ABSTRACT / OPTIONS TOO BE IMPLEMENTED IN THE APP:

Building UHI EUA App

This details out the steps to build your own EUA app including

* Sign up / Sign in with PHR Address
* Select a UHI service
  + Teleconsultations
    - Search HSPAs for Doctor
    - Display search results
    - Book selected Doctor/Facility
    - Collect Payment if required
    - Confirm Booking
    - Exchange Messages with Doctor
    - Share health records
    - Setup WebRTC (tele consults)
    - Initiate Call (tele consults)
    - Get final prescription
  + Appointment booking
    - Search HSPAs for Facility/Doctor
    - Display search results
    - Book selected Facility/Doctor
    - Collect Payment if required
    - Confirm Booking
  + Ambulance booking
    - Search HSPAs for Facility/Doctor
    - Display search results
    - Book selected Facility/Doctor

APP STRUCTURE AND FUNCTION’S:

Building a User Health Information (UHI) EUA App involves several key components and workflows. Below are detailed ideas and working conditions for implementing each specified feature.

1. User Authentication

Sign Up / Sign In with PHR Address

Implementation: Use OAuth 2.0 for secure sign-up and sign-in processes. Allow users to sign in with Personal Health Record (PHR) addresses, which can be validated through email or SMS.

UI/UX Consideration: Simple and intuitive login screens with options for password recovery.

2. UHI Service Selection

- Select a UHI Service:

- Implementation: Provide a dashboard displaying available UHI services (e.g., teleconsultation, ambulance booking, etc.). Use card UI elements to make services visually appealing.

- Database: Store service details in a structured database to allow easy retrieval and updates.

3. Teleconsultations

- Search HSPAs for Doctors

- Implementation: Implement a search feature that filters doctors based on specialties, ratings, and availability.

- Data Source: Use APIs from health service provider aggregators to fetch real-time data.

\* Display Search Results

- Implementation: Present results in a list format with key details (name, specialty, rating, availability).

- UI/UX Consideration: Include sorting and filtering options to enhance user experience.

4. Booking Process

- Book Selected Doctor/Facility

- Implementation: Allow users to select a doctor/facility and choose appointment slots from available times.

- Database Update: Ensure real-time updates to prevent double bookings.

\* Collect Payment if Required

- Implementation: Integrate a payment gateway (like Stripe or PayPal) for processing payments.

- Security: Ensure compliance with PCI DSS standards to protect user data.

\* Confirm Booking

- Implementation: Send confirmation notifications via email and app notifications.

- Database Update: Mark the appointment as confirmed in the database.

5. Communication with Doctor

\* Exchange Messages with Doctor

- Implementation: Integrate a secure messaging system that allows patients to communicate with doctors before and after consultations.

- Data Security: Ensure messages are encrypted end-to-end.

\* Share Health Records

- Implementation: Allow users to upload and share their health records securely with doctors through the app.

- File Formats: Support various formats (PDF, images, etc.) for convenience.

6. WebRTC Setup for Teleconsultations

\* Setup WebRTC

- Implementation: Use WebRTC APIs to enable real-time audio/video communication.

- Server Setup: Use a signaling server to manage connections and ensure reliability.

\* Initiate Call \*

- Implementation: Provide users with a button to start a teleconsultation that connects them with the selected doctor.

- User Interface: Create an interface that allows users to mute/unmute, turn video on/off, and end the call.

7. Get Final Prescription

- Implementation: Allow doctors to send prescriptions digitally post-consultation. This can be stored in the user's profile for future access.

- Integration: Consider integrating with pharmacies to allow direct prescription fulfilment.

8. Appointment Booking

- Search HSPAs for Facility/Doctor

- Implementation: Similar to the doctor search, but expand to include facilities.

-Display Search Results & Booking

- Implementation: As above, ensure a smooth user experience.

- Collect Payment and Confirm Booking

- Implementation: Follow the same process as teleconsultations for consistency.

9. Ambulance Booking

- Search HSPAs for Facility/Doctor

- Implementation: Enable users to find nearby ambulance services.

Display Search Results & Booking

- Implementation: Provide information about available services, estimated time of arrival, and collect payments.

Confirm Booking

- Implementation: Send confirmation details and an estimated time of arrival to the user.

Additional Considerations

- Compliance: Ensure the app complies with health regulations such as HIPAA (in the U.S.) or GDPR (in Europe).

- User Feedback: Implement a feedback mechanism post-consultation to continually improve services.

- Accessibility: Ensure the app is accessible to all users, including those with disabilities.

Technology Stack

- Frontend: Flutter for mobile; React or Angular for web.

- Backend: Node.js with Express for REST APIs; Python with Django or Flask as alternatives.

- Database: MongoDB or PostgreSQL for data storage.

- Messaging: Firebase Cloud Messaging or Socket.io for real-time messaging.