

# Hospital Management System

## App Dev Project Report

**Student Name:** Hariharan R

**Roll Number:** DS23F2005421

**Email:** 23f2005421@ds.study.iitm.ac.in

This document provides an overview of the design, development, and implementation of the Hospital Management System created as part of the IIT Madras BS Degree App Development Project.

## 1. Student Details

**Name:** Hariharan R

**Roll Number:** DS23F2005421

**Email:** 23f2005421@ds.study.iitm.ac.in

**About Me:** I am a passionate developer focused on building IoT, AI, VR/AR, and full-stack applications with emphasis on intelligent systems and backend engineering.

## 2. Project Details

**Project Title:** Hospital Management System

### Problem Statement:

To create a fully functional hospital management system capable of handling administrative workflows, doctor scheduling, patient records, and appointment management using a secure, modular, and scalable architecture.

### Approach:

The application follows a modular Flask architecture separating concerns into admin, provider, and patient modules. SQLAlchemy handles ORM mapping, Jinja2 templates build the UI, and role-based access ensures proper segregation of functionalities. Unique ID generation improves tracking and consistency.

## 3. AI / LLM Declaration

I have used ChatGPT only for doubt clarification and documentation formatting. All coding and debugging were done manually.

## 4. Technologies and Frameworks Used

- Flask – Backend web framework
- SQLAlchemy – ORM for DB
- SQLite – Application database
- Jinja2 – Templating engine
- WTForms – Form validation
- Flask-Login – Session & authentication
- Bootstrap 5 – Frontend styling

## 5. Database Schema

The database includes models for Account, AccessLevel, Clinic, Provider, Recipient, Appointment Sessions, TimeSlots, and ClinicalNotes. The system follows one-to-many and many-to-one relationships across the modules.

## 6. API & Route Overview

Routes are grouped into modules:

- identity – Authentication
- governance – Admin operations
- clientele – Patient operations
- provision – Doctor operations

Each route implements a clean separation of responsibilities.

## 7. Architecture & Features

The application is structured into:

- models/ – SQLAlchemy ORM
- routes/ – Blueprints
- templates/ – HTML/Jinja2 UI
- static/ – CSS, JS assets

Core Features:

- ✓ Authentication & RBAC
- ✓ Admin Dashboard
- ✓ Doctor Availability System
- ✓ Patient Appointment Booking
- ✓ Unique ID Generation
- ✓ Clinic & Session Management

## 8. Video Presentation

Video Link:

[https://drive.google.com/file/d/1WEmXTuI0C1P1aohMBWWpl\\_K\\_LYyOZnSW/view?usp=sharing](https://drive.google.com/file/d/1WEmXTuI0C1P1aohMBWWpl_K_LYyOZnSW/view?usp=sharing)

## 9. GitHub Repository

[https://github.com/future-genius/hospital\\_management\\_system\\_app.git](https://github.com/future-genius/hospital_management_system_app.git)