System Design Document:

Online Appointment and Record System

Members: Kayla Heady, Kateryna Hrishina, Carl Lazzeri

# Use Cases:

A diagram of a medical procedure

AI-generated content may be incorrect.

### Doctor Interaction Diagram:

The doctor interaction diagram displayed demonstrates the key aspects, and access points for various tasks that can be carried out through the portal. From the entry point of the web application (portal). The doctor will be able to access appointments, as well as interact with patient prescriptions stored in the remote database. It is to be mentioned that an admin panel will grant sudo access to the doctor’s access point as well.

A diagram of a diagram of a group of people

AI-generated content may be incorrect.

### Patient Interaction Diagram:

The patient interaction diagram above highlights the key utilities the web application will provide. From the login portal, a user can register an account, which will enable them to log in later via the login portal. The patients can then schedule appointments, and upload/fetch prescriptions stored from the secured remote database. The access points to the database will be described in the ER Diagram portion of the System Design Document. Highlighting the security behind the relational database procedures. It is to be noted that the admin panel will grant sudo access to the patient’s access points as well.

# ER Diagram:

A diagram of a medical application

AI-generated content may be incorrect.

### Description:

### The ER diagram defines the core structure of our **Online Appointment and Record System**,consisting of **five tables/entities**: patients, doctors, appointments, visit, and prescription. These tables support appointment scheduling, visit documentation, and prescription management between patients and doctors.

### Patients Entity:

* **Fields:** patient\_id, name, email, birthday, gender
* **Relationships:**
  + Linked to appointments (each patient can have many appointments).
  + Linked to prescription (each patient can receive multiple prescriptions).

### Doctors Entity:

* **Fields:** doctor\_id, name, specialty
* **Relationships:**
  + Linked to appointments (each doctor can have many appointments).
  + Linked to prescription (each doctor can issue many prescriptions).

#### **Appointments Entity:**

* **Fields:** appointment\_id, doctor\_id, patient\_id, appointment\_date
* **Relationships:**
  + Associates a patient and a doctor with a scheduled date/time.
  + Connected to the visit table (each appointment can have one visit summary).

#### **Visit Entity:**

* **Fields:** visit\_id, appointment\_id, visit\_summary
* **Relationships:**
  + Stores the doctor's notes and visit outcome for a specific appointment.
  + Each visit is tied to exactly one appointment.

#### **Prescription Entity:**

* **Fields:** prescription\_id, patient\_id, doctor\_id, prescription\_form, prescription\_name, quantity, refills
* **Relationships:**
  + Linked to both the patient receiving the medication and the doctor issuing it.
  + Includes detailed prescription data such as dosage form, name, quantity, and number of refills.

#### **Summary:**

#### This data model enables Patients to register and manage appointments, doctors to view appointments, submit visit notes, and write prescriptions, and a secure and relational structure for storing medical records and interactions.

# UI Mockup:

A screenshot of a computer

AI-generated content may be incorrect.

A login screen with blue and white text

AI-generated content may be incorrect.A screenshot of a login form

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

# Architecture Diagram:

A screenshot of a computer screen

AI-generated content may be incorrect.