

Title of the Project

Healthcare Management System

Submitted By

Abdul Awal Fahad

ID: 213-16-584

Department of Computing & Information System
Daffodil International University

Supervised By

Md. Nasimul Kader

Assistant Professor

Department of Computing & Information System
Daffodil International University

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computing and Information System (major:
Artificial Intelligence in IOT)



Department of Computing and Information System
Daffodil International University.
Dhaka, Bangladesh

Submission Date: 14/10/2025

APPROVAL

This Project titled “Healthcare Management System”, Submitted by **Abdul Awal Fahad**, ID No: **213-16-584** to the Department of Computing and Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on 14-10-2025.

BOARD OF EXAMINERS



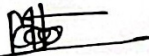
Md Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Chairman



Md. Nasimul Kader
Assistant Professor
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Md. Mehedi Hassan
Lecturer (Senior Scale)
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Ahmed Saif Reza
Managing Director & Chief Technology Officer
Medico Bio Limited

External Examiner

Declaration

I hereby declare that; this project has been done by me under supervision of **Md. Nasimul Kader (Assistant Professor)** department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By



Md. Nasimul Kader
Assistant Professor
Department of CIS
Daffodil International University

Submitted By



Name: Abdul Awal Fahad
ID: 213-16-584
Department of CIS
Daffodil International University

Acknowledgement

First and foremost, I would like to express my gratitude to Almighty Allah for providing this incredible opportunity to learn and grow. Without His guidance and support, I would not have been able to complete this project and all the associated tasks. I am immensely thankful to my department, supervisor and all the faculty for letting me work on this project as it has been a valuable learning experience for me. Throughout this development I have gained a wealth of knowledge and skills, especially in regards to new technologies and industry practices. I am deeply indebted to my project manager who served as my supervisor, **Md. Nasimul Kader** sir, and my academic faculty. Their unwavering support, guidance, and encouragement have been instrumental in my professional development. They have always been there to provide the right advice, help me make informed decisions, and inspire me to overcome challenges. I am sincerely grateful to them for their kindness, assistance and belief in my abilities. Their mentorship has not only helped me navigate through difficult tasks but has also taught me how to excel in demanding situations and handle large-scale projects in the future.

Abstract

Our proposed “**Healthcare Management System**” is an AI integrated Django-based web application project aimed to connect the patients and the best qualified doctors in an online platform so that both can save their time and money. Besides, its main objective is to reduce the sufferings of patient. Using this system, patient can easily find out the qualified doctors in short time, avoid long-time journey in traffic jam, reduce the healthcare cost and save time. Our system is a blessing for them especially who are incapable to move. Our AI integrated face detection login system will ease their suffering. Standing on a foundation of HTML, CSS, JavaScript, Bootstrap for mobile-friendly UI, Python Django for backend, PostgreSQL for database. The platform provides the patients registration, login approval, view options like see the upcoming appointments, recent appointments, updated reports from doctors, the detail information of doctor, quick action options like update profile, and finally the most important task book doctor. To join the platform, doctors have to be verified by admin panel after registration. After verification, it provides the doctors login approval, view options like see the upcoming appointments, pending appointments, the detail information of patient in dashboard, quick action options like edit profile, update report, manage their availability time, mark as completed or decline appointment. For administrators, it offers a set of powerful instruments for working with doctors, patient and promotions, as well as statistical specifications for the decision-making process. The project also focuses on factorability with features such as load balancing and efficient database queries for large traffic and volume data of doctors. Though, there has been a provision of security with compliance to standards for data protection. Some of the activities involved were designed to come up with the doctors included testing and documenting to ascertain whether the doctor was functional, reliable, and available for patient. The issue like performance, responsiveness, and the usability design was solved through the use of incremental problem-solving and teamwork. Through the use of these technologies and adopting the latest trends in designing, the platform seeks to help businesses grow while at the same time making searching for doctors easy and enjoyable for patients while equipping businesses with effective tools in their operation

Table of Contents

Approval	ii
Declaration	iii
Acknowledgement	iv
Abstract	v
Table of Contents	vi
List of Figures	viii
List of Tables	ix
1. Chapter 1: Introduction	1
1.1 Introduction	2
1.2 Purpose of Project	3
2. Chapter 2: Initial Study	4
2.1 Project Proposal	5
2.2 Project Scope	7
2.3 Background of the project	7
2.4 Objective	8
3. Chapter 3: Literature Review	9
3.1 Discussion on problem domain based on available solution	10
3.2 Discussion on problem solution based on available solution	10
3.3 Recommended Approach	11
4. Chapter 4: Methodology	13
4.1 What to use	14
4.1.1 Agile Model:	14
4.2 Why to use	15
4.3 Implementation Plan	16
5. Chapter 5: Project Plan	17
5.1 Work Break Down structure	18
5.2 Resource Allocation	19
5.3 Time Boxing	20
5.4 Gantt chart	21
6. Chapter 6: Feasibility Study	22
6.1 All possible types of feasibility study	23
6.2 Technical Feasibility	23
6.3 Economic Feasibility	23

6.4	Operational Feasibility	24
6.5	Legal & Compliance Feasibility	24
7.	Chapter 7: Foundation	25
7.1	The problem area identification	26
7.2	Interview:	26
7.3	Questionnaire:	26
7.4	Requirement Specification	27
8.	Chapter 8: Exploration	28
8.1	Activity diagram	29
8.2	Use case of Dashboard	30
9.	Chapter 9: Exploration	31
9.1	Module of the system	32
9.2	Class diagram of the system	34
9.3	High-fidelity prototype of the system	35
10.	Chapter 10: Development	39
10.1	Folder structure of the system	40
10.2	Core module output sample:	41
10.3	Prioritization while developing	64
11.	Chapter 11: Testing	65
11.1	Test Case	66
12.	Chapter 12: Implementation	69
12.1	Scaling	70
12.2	Load Balancing	70
13.	Chapter 13: Critical Appraisal and Evaluation	72
13.1	Objective that could be met:	73
13.2	How much better could have been done:	73
13.3	Which features could not be touched	74
14.	Chapter 14: Lesson Learned	75
14.1	Pre Project – Review – Closing	76
14.2	The Problem I Have Faced	77
14.3	What Solutions Occurred:	78
15.	Chapter 15: Lesson Learned	79
15.1	Summary of the project	80
15.2	Goal of the project	80
15.3	What I have done in Documentation	81
15.4	My Experience:	81

List of Figures:

Figure 4-1: Agile Model	15
Figure 5-1: Work Break down Structure	18
Figure 5-2: Resource Allocation	19
Figure 5-3: Time Boxing	20
Figure 5-4: Gantt chart	21
Figure 8-1: Activity Diagram (order process)	29
Figure 8-2: Use case of Dashboard	30
Figure 9-7: Low Fidelity prototype (Dashboard page)	33
Figure 9-8: High Fidelity prototype (Home page)	33
Figure 9-9: High fidelity prototype (customize page)	34
Figure 9-10: High fidelity prototype (Order page)	34
Figure 9-11: High fidelity prototype (Login page)	35
Figure 9-12: High fidelity prototype (dashboard page)	35
Figure 10-1: Root folder	37
Figure 10-2: Client-side Folder structure	37
Figure 10-3: Server-side folder structure	38
Figure 10-4: Actual folder Structure Sample	38
Figure 10-5: Output of Color-Picker Module	39
Figure 10-6: Output of Home page	39
Figure 10-7: Output of File-picker module (1)	40
Figure 10-8: Output of File-picker module (2)	40
Figure 10-9: Output of Ai-picker module (1)	41
Figure 10-10: Output of Ai-picker Module (2)	41
Figure 10-11: Output of Order processing Module	42
Figure 10-12: Output of Admin login Module	42
Figure 10-13: Output of Dashboard Module	43

List of Tables:

Table 8.1.3-1: Table of Module system	32
Table 10-8.1.3-1: Table of Module system	43
Table 11.1: Test Case	45

Chapter 1

Introduction

1.1 Introduction

Today, with the rapid growth of digital health services, connecting patients and healthcare providers through online platforms has become increasingly common. **Healthcare Management System** is a comprehensive online platform designed to efficiently connect patients, families, and healthcare institutions with best qualified doctors. The application has been designed and developed considering both user-friendliness and administrative efficiency in mind, ensures smooth interaction between doctors, patients, and system administrators. The platform is using HTML, CSS to build dynamic and interactive frontend, while Python is used for In combination with the Django framework, provides a solid and scalable backend infrastructure. PostgreSQL serves as a secure and fast database solution, guaranteeing that user information is safely stored. profiles, service requests, and booking details. For styling, Tailwind CSS and Daikyu create a clean, responsive, modern interface, while Vs Code is used as the build tool to enhance development speed and maintain a well-structured project architecture. In addition, on the administrative side, Hospital Management provides a strong admin dashboard that includes real-time statistics, charts, and tools for managing doctors, patients, bookings, payments, and service packages. Administrators can confirm doctor credentials, update user roles, approve or reject doctor applications, and manage service categories or promotional offers to enhance platform visibility. Custom admin authentication ensures that sensitive administrative features are securely accessed by authorized personnel only. From the user's standpoint, Doctor Finder is all about reliability, accessibility, and trust. Patients Or, families are easily registering, identifying themselves, and making a detailed profile. The platform allows users to browse through categories such as home care, elderly support, emergency nursing, or post- care in surgery. View doctors' profiles, check for qualifications, availability, then book services. securely. Among its special features are real-time notifications, secure payment options, 24/7 support, and a Clear service policy will enhance user confidence and improve platform credibility. With its well-rounded features, simple but powerful interface, and advanced administrative tools, Hospital management is the most effective solution in developing modern health services. platforms make it easier to find and recruit professional doctors online.

