

# Project Documentation

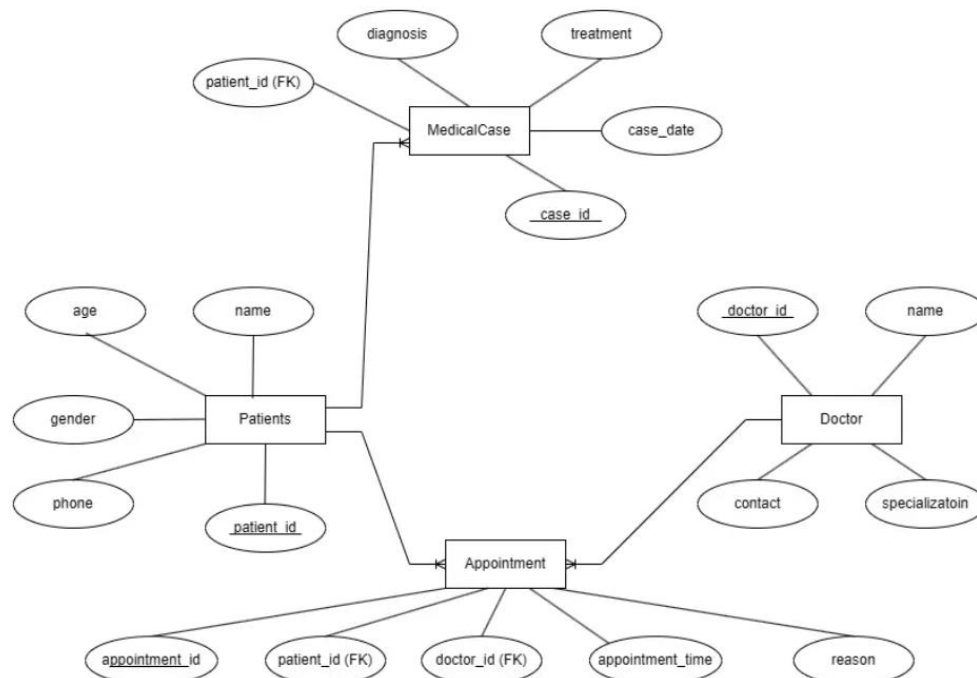
## 1. Project Architecture

### Technology Stack:

- **Backend:** Java (OOP, JDBC)
- **Database:** Oracle (normalized schema)

### Core Components:

- **Java Classes:**
  - Patient, Doctor, Appointment, Medical Case implemented using interfaces and inheritance
- **Database Schema:**
  - Normalized ER diagram including:
    - Patients
    - Doctors
    - Appointments
    - MedicalCases
- **ER Diagram**



- **CRUD Operations:**

- SQL scripts for insert, update, delete, and select
- JDBC code to connect, read, write, and update database records

**Architecture Flow:**

User Input (Console/UI) → Java Business Logic → JDBC Layer → MySQL DB

## **2. Setup Instructions**

1. Import the SQL schema into your Oracle database.
2. Compile Java classes: `javac *.java`
3. Run main program: `java MainClass`
4. Ensure the JDBC driver is correctly configured in your classpath.
5. Demonstrate:
  - Patient registration
  - Appointment creation
  - Basic record updates

### **Major Functionalities**

- **ERD & Database Setup:**  
Designed normalized tables for patients, doctors, appointments, and cases.
- **Java Object-Oriented Logic:**  
Implemented classes using interfaces, constructors, and inheritance.
- **SQL CRUD Operations:**  
Wrote scripts and connected them using JDBC for dynamic record handling.
- **Appointment Management:**  
Used Java Queue collections to manage appointment slots.
- **Timestamp Logging:**  
Implemented Java Date/Time API to track appointment creation times.