

# Modern Application Development Project Report

## Hospital Management Application

- Student Details:

Name: Aryan Shukla

Roll number: 23f3000013

Email: [23f3000013@ds.study.iitm.ac.in](mailto:23f3000013@ds.study.iitm.ac.in)

About me:

I am a Diploma-level student at IIT Madras, pursuing BS in Data Science and Applications. This was my first project and it helped me understand routing, template rendering, database-driven workflows in Flask and more.

---

- Project Details:

Project title: Hospital management application

You are required to build a Hospital Management System application (HMS) web application that allows Admins, Doctors and Patients to interact with system based on their particular assigned roles. The project uses Flask, SQLite and Bootstrap.

---

- AI/LLM Declaration:

I, Aryan Shukla declare that this project was developed primarily through my own learning, coding and implementation efforts.

AI tools such as ChatGPT were only used for

- Debugging errors
- Concept Clarification
- Getting explanations of Jinja and SQLAlchemy Concepts

All core logic, routes, templates, database modelling, and implementation was completed independently by me.

---

- Technologies Used:

Backend:

-Python 3.x

-Flask

-Flask-SQLAlchemy

Frontend:

-HTML5

-Jinja2 Templates

-Bootstrap 5

Database:

SQLite (file-based DB)

---

- Database Schema Design:

User Table:

- id (PK)
- name
- email(unique)
- password\_hash
- role (admin/doctor/patient)
- age, gender, contact, address
- is\_active

Doctor Table:

- id (FK to User.id)
- specialization
- experience\_years

Appointment Table:

- id (PK)
- patient\_id (FK)
- doctor\_id (FK)
- date, time, status, reason

Treatment table:

- id (PK)
  - appointment\_id (FK)
  - diagnosis
-

- Video Link:

<https://drive.google.com/drive/folders/1Y7VSTnNh7NIXZzEDZRyuKdIYiy2w8aLS>

- Conclusion:

The Hospital-Management-app successfully demonstrates a role-based web system supporting patients, doctor, and administrators. It fulfils core requirements of appointment scheduling, record viewing, and structured interactions while maintaining a clean and simple UI.