1.2 Purpose of Project

The aim of the Healthcare Management System project is the development and implementation of a new, effective IT Eco-system for patients, relatives, and booking appointments in real time. This portal promises to address to the demands of clients as well as administrators with its straightforward, user-friendly architecture. This paves the way for secure communication, seamless hiring patterns, and incredible doctors. management utilities. What we want to offer to patients and their families is an accessible, reliable experience of health. care. It has its features like quick registration, profile verification, Automated diagnosis, Automated face detection login and place request facility along with secure payment. Real-time units will be provided to the user in short notice time. Extra services include 24/7 support, Extensive doctor profiles with qualifications and flexible booking options improve user satisfaction. and peace of mind. For the administrators and management staff members, it brings features like: doctor profile. management, job allocation & tracking and performance analytics that make healthcare operation smoother. Admin Dashboard provides the ability to view key metrics, including doctor availability, service requests and payment. Admin can handle the payment transaction for patients. Tools for doctor verification, Service monitoring and report generation help to ensure quality service and the respective platform. integrity. The system also addresses the challenge of scalability and data security, ensuring smooth functionality. performance even with increased users and requests. Built using modern technologies such as HTML, CSS, Python Django, PostgreSQL, and Bootstrap, the platform ensures high Performance, intuitive interaction, and flexibility for future feature expansion. In the end, Health care management is a broad-based digital solution for connecting health seekers with professional doctors, ensuring convenience, trust, and efficiency in the process. healthcare service marketplace.

Chapter 2

Initial Study

2.1 Purpose of Project

The primary motive of the **Health care management system** is to establish a new age efficient and satisfactory AI integrated platform that connects the doctors and the patients in the online platform. This platform is intended to serve the needs of both patients and administrators through its practical design carrying out secure transactions and easy management tools.

1. Project Overview

This is a comprehensive Healthcare Service Platform to facilitate quick and easy doctor hiring & managing, for both patients and management. It leverages new age technologies to bring a dependable solution for healthcare organizations/homes to locate and hire quality doctors while making sure that the service is responsive, reliable, secure and feature-packed. Aims The objective is to develop a profile for the patient and for serving issues in virtual environments, analyzing the long-term demand of serving care via digital means that does not reach practice, but continues instead on written waiting lists.

2. Objectives

Deliver the best experience when a patient is booking doctors with automatic diagnosis and automatic face detection login. Provide administrators with good solutions for organizing doctors, patients, and promotions. Make it cost effective and time saving for patient. Especially reduce the suffering of patients who are incapable to move or more than serious conditions. Need for a platform that will be able to accommodate a growing traffic and amounts of data as it will play a key role in the organization's performance. Some of these include, creating patients accounts, verification of doctors account, protection of verification information details, and availability of patient care services, all aimed at empowering trust and ensuring people stick with you.

3. Key Features

Admin Features: Admin Dashboard for tracking of real-time statistic data, Tools used to manage and verify doctors, patients, Specific and authorized access solutions for administration, Handle successful payment transaction.

Patient Features: It is easy to sign up and verification is done through admin approval. It adds the home page with finding doctors, booking appointment and viewing details of doctors.

Doctor Features:

Qualified doctors are available for patient cares at near to them. Patient support available all day and night so that patients are well catered for at all times.

4. Technologies Used

Frontend: HTML, CSS, Bootstrap, JavaScript.

Backend: Python Django

Database: PostgreSQL

Build Tool: Vs code.

5. Benefits

For patients: Simplicity in engaging skilled doctors, now with the guarantee of safe transaction payments, verified doctor profiles, and trusted healthcare support any time and place.

For admins: streamline operations with centralized management tools and real-time analytics on doctor activity, service requests, and performance data.

