Final Project Report

Hospital Management System

AUTHOR

Name**:** Adesh Dey
Roll No: 23f1001104
Email**:** 23f1001104@ds.study.iitm.ac.in

DESCRIPTION

The goal of this project is to create a comprehensive hospital management system that will facilitate communication between physicians, patients, and hospital administrators. The system permits:

- Patients to register, book appointments with doctors, view their medical history, and manage their profiles
- Doctors to view their appointments, update patient records with diagnoses and prescriptions, and manage their availability schedules
- Administrators to oversee the entire system, manage doctors and patients, handle registrations, and monitor all appointments

With role-based access control guaranteeing the proper permissions for every user type, the application offers full CRUD (Create, Read, Update, Delete) functionality for users, appointments, and medical records.

TECHNOLOGIES USED

Here are the technologies I used in this project.

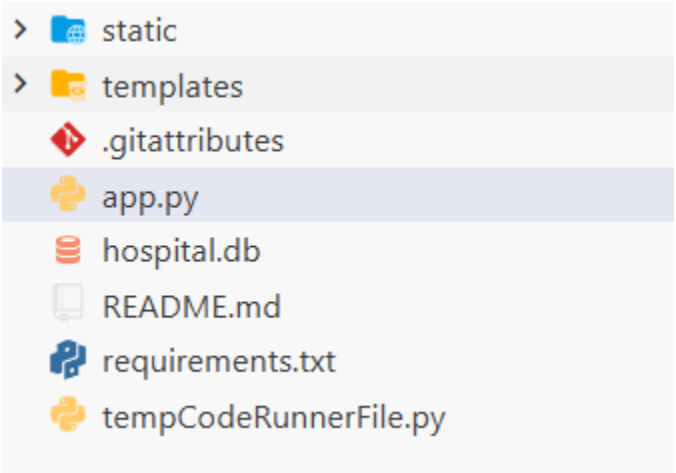
Python, HTML/CSS, Jinja, bootstrap, Flask, SQLite, Flask-SQLAlchemy,

- Python is the core programming language used; Flask is the main framework used for the WebApp.

- SQLite is used for creating a database file for data storage and access.

- Flask-SQLAlchemy is the SQL toolkit used to connect with the database file

ARCHITECTURE



app.py file contains the main code.
File names are pretty obvious for their roles.
“static” folder contains all the CSS files and “templates” folder contains all the HTML files.
“hospital.db” contains the database.

FEATURES

Core Features:

1. User Management

- Registration System - New patients can self-register
- Authentication - Secure login with role-based routing
- Profile Management - Users can update their information
- Status Control - Admin can blacklist/unblacklist users

2. Doctor Management (Admin)

- Add Doctors - Register new doctors with department and bio
- Edit Doctor Info - Update email, department, bio, and availability
- Remove Doctors - Delete doctor records with confirmation
- Availability Management - Set weekly schedules with time slots
- Department Assignment - Five departments: Ophthalmology, Gynecology, Neurology, Orthopedics, ENT

3. Appointment System

- Book Appointments - Patients select doctor, day, and time slot
- Slot Capacity Control - Maximum 8 bookings per slot
- View Appointments - Role-specific views of appointment lists
- Cancel Appointments - Patients can cancel their bookings
- Status Tracking - Booked, Completed, Cancelled statuses

4. Medical Records

- Patient History - Complete medical record tracking
- Doctor Updates - Add diagnosis, prescriptions, tests, and medicines
- Appointment Completion - Mark appointments as completed with details
- Historical View - Patients can view their entire medical history

5. Admin Dashboard

- Comprehensive Overview - View all patients, doctors, and appointments
- Search Functionality - Search across all entities by name, email, ID, or department
- User Management - Edit, remove, blacklist/unblacklist users
- Statistics Display - Total counts of doctors, patients, and appointments

VIDEO LINK

Video demonstration of the project is available here:
[mad 1 project video.mp4]

DB SCHEMA DESIGN:

