**Department of Computer Science**

**CS – 2nd Year – A**

**Course No : CS-458**

**Team Orbit**

**Team Members:**

**Husain Ahmad Khan B23110006045**

**Muhammad Hamza Abbas B23110006096**

**Muhammad Hasan Khan B23110006099**

**Muhammad Mujtaba B23110006107**

**Rayyan Ibrahim B23110006137**

**Project Plan**

1. **Project Title:** MedMate – Smart Prescription & Medicine Tracker
2. **Project Objectives:** MedMate aims to simplify how families and individuals manage their daily medications. By offering digital prescription tracking and timely reminders, it reduces the chances of missed doses and health risks.
3. **Project Scope:**

**INCLUDED:**

These are the things we will deliver and develop:

**1. User Registration and Management**

Sign-Up and Login: Users will be able to create an account and securely log in using their email and password.User Roles: Different types of users will be able to access different parts of the app:

Regular users (patients or caregivers)Admin users (managing the backend system)

**2. Prescription Management**

Prescription Upload: Users can upload their prescriptions in either image form (like a photo of a paper prescription) or through a digital form.

Store and Organize Prescriptions: All prescriptions uploaded by users will be stored securely. They will be organized by user name, date of upload, and other basic details like the doctor’s name.

View, Update, and Delete: Users will be able to view their prescriptions, make simple edits (like correcting details), and delete them if needed.

**3. Medication Reminder System**

Set Reminders: Users can set reminders for taking their medication at specific times, so they don’t forget to take their doses.

Multiple Reminders: Users can set multiple reminders for each prescription if they need to take the medicine more than once a day.

Notifications: The app will send users reminders via email or through the app itself (e.g., a push notification).Mark as Taken: Users can mark their medication as "taken" or "snooze" the reminder if they are unable to take it at the scheduled time.

**4. Family Profile Management**

Multiple Family Member Support: Users can manage prescriptions for multiple people in their family (for example, children or elderly parents) from a single account.

Easy Profile Management: Caregivers will have an easy way to track medications for everyone in their family from one dashboard.

**5. Frontend (What the User Sees)**

Responsive Design: The app will be designed to work well on computers and mobile devices (like phones or tablets), so users can access it from any device.

Simple Pages: We will have basic pages like:

Login/Signup Page

Dashboard (where users can see all their prescriptions and reminders)

Profile Page (where users manage their family members)

Prescription Management Page (where prescriptions can be uploaded and viewed)

Reminder Setup Page

**6. Backend (What Happens Behind the Scenes)**

Secure Authentication: The system will securely handle user logins using a technology called JWT (JSON Web Token) to ensure that only the right people access the system.

Data Management: The backend will manage the data such as prescriptions, reminders, and user profiles. This will be stored in a database (MongoDB).

APIs for Communication: The backend will have APIs (programming interfaces) that allow the frontend (what users see) to communicate with the data stored in the database.

**7. Admin Panel**

Manage Users: Admin users will be able to view and manage the users in the system (but they will not be able to edit or delete user data).

Monitor Prescription Activity: Admins can see basic records of prescription uploads but won't have access to sensitive health data.

**8. Security Features**

Encryption: The app will use strong encryption to ensure that prescription data is stored safely and that no one can access it without permission.

Privacy: We will follow basic security rules (such as HTTPS for all communications) to ensure user data is safe.

**9. Deployment**

Where It Will Live: The app will be deployed to the internet using reliable platforms like Vercel (for the frontend) and Heroku or Render (for the backend).

Easy Setup: The app will be easy for users to access through a web browser, without any complex setup on their end.

**10. Documentation & Handover**

Guides and Instructions: After the app is built, we will provide user guides and basic technical documentation explaining how everything works.

Source Code Handover: The complete code for the app will be handed over to the client.

**❌ EXCLUDED FROM SCOPE**

These are things that will not be included in this project:

**1. Mobile Applications**

No Mobile App: The project will only be a web-based application and will not include mobile apps for Android or iOS.

**2. Doctor/Clinic Interaction**

No Prescription Issuing: Doctors will not be able to issue prescriptions through this app. Users will only be able to upload and manage their existing prescriptions.

No Doctor Portal: There will be no doctor-specific features for managing prescriptions.

**3. Third-Party Integration**

No Integration with Pharmacies: The app will not connect with external systems such as pharmacies or hospitals.

No External Data Sync: The app will not sync with third-party systems (like pharmacy or insurance systems).

**4. E-commerce Features**

No Payments: The app will not allow users to buy medications online. There will be no shopping cart or payment systems.

**5. Multiple Languages**

English Only: The app will be developed in English and will not support other languages.

**6. Advanced Features**

No Artificial Intelligence (AI): There will be no AI-powered health advice or recommendations. The app will only remind users about their medication.

No Prescription Scanning or OCR: The app will not automatically read prescriptions from images.

**7. Legal & Compliance Features**

No Full Medical Compliance: We will include basic privacy features (such as encryption), but the app will not be fully HIPAA or GDPR-compliant (which would require legal audits and certification).

**8. Marketing & Branding**

No Branding: The app won’t include custom branding or a marketing website. It will be focused purely on the functionality.

No SEO/Analytics: The app won’t have search engine optimization (SEO) or integrated analytics tools.

