

# Second Opinion App - Architecture Document

## 1. Application Architecture

Second Opinion follows a **Microservices Architecture** to ensure modularity and scalability. The system is divided into the following services:

- **User Service:** Handles authentication, user profiles, and medical history preferences.
- **Medication [REDACTED]:** Manages the database of medications, their effects, interactions, and usage guidelines.
- **Consultation Service:** Processes user queries about medications and provides information.
- **Notification Service:** Sends medication reminders and important health alerts.
- **[REDACTED] Gateway:** Manages requests between the frontend and backend services.

## 2. Database

- **ER Diagram:** Represents relationships between users, medications, consultations, and reminders.

[Placeholder for ER Diagram showing relationships between users, medications, interactions, and consultations]

- **Schema Design:**
  - **Users** (id, name, email, password\_hash, age, medical\_conditions, allergies, current\_medications)
  - **Medications** (id, name, generic\_name, category, usage, side\_effects, interactions, contraindications, dosage\_info)

- **UserMedications** (id, user\_id, medication\_id, prescribed\_date, dosage, frequency, notes)
- **Consultations** (id, user\_id, medication\_id, question, response, created\_at)
- **Reminders** (id, user\_medication\_id, reminder\_time, notification\_sent)

### 3. Data Exchange Contract

- **Frequency of data exchanges:**
  - Medication information is updated regularly from trusted pharmaceutical databases.
  - User consultations are processed in real-time.
  - Medication reminders are scheduled according to user preferences.
- **Data Sets:**
  - User data (for authentication, medical history, and preferences)
  - Medication data (comprehensive information about medicines)
  - Consultation data (user queries and system responses)
- **Mode of Exchanges:**
  - **API:** RESTful APIs for medication information and user management.
  - **WebSockets:** For real-time consultation responses.
  - **Push Notifications:** For medication reminders and important alerts.

### 4. Deployment Architecture

- **Frontend:** [REDACTED] Native mobile app and [REDACTED]-based web application.
- **Backend:** [REDACTED] ([REDACTED].js) hosted on [REDACTED] or [REDACTED].

- **Database:** PostgreSQL for structured medical data with [REDACTED] for unstructured consultation data.
- **Authentication:** JWT-based authentication with additional security for healthcare data.
- **Storage:** [REDACTED]-compliant cloud storage for user medical information.
- **Third-party Integrations:** APIs for medication databases (like [REDACTED], [REDACTED], etc.)

## 5. Security and Compliance

- [REDACTED] compliance for handling sensitive medical information
- End-to-end encryption for all user data
- Regular security audits and penetration testing
- Data anonymization for analytics purposes

## UML Diagrams

### Sequential Diagram for Medication Consultation Flow

[Placeholder for Sequential Diagram showing the flow from user query about a medication to system response with relevant information]