I have created a total of 6 tables to make my work easy and more organized. I have tried to Normalize the schema.
ER diagram for the schema has been attached by me on the last page, kindly refer to that for more details

Tables Overview

Table Name	Purpose	Key Columns
users	Stores all user accounts	id, fullname, email, password, role, status
doctors	Extended doctor information	id, fullname, email, password, department, bio, status
doctors_availability	Doctor scheduling	id, doctor_id, doctor_name, day, morning, afternoon, evening, night
appointments	Booking records	id, doctor_id, patient_id, doctor_name, patient_name, department, day, slot, created_at, status
patient_history	Medical records	id, appointment_id, doctor_id, patient_id, doctor_response, test_done, medicines, diagnosis, prescription

Table : users

Column	Type	Constraints
id	INTEGER	PRIMARY KEY, AUTOINCREMENT
fullname	TEXT	NOT NULL
email	TEXT	UNIQUE, NOT NULL
password	TEXT	NOT NULL
role	TEXT	CHECK(role IN ('admin', 'doctor', 'patient')), NOT NULL
status	TEXT	CHECK(status IN ('active', 'blacklisted')), DEFAULT 'active'

Table : doctors

Column	Type	Constraints
id	INTEGER	PRIMARY KEY, FOREIGN KEY (users.id) ON DELETE CASCADE
fullname	TEXT	NOT NULL
email	TEXT	UNIQUE, NOT NULL
password	TEXT	NOT NULL
department	TEXT	CHECK(department IN ('Ophthalmology', 'Gynecology', 'Neurology','Orthopedics', 'ENT')), NOT NULL
bio	TEXT	-
status	TEXT	CHECK(status IN ('active', 'blacklisted')), DEFAULT 'active'

Table : doctors_availability

Column	Type	Constraints
id	INTEGER	PRIMARY KEY, AUTOINCREMENT
doctor_id	INTEGER	NOT NULL, FOREIGN KEY (users.id) ON DELETE CASCADE
doctor_name	TEXT	NOT NULL
day	TEXT	CHECK(day IN ('Monday','Tuesday',...)), NOT NULL
morning	INTEGER	CHECK(morning IN (0,1)), DEFAULT O
afternoon	INTEGER	CHECK(afternoon IN (0,1)), DEFAULT O
evening	INTEGER	CHECK(evening IN (0,1)), DEFAULT O
night	INTEGER	CHECK(night IN (0,1)), DEFAULT O
-	-	UNIQUE(doctor_id, day)

Table : appointments

Column	Type	Constraints
id	INTEGER	PRIMARY KEY, AUTOINCREMENT
doctor_id	INTEGER	NOT NULL, FOREIGN KEY (doctors.id)
patient_id	INTEGER	NOT NULL, FOREIGN KEY (users.id)
doctor_name	TEXT	NOT NULL
patient_name	TEXT	NOT NULL
department	TEXT	NOT NULL, CHECK(department IN (...))
day	TEXT	NOT NULL
slot	TEXT	NOT NULL
created_at	DATETIME	DEFAULT CURRENT_TIMESTAMP
status	TEXT	DEFAULT 'booked'

Table : patient_history

Column	Type	Constraints
id	INTEGER	PRIMARY KEY, AUTOINCREMENT
appointment_id	INTEGER	NOT NULL, FOREIGN KEY (appointments.id) ON DELETE CASCADE
doctor_id	INTEGER	NOT NULL, FOREIGN KEY (doctors.id) ON DELETE CASCADE
patient_id	INTEGER	NOT NULL, FOREIGN KEY (users.id) ON DELETE CASCADE
doctor_name	TEXT	NOT NULL
patient_name	TEXT	NOT NULL
department	TEXT	NOT NULL
doctor_response	TEXT	CHECK(doctor_response IN ('Cancelled', 'Completed', 'Yet to be', 'Cancelled By Patient')), NOT NULL
test_done	TEXT	-
medicines	TEXT	-
diagnosis	TEXT	-
prescription	TEXT	-
created_at	DATETIME	DEFAULT CURRENT_TIMESTAMP

ER DIAGRAM