**Data Management Plan (Iteration 2) ‐ Updated**

**Link :** [Data Management Plan (Iteration 2) ‐ Updated · Tejaswireddyallam/GDPFall2024-Group2 Wiki](https://github.com/Tejaswireddyallam/GDPFall2024-Group2/wiki/Data-Management-Plan-(Iteration-2)-%E2%80%90-Updated)

**Project Title : Telemedicine Application**

**Data Summary**

The Telemedicine Application will store and manage the following types of data:

**Patient**

* PatientID (P.K)
* Name
* Email
* Password
* Contact Info
* Role (Patient)
* Address

**Doctor**

* DoctorID (P.K)
* Name
* Email
* Password
* Contact Info
* Availability
* Role (Doctor)
* Specialization
* Qualifications

**Appointments**

* AppointmentID (P.K)
* PatientID (F.K) (References User)
* DoctorID (F.K) (References User with Role = Doctor)
* AdminID (F.K)
* Date & Time
* Status (Confirmed, Cancelled, Rescheduled)

**Medical Records**

* RecordID (P.K)
* PatientID (F.K) (References User with Role = Patient)
* DoctorID (F.K) (References User with Role = Doctor)
* AdminID (F.K)
* Consultation Notes
* Upload File Path (for records, reports, etc.)

**Prescriptions**

* PrescriptionID (P.K)
* DoctorID (F.K)
* PatientID (F.K)
* AppointmentID (F.K)
* AdminID (F.K)
* Medication Details
* Date
* Dosage

**Messages**

* MessageID (P.K)
* DoctorID (F.K) (References User)
* PatientID (F.K) (References User)
* AdminID (F.K)
* Content
* Timestamp

**Notifications**

* NotificationID (P.K)
* PatientID (F.K)
* DoctorID (F.K)
* AdminID (F.K)
* Content

