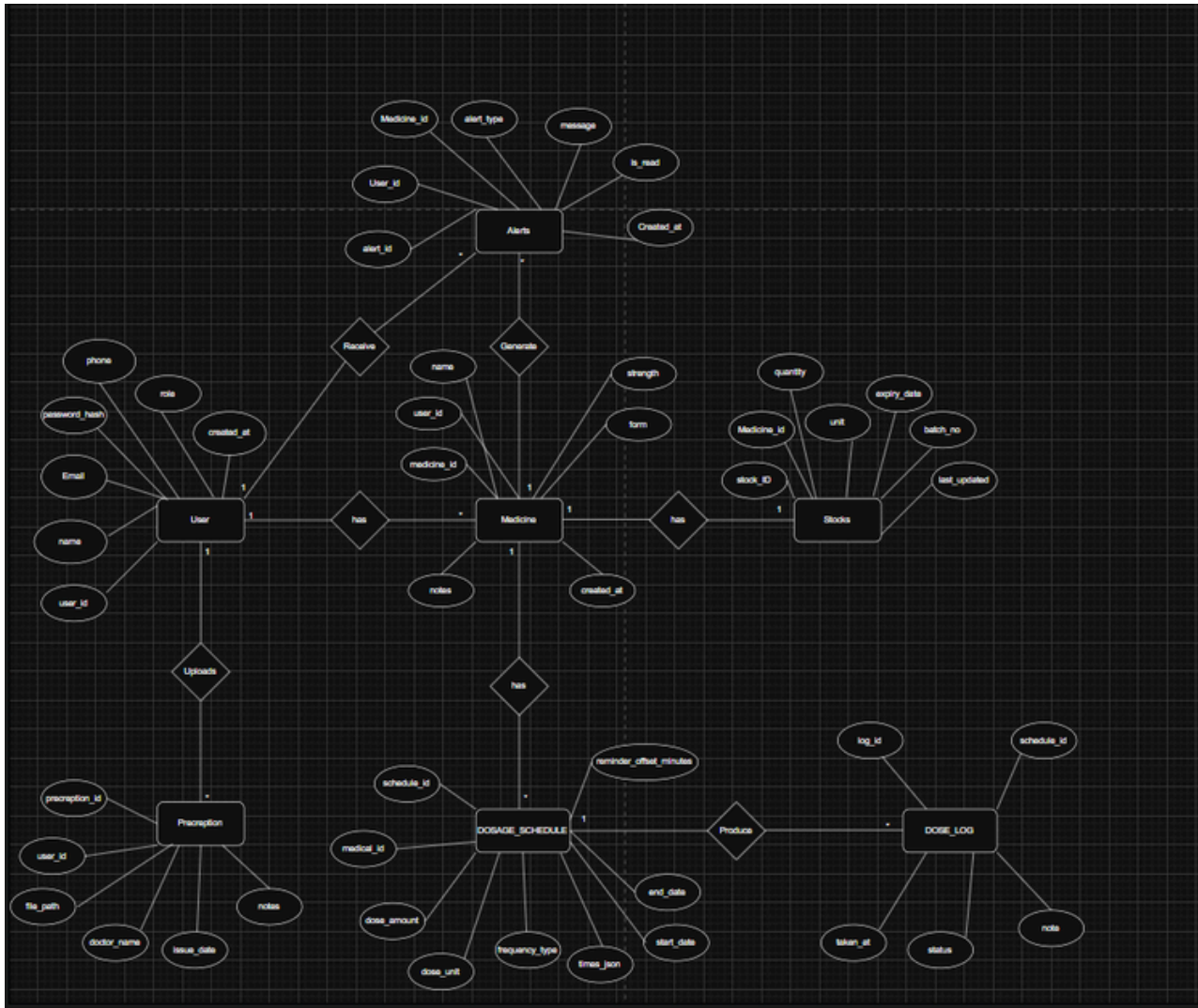
7. ALERTS

Column	Data Type	Constraints
alert_id	INT	PRIMARY KEY, AUTO_INCREMENT
user_id	INT	FOREIGN KEY → users.user_id, NOT
medicine_id	INT	FOREIGN KEY → medicines.medicin
alert_type	ENUM('low_stock', expiry','missed_do	NOT NULL
message	TEXT	NOT NULL
is_read	BOOLEAN	DEFAULT FALSE
created_at	TIMESTAMP	DEFAULT CURRENT_TIMEST

