System Design Document:

Online Appointment and Record System

Members: Kayla Heady, Kateryna Hrishnia, 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 network

AI-generated content may be incorrect.

### Description:

The ER diagram highlights the key aspects of each data type (table) we will be storing. There are a total of four key data tables: patients, doctors, appointments, and prescriptions.

#### Patient\_data:

Data fields: patient\_name, patient\_id, and patient\_email.

Interacts with appointments (scheduling system), prescriptions (fetching doctor submitted forms).

#### Appointment\_data:

Data fields: doctor\_id, patient\_id, date/time\_data.

Interacts with patients (patient\_id populated on event creation), and doctors (finds available doctor and assigns doctor via doctor\_id).

#### Doctor\_data:

Data fields: doctor\_name, doctor\_id.

Interacts with appointments (doctor assigned after appointment creation), and prescriptions (doctor inputs data into prescription form and can request local copy of form).

#### Prescription\_data:

Data fields: patient\_id, prescription\_name, prescription\_form.

Interacts with patients (patients are assigned a prescription from doctor, local copy of prescription can be requested from database) and doctors (doctors can submit prescriptions and assign a patient, doctors can request local copy of prescription form).

# UI Mockup:

# Architecture Diagram: