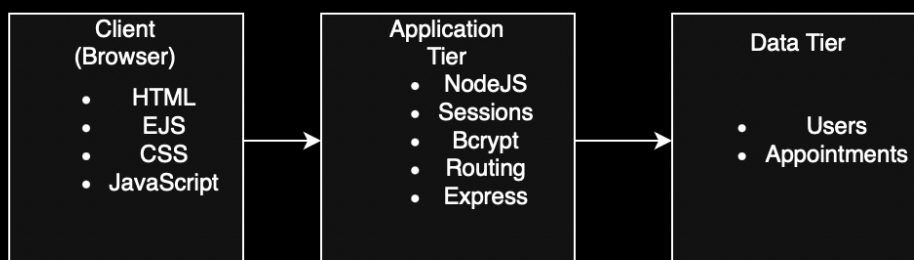# Hakan's Clinic

Outline

A web application that allows patients and doctors to book and manage appointments. My project has 3 types of user's admin (which manages everything such as profiles of other users and appointment details), doctor (who just manages appointments: cancel accept or complete) and patient (who can book an appointment with doctors and view their appointment status). There are also other functionalities like looking up doctors and registering.

Architecture

```
Project Structure

├── public/            # Static files (CSS, images)
├── routes/            # Express route handlers
│   ├── admin.js           # Admin routes
│   ├── appointments.js    # Appointment handling
│   ├── doctors.js         # Doctor routes
│   ├── main.js            # General routes
│   ├── patient.js         # Patient routes
│   └── users.js           # User & auth routes
├── views/             # EJS templates
│   ├── dashboard_*.ejs    # Role dashboards
│   ├── login.ejs          # Login page
│   ├── register.ejs       # Registration page
│   └── search.ejs         # Search interface
├── create_db.sql          # Database schema setup
├── insert_test_data.sql   # Sample data
├── index.js               # App entry point
└── .env                   # Env variables
```

As shown in my diagram, the system is divided into three tiers: client, application, and data. The application tier is built primarily with Node.js and Express.js, which manage routing and session handling. In addition, the bcrypt library is used to securely hash user passwords.

The data tier uses MySQL to store user information and appointment records. This tier directly supports frontend features such as bookings and user profiles. Communication between the application and data tiers occurs through SQL queries, ensuring that the database is accessed and updated whenever needed—for example, when a user creates a new appointment.

Data Model

id	first_name	last_name	email	phone_number	age	password_hash	role	status	created_at
1	System	Admin	admin@gold.ac.uk	+440000000000	20	$2a$10$3xXiFFRjy/DXYiLhS9RLB.FCDEaQaZb...	admin	active	2025-12-10 18:28:06
2	Gregory	House	greg@clinic.com	555-0101	52	$2a$10$3xXiFFRjy/DXYiLhS9RLB.FCDEaQaZb...	doctor	active	2025-12-10 18:28:06
4	Stephen	Strange	strange@clinic.com	555-0103	45	$2a$10$3xXiFFRjy/DXYiLhS9RLB.FCDEaQaZb...	doctor	active	2025-12-10 18:28:06
5	John	Doe	john@gmail.com	555-0201	25	$2a$10$3xXiFFRjy/DXYiLhS9RLB.FCDEaQaZb...	patient	active	2025-12-10 18:28:06
6	Alice	Smith	alice@gmail.com	555-0202	30	$2a$10$3xXiFFRjy/DXYiLhS9RLB.FCDEaQaZb...	patient	active	2025-12-10 18:28:06
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

id	patient_id	doctor_id	appointment_dateti...	reason	status	created_at
1	5	2	2023-11-15 09:00:00	Severe leg pain	completed	2025-12-10 18:28:06
4	5	2	2025-12-12 12:12:00		completed	2025-12-11 15:56:01
5	5	2	2025-12-19 12:12:00		pending	2025-12-11 15:57:11
6	5	2	2025-12-13 12:12:00	yes	completed	2025-12-11 15:59:00
7	5	4	2025-12-27 12:12:00	xcfgvhbjnkm	pending	2025-12-11 20:21:42
8	5	2	2025-12-19 12:12:00	dsfghjkl	pending	2025-12-11 20:24:02
NULL	NULL	NULL	NULL	NULL	NULL	NULL

My data model primarily consists of two key elements: **users** and **appointments**. The *users* table stores the information submitted through the registration page—for example, when users enter their username and password, these details are saved into the SQL database. The *appointments* table stores appointment-related data, including the date, time, reason for the appointment, and both the patient ID and doctor ID.