**Pharmacy**

* PharmacyID (P.K)
* Name
* Location
* Contact Info

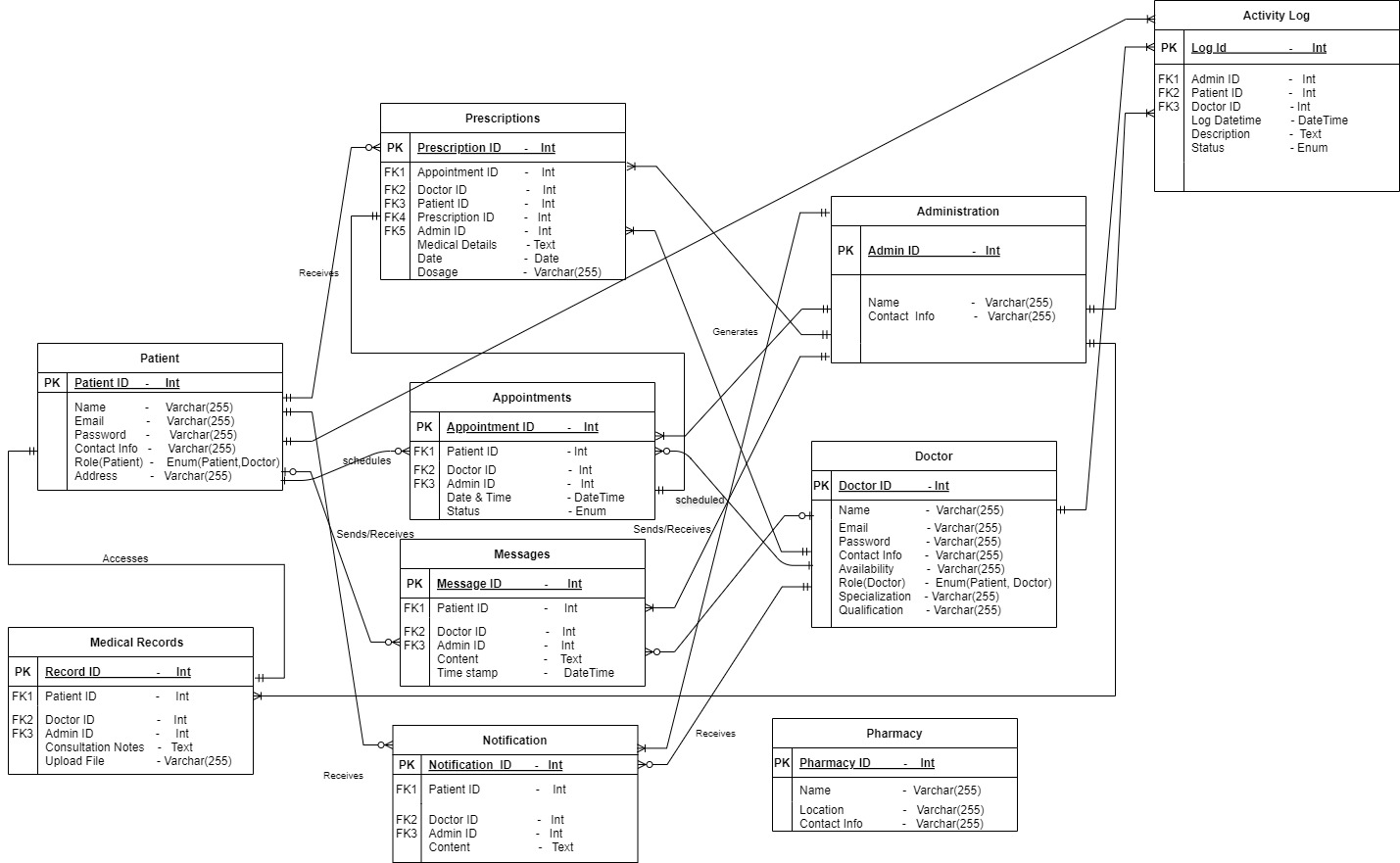
**Administration**

* AdminID (P.K)
* Name
* Contact Info

**Activity Log**

* LogID(P.K)
* PatientID (F.K)
* DoctorID (F.K)
* AdminID (F.K)
* Log Datetime
* Description
* Status

**ER Diagram**



**Initial Plans to Secure Data**

**1. Access Restriction**

**Role based Access Restriction**

* Patients will only access their own records and messages.
* Doctors will access the records and appointments of their assigned patients.
* Different roles (e.g., doctor, patient, admin) will have specific access rights. Only authorized users will access certain information, such as doctors accessing patient medical records but patients not accessing doctor credentials.

**Multi-factor Authentication (MFA):**

* To add an extra layer of security, the system will require users to verify their identity using an additional method (e.g., one-time password or phone authentication).

**2. Encryption**  
All communication between the server, client, and external systems (e.g., pharmacy or other services) will be secured using Transport Layer Security (TLS) to ensure data is encrypted during transmission over the network. This protects against man-in-the-middle attacks.

**3. Compliance**  
Ensure the application meets regulatory compliance (e.g., HIPAA) by maintaining strict access control, encryption, audit trails, and regular compliance checks.

**4. Database Security**  
Only authorized services and applications can connect to the database. Connections will use strong authentication methods (e.g., API keys, tokens, or SSL certificates).

**5. Password Security**

* Passwords will never be stored in plain text. Instead, they will be securely hashed using algorithms like bcrypt or Argon2 with salt to prevent brute force attacks.
* Password policies (minimum length, complexity, periodic expiration) will be enforced.

**Mapping of Functional Requirements to Data Storage**

**1. User Registration and Management**

* Table: Patient, Doctor
* Fields:
  + Patient.PatientID, Doctor.DoctorID (to uniquely identify users)
  + Patient.Name, Doctor.Name, Contact Info, Email (to store user details)
  + Role (to distinguish between patients and doctors)
  + Password (for login, securely hashed)
  + Doctor.Specialization, Doctor.Qualification, Doctor.Availability (for doctor profile and availability)
  + Updating qualifications, contact info, etc. would update the relevant fields in the Doctor and Patient tables.

**2. Appointments Management**

* Table: Appointments
* Fields:
  + AppointmentID, PatientID, DoctorID, Date & Time, Status (Confirmed, Cancelled, Rescheduled)
  + Bookings, rescheduling, and cancellations will update the Date & Time and Status fields accordingly.

**3. Search Functionality**

* Table: Doctor
* Fields:
  + Doctor.Name, Doctor.Specialization(to store searchable doctor information)

**4. Video and Audio Communication**

* This functionality doesn’t directly map to data storage but would use live streaming infrastructure.

**5. Prescription Management**

* Table: Prescriptions,Medical Records
* Fields:
  + PrescriptionID, AppointmentID, DoctorID, PatientID, Medication Details, Dosage, Date
  + Uploaded reports and prescriptions are stored in the Upload File Path field in Medical Records.

**6. Pharmacy List**

* Table: Pharmacy
* Fields:
  + PharmacyID, Pharmacy.Name, Pharmacy.Location, Pharmacy.Contact Info
  + This data will be fetched to display the list of pharmacies.

**7. Secure Messaging Platform**

* Table: Messages
* Fields:
  + MessageID, SenderID, ReceiverID, Content, Timestamp
  + These fields store message details. Sender and receiver are referenced from either the PatientID or DoctorID.

**8. Store Medical Records**

* Table: Medical Records
* Fields:
  + RecordID, PatientID, DoctorID, Consultation Notes, Upload File Path
  + Medical records are stored securely, and access is controlled via role-based restrictions.

**9. Appointment Reminders and Notifications**

* Table: Appointments, Notifications
* Fields:
  + Reminders will reference Date & Time in Appointments.
  + Notification delivery details may be stored in a separate notification table.