For care providers/ agencies: Provides for broader reach to increase patient or service request volume, faster response time and scalable digital model that can accommodate change and growth in health.

2.2 Project Scope

The main project for **Health care management system** is the creation of a functional and secure e-platform conception for the patients and administration. This entails having an effective frontend to facilitate easy navigation, and actual booking process, as well as a robust backend to safely store all relevant information and securely carry out all essential transactions. The features to be included within the scope are basic, yet contain registration, personal account, service catalog, safe methods of payment option, and support. It offers the administrative control panel for doctors, users and promotions, real-time data and statistical data visualization. Technologically developed through HTML, CSS, Python Django, Bootstrap and PostgreSQL, the intended platform architecture for the transactions and the patients is comprehensive, secure, and malleable for any further enhancements that may be needed to provide long-term value.

2.3 Background of the project

This change of growth in e-platform has impacted the whole healthcare system as most organizations use the internet to expand their service. It means that the modern patient demands the best qualified doctor's appointment at the shortest time, safely booking, payment etc. However, the development, management and scaling of such practices present many problems for the small and medium-sized clinics and hospitals. Understanding this necessity, Hospital Management was designed to be a fully functional online booking best doctors that eliminating these problems and offering an easily navigable environment for patients.

It therefore models the project with e-healthcare giants attempting to import most of the present technological and design aspects into the platform. Built with the help of HTML, CSS, Python Django, Bootstrap and PostgreSQL, the platform guarantees fast and secure work with a high degree of scalability. It also focuses on such aspects as personal user profiles, safety of payments, and advertising campaigns in order to increase the share of patients' satisfaction. For administrators, such options as the integrated management tools delivering superior performance information in real time will help make work easier and allow healthcare system to concentrate in the areas of growth and patient satisfaction. It is therefore the goal of Hospital Management organization to ensure that any technological development will meet the applicable practical needs of the digital healthcare system.

2.4 Objective

Using appropriate internet technologies will make the platform convenient for users, making the healthcare experience smooth and accessible while giving administrators the essential tools to manage doctors, patients, and service requests effectively.

The emphasis is on integrating the functionality, scalability, and security to build a solid foundation for healthcare services. Management. For patients, the goal is to provide a one-stop solution where viewing and booking the best doctor is concerned. Simple, fast, and safe.

The use of AI allows key features such as easy registration and automated login. Face detection-based login, personalized patient profiles, automated diagnosis, secure online book and pay for your options, and real-time online support to ensure complete confidence in satisfaction. By adding such dynamic features as Top-rated Doctors, Urgent Care Requests, and Personalized Recommendations keep the users informed about available doctors and enhance the overall experience. From the business perspective, the project is intended to provide administrators with effective tools by which to manage their operations.

The Admin Dashboard offers clear analytics, charts, and insights. Bookings, doctor performance, and user engagement metrics are some key aspects. A range of management tools for doctors, patients, and service records is incorporated to help the organization adapt to evolving healthcare needs and maintain service quality. Thus, aiming to balance patient convenience and business efficiency, the **Health care management system** is designed to be a reliable, secure, and user-friendly platform for the growing digital healthcare ecosystem.

Chapter 3

Literature Review

3.1 Discussion on problem domain based on available solution

A major issue facing the entire world today is the lack of qualified doctors and other healthcare workers. Finding trustworthy nursing support when needed can be difficult for patients, families, and healthcare facilities. Solutions have historically been dispersed, depending on third-party organizations, hospital contacts, or word-of-mouth recommendations. This can be expensive, time-consuming, and ineffective. There are currently a number of web-based and mobile-based platforms available to address this problem. Many of these solutions offer marketplaces for general caregivers or job boards specifically for doctors. Although these platforms link doctors with patients or healthcare providers, they frequently lack specialized features like automated diagnosis, automated face detection login, emergency booking, verified doctor credentials, real-time availability tracking, and secure client-doctor communication. Furthermore, current solutions are often regionally restricted, fail to sufficiently validate professional credentials, or prioritize long-term staffing over on-demand services. Users thus struggle to guarantee trust, openness, and prompt access to qualified doctors. Consequently, a specialized **Health care management system** can enhance current options by providing:

A centralized AI integrated platform that is easy to use, validated doctor profiles that include credentials and experience options for scheduled and on-demand reservations.

While guaranteeing effectiveness, dependability, and accessibility for doctors and patients alike, Such a system directly fills in the gaps in existing solutions.

