**HEALTHAIDE**

System Documentation

1. **Introduction**

**Project Name:** HealthAide

**Project Overview:** HealthAide is a comprehensive health management system designed to assist users in booking doctor's appointments, tracking health progress, setting medication and appointment reminders, enabling virtual consultations with personal doctors, ordering medicines with home delivery, providing emergency support, and facilitating services payments.

**Objective:** To provide a seamless and user-friendly platform for managing personal health and medical needs.

**Stakeholders:** Patients, doctors, healthcare providers, Administrators, Logistics Team .

**2. Requirements**

**2.1 Functional Requirements**

* **User Registration and Login:**
  + Users can create an account using their email and password.
  + Users can log in and log out securely.
* **User Profiles:**
  + Users can view and edit their profiles.
  + Profiles display user health information, appointments, and medication schedules.
* **Doctor Appointment Booking:**
  + Users can search for doctors and book appointments.
  + Users receive confirmation and reminders for upcoming appointments.
* **Health Progress Tracking:**
  + Users can log and track their health metrics (e.g., weight, blood pressure, glucose levels).
  + Users can view health progress over time through charts and graphs.
* **Medication and Appointment Reminders:**
  + Users can set reminders for taking medication and upcoming appointments.
  + Users receive notifications for reminders.
* **Virtual Consultations:**
  + Users can have virtual consultations with their personal doctors via video calls.
  + Users can exchange messages with doctors for follow-up questions.
* **Medicine Ordering and Home Delivery:**
  + Users can order prescribed medicines through the app.
  + Users can choose home delivery options.
* **Emergency Support:**
  + Users can access emergency support services through the app.
  + Users can call emergency numbers or request immediate assistance.
* **Services Payment:**
  + Users can make payments for consultations, medicines, and other services.
  + Users receive invoices and payment confirmations.

**2.2 Non-Functional Requirements**

* **Performance:**
  + The system should handle multiple concurrent users without performance degradation.
* **Usability:**
  + The application should be user-friendly and intuitive.
* **Security:**
  + User data should be securely stored and transmitted.
* **Scalability:**
  + The system should be scalable to handle increased user load.
* **Availability:**
  + The application should be available 24/7 with minimal downtime.

**3. System Design**

**3.1 Architecture**

* **Frontend:**
  + Developed using Flutter and Dart. For the mobile app
  + A Responsive for both web and mobile Will be developed using HTML, CSS and JavaScript.
* **Backend:**
  + **Django/Dart for web backend**
  + Firebase for authentication, database, and storage.
  + Cloud Functions for server-side logic.
* **Database:**
  + Firestore for storing user data, health metrics, appointments, and messages.

**3.2 Data Flow**

1. User registers and logs in.
2. User books a doctor's appointment.
3. User logs health metrics and sets reminders.
4. User orders medicines and chooses delivery options.
5. User has a virtual consultation with a doctor.
6. User accesses emergency support if needed.
7. User makes payments for services.

---------- The Data Flow Diagram to be added here -------------------

**4. Modules and Features**

**a. Patients Modules**

* **User Management Module**
* **Features:**
  + User registration
  + User login/logout
  + User profile management
* **Appointment Booking Module**
* **Features:**
  + Search for doctors
  + Book appointments
  + Appointment confirmations and reminders
* **Health Tracking Module**
* **Features:**
  + Log health metrics
  + View health progress through charts and graphs
* **Reminder Module**
* **Features:**
  + Set medication and appointment reminders
  + Receive notifications for reminders
* **Virtual Consultation Module**
* **Features:**
  + Schedule and have video calls with doctors
  + Exchange messages with doctors
* **Medicine Ordering Module**
* **Features:**
  + Order prescribed medicines
  + Choose home delivery options
* **Emergency Support Module**
* **Features:**
  + Access emergency support services
  + Call emergency numbers or request assistance
* **Payment Module**
* **Features:**
  + Make payments for consultations, medicines, and services
  + Receive invoices and payment confirmations
    1. **Doctors Modules**
* Appointment Management Module
* Features:
  + View and manage appointments
  + Confirm or reschedule appointments
  + Receive reminders for upcoming appointments
* Patient Records Module
* Features:
  + Access patient health records and history
  + Update patient records with new health data
* Virtual Consultation Module
* Features:
* Conduct video consultations with patients
* Exchange messages with patients for follow-up questions
* Prescription Module
* Features:
* Prescribe medicines to patients
* Send prescriptions to pharmacies for home delivery
  + 1. **Healthcare Providers (Pharmacies, Emergency Services) modules**
* Medicine Management Module
* Features:
* Receive and manage medicine orders
* Prepare medicines for home delivery
* Update order status and notify patients
* Emergency Support Module
* Features:
* Provide emergency support services
* Coordinate with emergency services for immediate assistance
* Payment Management Module
* Features:
* Process payments for ordered medicines and services
* Generate and send invoices to patients
  + 1. **Administrators Module**
* System Management Module
* Features:
* Manage system configurations and settings
* Implement updates and improvements
* Monitor system performance and security
* Support Module
* Features:
* Provide technical support to users
* Address and resolve bugs or issues reported by users

**5. Implementation Plan**

**5.1 Phase 1: Planning**

* Define project scope and requirements.
* Create project timeline and milestones.
* Assemble development team.

**5.2 Phase 2: Design**

* Create UI/UX designs and wireframes.
* Design system architecture and database schema.

**5.3 Phase 3: Development**

* Develop frontend using Flutter and Dart.
* Implement backend with Firebase.
* Integrate frontend and backend.
* Implement authentication and user management.

**5.4 Phase 4: Testing**

* Perform unit testing and integration testing.
* Conduct user acceptance testing.
* Fix any identified issues and bugs.

**5.5 Phase 5: Deployment**

* Deploy the application to a hosting platform.
* Conduct final testing on the live environment.
* Launch the application.

**5.6 Phase 6: Maintenance**

* Monitor application performance and user feedback.
* Implement updates and improvements as needed.
* Provide ongoing support and troubleshooting.

**6. Timeline**

**Week 1-2:** Planning

**Week 3-5:** Design

**Week 6-10:** Development

**Week 11-12:** Testing

**Week 13:** Deployment

**Onwards:** Maintenance

**7. Conclusion**

The HealthAide system aims to provide a seamless platform for managing personal health and medical needs. By following the Waterfall methodology, we ensure a structured and efficient development process from planning to maintenance.