RELATIONSHIP SUMMARY

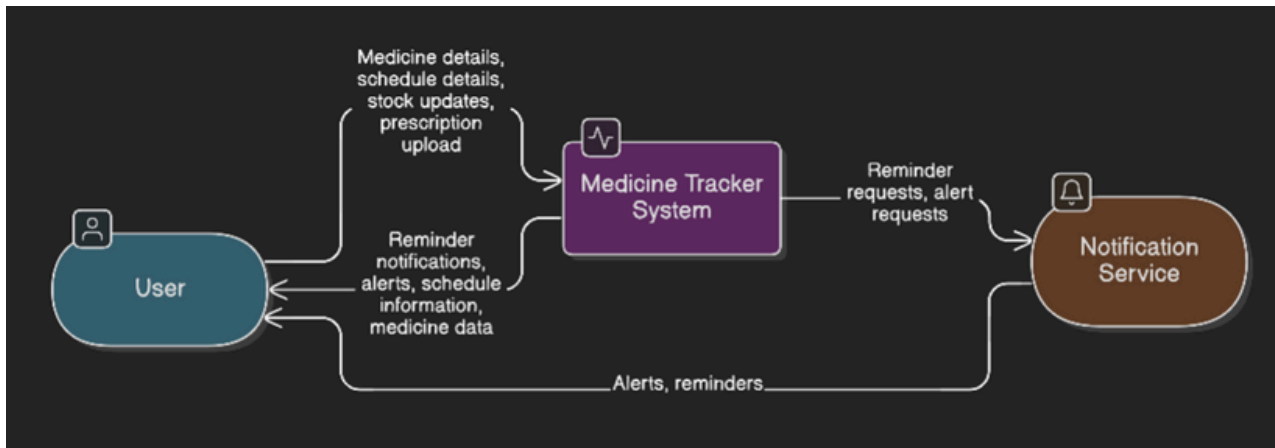
- USERS (1) → MEDICINES (∞)
- MEDICINES (1) → STOCK (1)
- MEDICINES (1) → DOSAGE_SCHEDULE (∞)
- DOSAGE_SCHEDULE (1) → DOSE_LOG (∞)
- USERS (1) → PRESCRIPTIONS (∞)
- USERS (1) → ALERTS (∞)
- MEDICINES (1) → ALERTS (∞)

ER Diagram

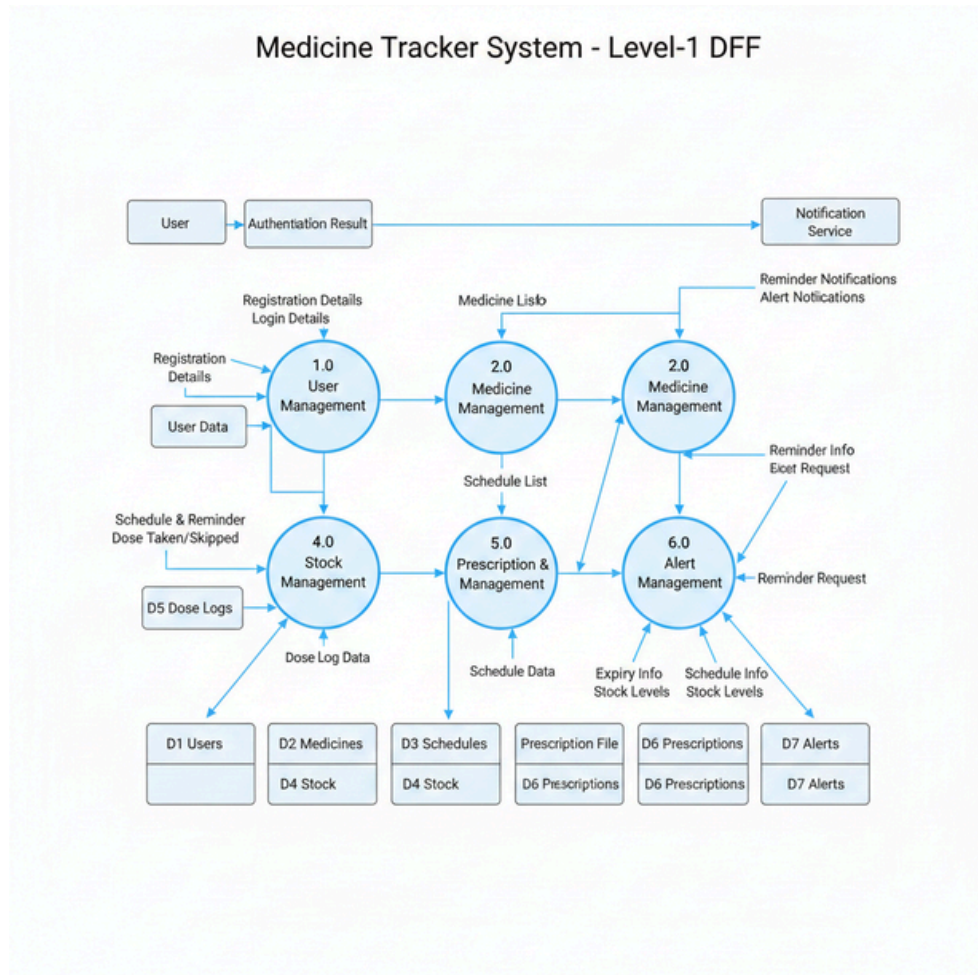


Data Flow Diagram

0-Level:



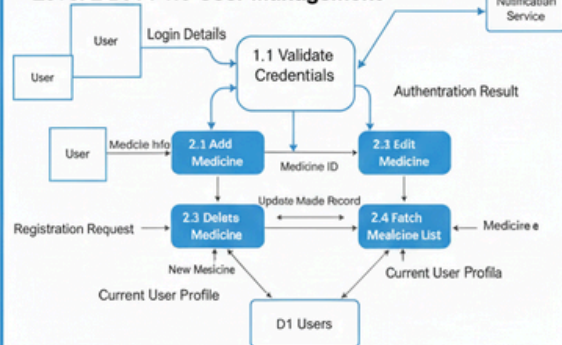
1-Level:



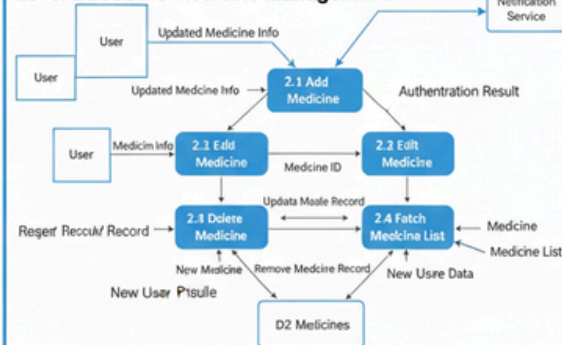
2-Level:

Medicine Tracker System - All Level 2 DFDs

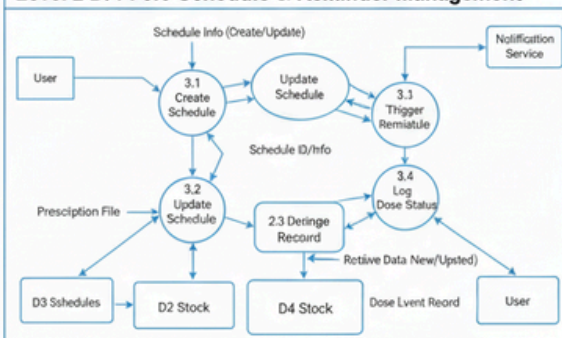
Level 2 DFF: 1.0 User Management



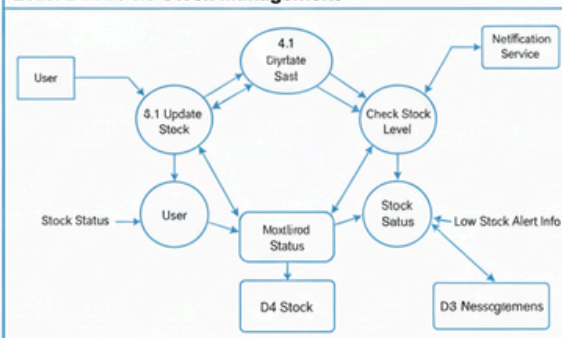
Level 2 DFF: 2.0 Medicine Management



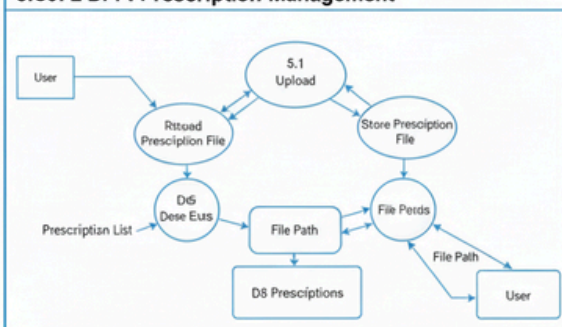
Level 2 DFF: 3.0 Schedule & Reminder Management



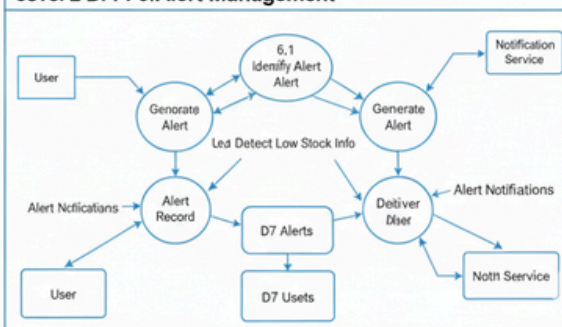
Level 2 DFF: 4.0 Stock Management



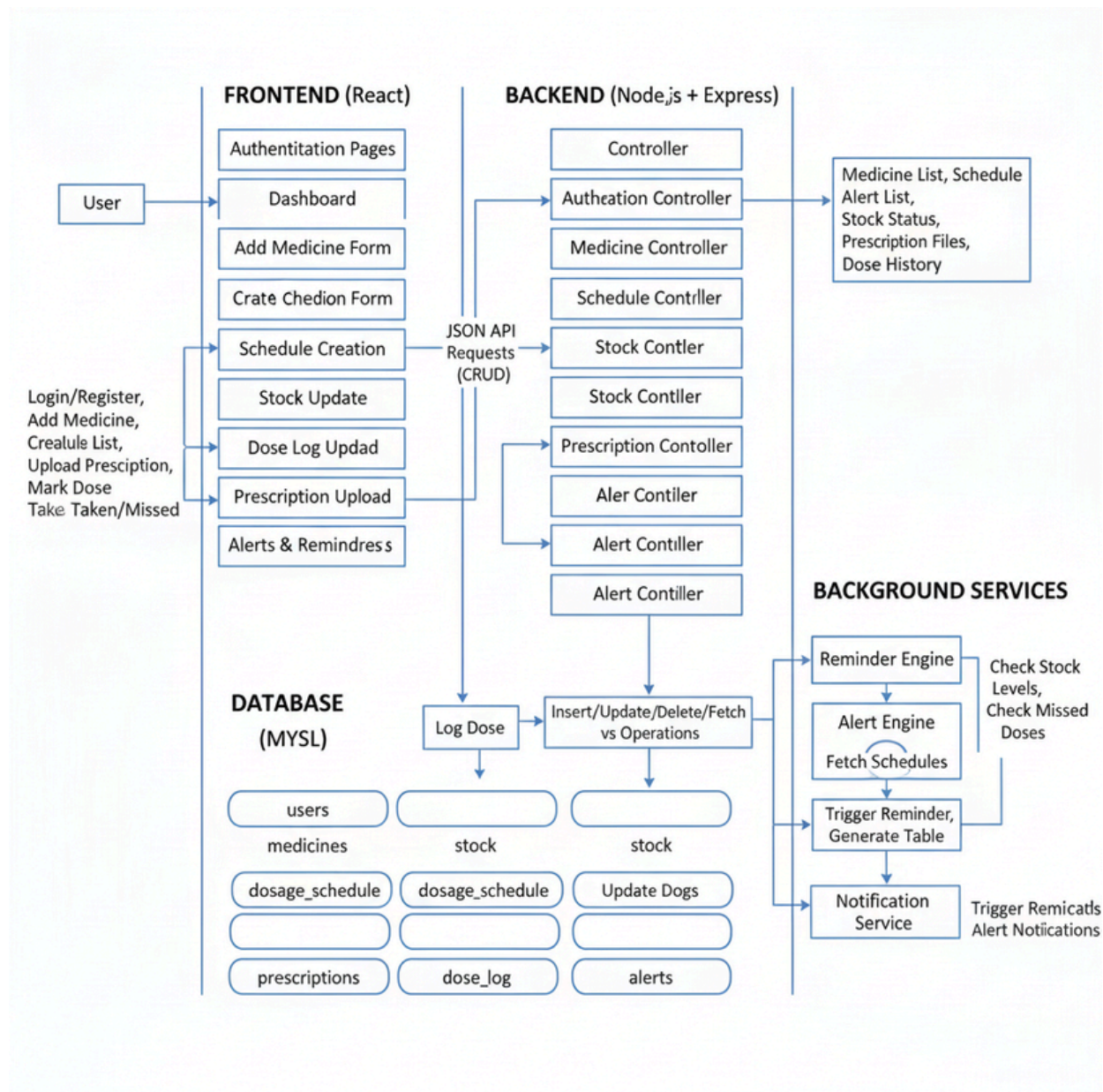
Level 2 DFF: 5.0 Prescription Management



Level 2 DFF: 6.0 Alert Management



Process Logical Diagram



Platform Used:

Hardware Requirements:

Standard PC or smartphone; minimum 4 GB RAM recommended for development and testing.

Software Requirements:

- Frontend: React.js (create-react-app / Vite)
- Backend: Node.js + Express
- Database: MySQL
- Background jobs: node-cron or Bull + Redis
- Notifications: Web Push API, SMTP/email service, SMS provider
- Tools: Git, Postman, VS Code

Future Scope

- AI-based drug interaction checks and warnings.
- Integration with e-pharmacies for automatic reorder suggestions.
- Doctor or clinic portal for prescription issuance and review.
- Family accounts with shared medicine lists and role-based access.
- Exportable reports (PDF/CSV) and printer-friendly views.

Bibliography

- [MySQL Official Documentation](#)
- [React Official Documentation](#)
- [Node.js & Express Documentation](#)
- [Web Push and Notification API Guides](#)
- [Healthcare IT Standards & Best Practices](#)