**9. Post-Deployment Support**

No Ongoing Maintenance: We will not provide ongoing updates or 24/7 support after deployment.No Long-Term Hosting Management: The hosting will be on platforms like Vercel/Heroku, but no long-term management or custom server setup is included.

**10.** **Data Migration**

No Migration from Other Systems: If users have data stored elsewhere, it will not be transferred to this app as part of this project.

4**. Work Breakdown Structure:**

### 1. Project Planning & Research

1.1 Requirement Gathering  
1.2 Stakeholder Identification  
1.3 Scope Definition  
1.4 Technical Feasibility Study  
1.5 Project Timeline & Milestones

**2. UI/UX Design**

2.1 Wireframing (Home, Dashboard, Reminders, etc.)  
2.2 User Journey Mapping  
2.3 Mobile-Responsive Layouts  
2.4 High-Fidelity UI Design (Figma/Adobe XD)  
2.5 User Feedback Collection (Optional)

**3. Frontend Development (React.js)**

3.1 Project Setup with React & Routing  
3.2 Component Development  
 Header, Footer, Login, Signup, Dashboard  
 Prescription Upload & Tracker  
 Reminder & Notification Views  
 Medicine Reorder Interface  
3.3 State Management (Redux or Context API)  
3.4 Form Validation & Error Handling  
3.5 Integration with Backend APIs  
3.6 Testing & Responsiveness

**4. Backend Development (Node.js + Express.js)**

4.1 Project Initialization & API Structure  
4.2 Authentication & Authorization (JWT)  
4.3 RESTful APIs for:  
 Users (Signup/Login)  
 Prescription Management  
 Reminder Scheduling  
 Order Requests  
4.4 Role Management (User/Admin)  
4.5 Error Handling & Middleware  
4.6 Backend Testing (Postman, etc.)

**5. Database Management (MongoDB/PostgreSQL)**

5.1 Schema Design (Users, Prescriptions, Reminders, Pharmacies)  
5.2 Relationship Mapping (if required using references)  
5.3 CRUD Operation  
5.4 Data Validation and Indexing  
5.5 Backup Strategy (Optional)

**6. Admin Panel**

6.1 User Management  
6.2 Monitor Prescription Logs

**7. Testing & Debugging**

7.1 Unit Testing (Frontend & Backend)  
7.2 Integration Testing  
7.3 UI Testing (Responsive, Mobile View)  
7.4 Bug Fixing & QA Review

**8. Deployment & Handover**

8.1 Final Build and Optimization  
8.2 Deployment on Vercel/Netlify (Frontend)  
8.3 Deployment on Vercel/Heroku (Backend)  
8.4 MongoDB Atlas/PostgreSQL Setup  
8.5 Final Documentation (README, SRS, Guide)  
8.6 Handover & Final Presentation

1. **Schedule and Milestones:**

|  |  |
| --- | --- |
| Milestones | Target Date |
| Planning and Design | Week 1 |
| UI/UX Finalization & Project Setup | Week 2 |
| Frontend Development | Week 3 |
| Backend Development | Week 4 |
| Frontend-Backend Integration | Week 5 |
| **Admin Panel + Core Logic** | Week 6 |
| Testing & Deployment | Week 7 |

1. **Risk Identification and Mitigation:**

|  |  |  |
| --- | --- | --- |
| Risk Description | Impact Level(H/M/L) | Mitigation Strategy |
| Data Breaching | High | Use strong encryption |
| **Server Crash** | High | Deploy backend on a reliable cloud platform |
| Project Timeline Slippage | Medium | Break development into weekly sprints with clear milestones |
| Browser Compatibility | Low | Test UI on major browsers and devices |
| Lack of User Adoption | High | Promote the platform through awareness sessions |
| Data Loss Due to Improper DB Handling | Medium | Use proper validations |

1. **Roles and Resources:**

|  |  |
| --- | --- |
| Role/Resource | Team Member |
| Project Manager | Muhammad Hasan Khan |
| UI/UX Designer | Rayyan Ibrahim |
| Lead Developer | Muhammad Mujtaba |
| QA Tester | Husain Ahmad Khan |
| Deployment | Muhammad Hamza Abbas |

1. **Communication Plan:**

* Use **Slack** for daily communication among the development team and **email** for formal updates.
* Conduct **weekly sync meetings via Google Meet** to discuss progress.
* Organize **review sessions at each project milestone** to demonstrate completed features and gather feedback.
* Share **weekly progress reports** via email and maintain daily status updates on Slack.

1. **Change Control Process:**

* **Change Request Submission**  
  Any team member or stakeholder can submit a **Change Request (CR)** through a standardized form, detailing the proposed change, reason, and expected impact.
* **Initial Review**  
  The **Project Manager (PM)** reviews the request for clarity, relevance, and alignment with project scope and timeline.
* **Impact Analysis**  
  The development and design leads evaluate how the change affects **cost, timeline, technical feasibility, and resources**.
* **Approval/ Rejection**  
  The **Change Control Board (CCB)** consisting of the PM and key team leads decides whether to approve, reject, or defer the change.
* **Implementation & Documentation**  
  Approved changes are scheduled, implemented, and tracked. All modifications are recorded in a **Change Log**, ensuring traceability.