Functionality

The Hakan's Clinic web application provides a range of features tailored to three different user roles: **Admin**, **Doctor**, and **Patient**. Each role has access to specific capabilities that support clinic operations, streamline appointment scheduling, and maintain user data securely.

1. Authentication & User Management
- **Registration:** New users can create an account by providing a username, password, and role (patient or doctor). All passwords are hashed using **bcrypt** before being stored in the MySQL database.
- **Login:** Users authenticate using their registered credentials. Sessions are maintained via Express.js session management.

- **Role-Based Access Control (RBAC):** Once logged in, users are granted permissions based on their assigned role. This ensures that sensitive operations are restricted to authorised users.

2. Patient Functionality

Patients interact with the system primarily to book and manage their appointments. Key features include:
- **Browse Doctors:** Patients can view a list of available doctors, along with their specialisations (if applicable), to help choose an appropriate practitioner.
- **Book Appointment:** Patients can select a doctor, specify a date and time, and submit an appointment request. This information is saved in the appointments table.
- **View Appointment Status:** Patients can track whether their appointment has been *accepted*, *rejected*, *cancelled*, or *completed*.

Welcome, John

Logout

Book an Appointment

Fill out the form below to schedule a consultation.

Select Doctor:
-- Choose a Doctor --

Date: 12/12/2025   **Time:** 12:30

Reason / Notes:
Describe your symptoms...

Book Appointment

My Appointments

Date & Time	Doctor	Reason	Status
15/11/2023, 09:00:00	Dr. Gregory House	Severe leg pain	COMPLETED
01/12/2023, 10:00:00	Dr. Meredith Grey	Follow up checkup	COMPLETED
18/12/2025, 12:12:00	Dr. Gregory House	leg pain	COMPLETED
27/12/2025, 12:21:00	Dr. Gregory House	dsfghj	COMPLETED

3. Doctor Functionality

Doctors use the system to manage appointments they have been assigned. Their features include:
- **View Appointments:** Doctors can see all appointment requests from patients.
- **Update Appointment Status:** Doctors can:
  - **Accept** an appointment request
  - **Reject** a request (e.g., if unavailable)
  - **Mark as Completed** after the consultation
- **Manage Daily Schedule:** By updating appointment statuses, doctors maintain an organised and clear timetable for their consultations.

Welcome, Dr. Gregory

Logout

Filter by Date: 12/12/2025 Patient Name: Enter name...   Filter   Clear Filters

Upcoming Appointments

Date	Time	Patient	Reason	Status	Actions
No upcoming appointments found.					

Appointment History

Date	Time	Patient	Reason	Final Status
2025-12-27	12:21	John Doe	dsfghj	completed
2025-12-18	12:12	John Doe	leg pain	completed
2023-11-15	09:00	John Doe	Severe leg pain	completed

4. Admin Functionality

Admins have the highest level of control and can manage both users and appointments throughout the entire system.

- **Manage User Accounts:** Admins can view all registered users, update their details, or remove accounts if necessary.
- **Manage Appointments:** Admins may edit, delete, or review appointments for auditing and maintenance purposes.
- **System Oversight:** Admins ensure that all components (patients, doctors, appointments) operate smoothly and remain up to date.

Admin Control Panel

Logout

Filter Records: Search Name... [Search] [Reset]

Pending Doctor Requests

Name	Email	Actions
No pending requests.		

Manage Patients

First Name	Last Name	Email	Actions
Alice	Smith	alice@gmail.com	Update Delete
gold	smiths	gold@patient.com	Update Delete
John	Doe	john@gmail.com	Update Delete

Manage Doctors

First Name	Last Name	Email	Actions
gold	smiths	gold@doctor.com	Update Delete
Gregory	House	house@clinic.com	Update Delete
Meredith	Grey	grey@clinic.com	Update Delete

Manage Appointments

Date	Time	Patient	Doctor	Reason	Status	Action
27/12/2025	12:21	John Doe	Dr. Gregory House	dsfghj	Completed	Update
18/12/2025	12:12	John Doe	Dr. Gregory House	leg pain	Completed	Update
25/12/2024	14:30	Alice Smith	Dr. Meredith Grey	Consultation for	Pending	Update
01/12/2023	10:00	John Doe	Dr. Meredith Grey	Follow up check	Completed	Update
15/11/2023	09:00	John Doe	Dr. Gregory House	Severe leg pain	Completed	Update

5. Additional Features

- **Search Functionality:** Users can search for doctors based on name or other criteria implemented within the system.
- **Error Handling & Validation:** Forms throughout the system include validation to prevent incorrect or incomplete submissions. Errors such as incorrect login credentials or invalid booking times trigger user-friendly feedback.

Home | Register

Login

Login failed: Incorrect password.

Email:

Password:

Login