

Patient Appointment Management System

Project Description

This project aims to create a comprehensive appointment management system for a healthcare facility. The system will facilitate a user-friendly platform that supports both patient and doctor interactions, maintaining the integrity and usability of the platform.

The system will include the following features:

Public Features

- **View Available Slots for Appointments:** Unauthenticated users will have the ability to view available slots for appointments. This feature ensures that anyone visiting the platform can explore available appointment times without needing to log in.
- **View Doctor Profiles and Specializations:** Unauthenticated users can view profiles and specializations of doctors to make informed decisions.

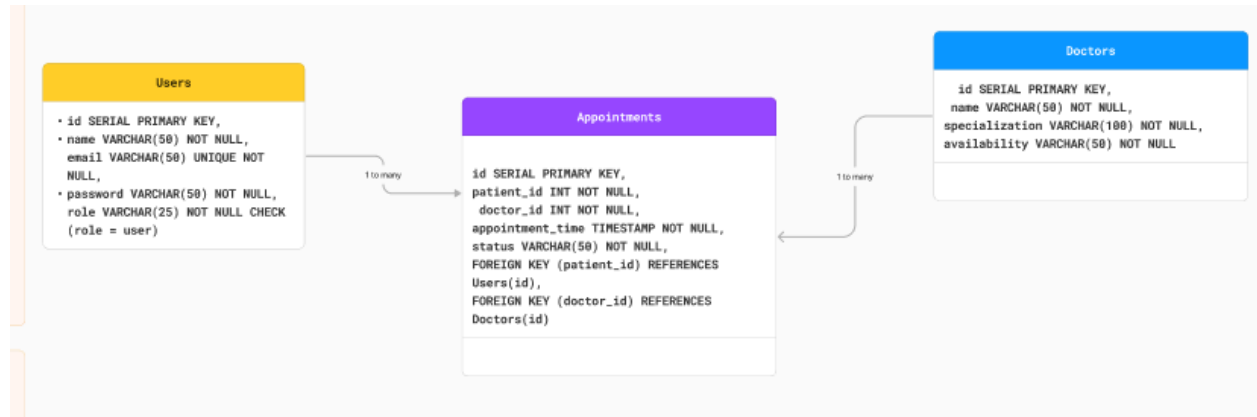
User Features

- **Book Appointments:** Authenticated patients can book new appointments with doctors. Each appointment will include the following fields:
 - **Patient ID:** The ID of the patient booking the appointment.
 - **Doctor ID:** The ID of the doctor for the appointment.
 - **Appointment Time:** The date and time of the appointment.
 - **Status:** The status of the appointment
- **Manage Appointments:** Authenticated patients can update or cancel their own appointments, allowing them to manage their personal schedules effectively.

Functional Requirements

- **As an unauthenticated user, I can:**
 - View available slots for appointments.
 - View doctor profiles and specializations.
- **As an authenticated user (patient), I can:**
 - Book an appointment with a doctor.
 - View my upcoming and past appointments.
 - Cancel my appointment.

ERD



API Endpoints

Endpoint	Method	Query Parameters	Description	Success	Error	Authentication
/doctors	GET	none	Get all doctors profiles	200	400	None
/appointments	GET	Date from, date to	Get all appointments	200	400	none
/appointments	Post	None	Book a new appt	201	400, 422	USER
/appointments/{id}	Put	None	Reschedule an appointment	200	404	User
/appointments/{id}	Delete	None	Cancel an appointment	204	401	User

```
-- database m2_final_project
BEGIN TRANSACTION;
```

```
-- *****
```

```
-- Drop all db objects in the proper order
```

```
-- *****
```

```
DROP TABLE IF EXISTS users;
```

```

-- *****
-- Create the tables and constraints
-- *****

--users (name is pluralized because 'user' is a SQL keyword)
CREATE TABLE users (
    user_id SERIAL,
    username varchar(50) NOT NULL UNIQUE,
    password_hash varchar(200) NOT NULL,
    role varchar(50) NOT NULL,
    CONSTRAINT PK_user PRIMARY KEY (user_id)
);

CREATE TABLE doctors (
    doctor_id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    speciality VARCHAR(100) NOT NULL,
    availability VARCHAR(100) NOT NULL
);

CREATE TABLE appointments (
    appointment_id SERIAL PRIMARY KEY,
    patient_id INT NOT NULL,
    doctor_id INT NOT NULL,
    appointment_time TIMESTAMP NOT NULL,
    status VARCHAR(50) NOT NULL,
    FOREIGN KEY (patient_id) REFERENCES users(user_id),
    FOREIGN KEY (doctor_id) REFERENCES doctors(doctor_id)
);

-- *****
-- Insert some sample starting data
-- *****

-- Users
-- Password for all users is password
INSERT INTO
    users (username, password_hash, role)
VALUES
    ('Yacob',
'$2a$10$tmxuYYg1f5T0eXsTPlq/V.DJUKmRHfYfBJ.o.lil1T35TFbjs2xiem','ROLE_USER'),
    ('Jimin',
'$2a$10$tmxuYYg1f5T0eXsTPlq/V.DJUKmRHfYfBJ.o.lil1T35TFbjs2xiem','ROLE_USER'),

```

```
('Noah',  
'$2a$10$tmxuYYg1f5T0eXsTPlq/V.DJUKmRHyFbJ.o.lil1T35TFbjs2xiem','ROLE_USER'),  
('user',  
'$2a$10$tmxuYYg1f5T0eXsTPlq/V.DJUKmRHyFbJ.o.lil1T35TFbjs2xiem','ROLE_USER'),  
  
('admin','$2a$10$tmxuYYg1f5T0eXsTPlq/V.DJUKmRHyFbJ.o.lil1T35TFbjs2xiem','ROLE_ADMIN');
```

```
INSERT INTO doctors (name, speciality, availability)  
VALUES  
('Dr. Tom Doe', 'Neonantal', 'Mon-Fri, 9am-5pm'),  
('Dr. Yacob Wafayee', 'Generalist', 'Mon-Fri, 10am-4pm');
```

```
INSERT INTO appointments (patient_id, doctor_id, appointment_time, status)  
VALUES  
(1, 1, '2024-07-30 10:00:00', 'booked'),  
(2, 2, '2024-07-31 11:00:00', 'booked'),  
(3, 1, '2024-07-31 11:00:00', 'booked');
```

```
COMMIT TRANSACTION;
```