



Virtual Healthcare Service Management Database Design

Group 10 – *Anjana Nandam | Vaibhavi Shinde | Sahithi Dachepally | Saideepak Reddy*

1. Introduction

This document outlines the design for a Virtual Healthcare Service Management Database. As virtual consultations become more popular, especially in light of global health crises, there's a growing need for an efficient database to manage patient data, appointments, medical records, and prescriptions. The goal of this database is to streamline these processes, making it easier for healthcare providers to offer high-quality care remotely. It also ensures data accuracy, secure handling of medical records, and a better overall patient experience.

2. Business Problems Addressed

The Virtual Healthcare Service Management Database is built to solve the following key challenges:

- **Appointment Management:** Ensuring appointments are scheduled efficiently by coordinating patient availability with doctor schedules.
- **Medical Records Management:** Organizing comprehensive medical histories to support accurate diagnoses and treatment plans during consultations.
- **Prescription Handling:** Tracking electronic prescriptions, from issuance to medication delivery, ensuring patients receive the correct medicines on time.
- **Monitoring and Reporting:** Allowing healthcare administrators to track consultation outcomes, patient feedback, and medication usage, and generate activity reports.
- **Data Accuracy and Efficiency:** Reducing errors in data handling to enhance the overall efficiency of virtual healthcare services.



3. Entities Description

3.1 Patient

- Attributes: patient_id, name, contact_info, medical_history
- Relationships: A patient can have multiple appointments, medical records, and prescriptions. They can also provide feedback on consultations.

3.2 Doctor

- Attributes: doctor_id, name, specialty, availability
- Relationships: A doctor can manage multiple appointments and issue multiple prescriptions. They work in departments and receive feedback based on their consultations.

3.3 Appointment

- Attributes: appointment_id, appointment_time, patient_id, doctor_id, consultation_room_id
- Relationships: Each appointment is linked to a patient, doctor, and consultation room, and can include feedback from the patient.

3.4 Medical Record

- Attributes: record_id, patient_id, diagnosis, treatment_plan, date
- Relationships: Each medical record is associated with a patient and accessed during appointments.

3.5 Prescription

- Attributes: prescription_id, patient_id, doctor_id, issue_date
- Relationships: A prescription is issued by a doctor for a specific patient and can contain multiple medications.



3.6 Medication

- Attributes: medication_id, medication_name, price, usage_instructions
- Relationships: Medications are included in prescriptions and linked to inventory for stock management.

3.7 Department

- Attributes: department_id, department_name
- Relationships: Doctors are assigned to departments, and each department can have multiple doctors.

3.8 Consultation Room

- Attributes: consultation_room_id, room_name, availability
- Relationships: Consultation rooms host appointments and are linked to both patients and doctors.

3.9 Feedback

- Attributes: feedback_id, appointment_id, rating, comments
- Relationships: Each feedback entry is linked to a specific appointment.

3.10 Medicine Inventory

- Attributes: inventory_id, medication_id, quantity, stock_date
- Relationships: The inventory tracks stock levels of medications and is linked to specific medications.

3.11 Shipment

- Attributes: shipment_id, prescription_id, shipment_date, delivery_status, tracking_number
- Relationships: The shipment is linked to a specific prescription and indicates the delivery status of the medication.



4. Entity Relationships

4.1 Patient

- Appointments: One-to-Many (A patient can have multiple appointments).
- Prescriptions: One-to-Many (A patient can receive multiple prescriptions).
- Feedback: One-to-Many (A patient can give feedback for multiple consultations).

4.2 Doctor

- Appointments: One-to-Many (A doctor can manage multiple appointments).
- Prescriptions: One-to-Many (A doctor can issue multiple prescriptions).
- Department: Many-to-One (Each doctor belongs to one department).

4.3 Appointment

- Patient: Many-to-One (Each appointment is for a single patient).
- Doctor: Many-to-One (Each appointment is conducted by a single doctor).
- Consultation Room: Many-to-Many (Appointments can occur in various rooms, and rooms can host multiple appointments).

4.4 Medical Record

- Patient: One-to-One (Each patient has one unique medical record).
- Appointment: Many-to-One (Medical records are accessed during appointments).

4.5 Prescription

- Patient: Many-to-One (A patient can have multiple prescriptions).



- Doctor: Many-to-One (A doctor can issue multiple prescriptions).
- Medication: Many-to-Many (A prescription can include multiple medications).

4.6 Medication

- Prescription: Many-to-Many (A medication can be part of multiple prescriptions).

4.7 Department

- Doctors: One-to-Many (Each department has several doctors).

4.8 Consultation Room

- Appointments: Many-to-Many (Rooms can host multiple appointments).

4.9 Feedback

- Appointment: One-to-One (Feedback is tied to each appointment).

4.10 Medicine Inventory

- Medication: One-to-One (Each medication has an inventory record to track stock levels).

4.11 Shipment

- Prescription: One-to-One (The shipment is linked to a specific prescription, indicating that each prescription can have only one associated shipment).



5. ER Diagram

