# Database Design

## Overview of the Schema

As part of EPS, we banked prescription data on a MySQL relational database with the aim of safeguarding the data while ensuring accessibility whenever dizziness clouds a physician's decision. We needed to observe the components of system design in healthcare so that the encryption of prescription records is done accurately.

Achieving these goals warranted optimization of the design for logical construction using the Third Normal Form. Duplication of data minimization of strategic and active records is a design priority. Relationships between tables and records are maintained through the use of primary and foreign keys, fostering consistency and connectivity throughout the database.

## Structure and Organization

## The eps\_db database is designed according to the relational model using MySQL system. Its main tables include Users, Patients, Prescriptions, QR Codes, Dispensing, and Audit Logs.

## Optimization is done by creating indexes on user\_id, prescription\_id, and patient\_id fields. Foreign keys use Cascade rules which ensure any changes or deletions of related data will be done in the entire system.

## The system is designed with confidentiality in mind. UUID and HMAC based secure tokens protect QR codes, and bcrypt hashed passwords are stored. Regularly scheduled automatic backups guarantee data availability and restore it in case of an emergency.

## Tables and Their Purpose

Each table in the database serves a specific role:

• Users: Keeps account details for doctors, patients, pharmacists, and administrators. Along with usernames and contact details, it stores encrypted passwords to protect sensitive information.

• Patients: Holds personal and medical information such as name, date of birth, gender, and history of treatments. Each patient is linked to a user profile.

• Prescriptions: Records the prescriptions issued by doctors. Information such as the medicine name, dosage, treatment duration, and notes are stored here, together with links to the prescribing doctor and the patient.

• QR Codes: Creates and stores secure tokens connected to prescriptions. These codes make it possible to verify that a prescription is authentic.

• Audit Logs: Tracks key system events, including user logins, new prescriptions, QR code verifications, and dispensing actions. These logs are essential for accountability and monitoring.

• Dispensing: Documents when pharmacists hand out prescribed medicines, linking back to both the prescription and the pharmacist responsible.

## Entity–Relationship Diagram (ERD)

The ERD for this system shows how all of the above tables connect. Users and Patients are linked one-to-one, while Patients can have multiple Prescriptions. Each Prescription can generate a QR Code and may appear in the Dispensing table once it is fulfilled. Audit Logs connect to many actions across the system, capturing important events for future reference.

A screenshot of a computer

AI-generated content may be incorrect.

## Tools Used

Several tools supported the design and implementation process:  
- MySQL Workbench for schema modeling and ER diagrams.  
- phpMyAdmin for day-to-day management and testing queries.  
- Draw.io and Lucidchart for building clear and shareable ER diagrams.  
- DBeaver for running SQL queries across different environments.