3.2 Discussion on problem solution based on available solution

When it comes to hiring nursing services, the **Health care management system** offers a comprehensive solution to the problems that patients and healthcare provider's encounter. Current software solutions in this field frequently offer few search options, do not adequately verify doctor credentials, and do not guarantee that medical professionals are available in real time. Furthermore, the majority of systems in use today rely on manual procedures or simple listings, which can lead to errors, delays, and a decline in user trust. The Hospital Management Web Application uses cutting-edge computer technologies and prioritizes a seamless healthcare service experience to address these issues. A smooth user interface and responsive design across devices is ensured, for example, by using HTML, CSS for the front end, and feature-rich Bootstrap. This eliminates the

subpar optimization and non-responsiveness that many current health platforms have. This ensures accessibility to both the patients and the doctors, regardless of the device type.

The Admin Dashboard provides an effective yet simple way for administrators to track patient volume in real time. requests, booking status, and doctor availability and specialty. The most common issue with relying on ineffective manual monitoring systems, which often lead to errors and mismanagement, is Directly addressed by this.

Moreover, the backend system guarantees scalability and handling of data in an asynchronous manner, which This is driven by PostgreSQL and Python Django.

This prevents performance degradation even under high handles loads and offers a reliable solution for growing numbers of users and requests for services. For this reason, to maintain confidentiality and trust, it also allows for the safe storage of doctor credentials, patient histories, and booking information.

Many of the shortcomings of current healthcare hiring systems are removed by the Hospital Management. Application's feature-based approach includes verified doctor profiles, location-based Search, real-time availability, and secure communication. For patients seeking high-quality care and for the doctor seeking dependable opportunities, this makes it more efficient, safe, and user-friendly.

3.3 Recommended Approach

The most suitable methodology for developing this platform is Agile, since it allows continuous improvements and flexibility during the development cycle. The project setup should start with the basic functionalities: user registration, user manual or automated face detection login, creation of a doctor's profile, management of patient bookings, and secure payment. processing. When the main features have been implemented, the advanced functionalities of the Admin Integration with the dashboard, real-time tracking of doctor availability, and analytics can be done gradually.

By focusing on patient convenience, trust, and transaction security from the very start, the platform can provide a reliable healthcare service journey. In terms of scalability, picking JavaScript on the frontend guarantees quick response times and smooth interactions while the number of patients and doctors increases. PostgreSQL Integration allows for scalable, flexible storage of data; can handle increase in user records and bookings. histories, and medical credentials. For styling, using HTML, CSS, or Bootstrap allows for a clean, responsive, and professional look across devices

without compromising performance. The main development effort should be focused on the administration interface, which will act as Central hub for managing doctors' profiles, patient requests, feedback, and performances. analytics. This interface, supported by real-time

Finally, continuous reviews and feedback from both patients and doctors during the development stages will help identify challenges early and allow timely adjustments. This iterative process ensures that the Hospital Management evolves into a **time-efficient, cost-effective, and user-friendly solution** for patients in need of medical support including incapacitated or paralyzed individuals while also providing doctors with a trustworthy platform to connect with those who require their services.

Chapter 4

Methodology

When it comes to Hospital Management, the approach used is the agile development based on the development of a set of core technologies with periodic enhancements. Python Django, JavaScript, PostgreSQL, HTML and CSS makes the design effective, scalable, responsive and enable an optimal user and admin management.

4.1 What to use

The Project, Hospital Management has opted for HTML, CSS, Bootstrap, JavaScript for the front end, and Python Django for the back end, an efficient database for data storage is also considered. Styling is done by CSS and Bootstrap for mobile-friendly UI, web building tool Visual Studio code. Others are the verification; high-level admin panels; data visualization libraries for the analytics sections.

4.1.1 Agile Model

Agile model is another style of software development that is iterative and collaborative in nature. It is centered on releasing usable software every sprint, and the flexibility and user involvement. Daily scrum meetings, sprint reviews and other gatherings that occur allow for the free exchange of information and permit fast decision making. This brings the ability to change or to make iterations to be added at any given number of phases in development based on changing requirements or user demands.



Figure 4-1: Agile Model

4.2 Why to use

The technologies in Hospital Management select for offering a modern, scalable and efficient healthcare solution.

Bootstrap makes the UI highly responsive to the user and has reusable components for the enhanced development of processes.

Django offer a framework that is flexible and compatible with the extensive traffic and real-time operations.

PostgreSQL has adaptable features when it comes to handling and storing massive data, popular during changes in inventories.

CSS and Bootstrap offer a quick way to implement attractive aesthetics that adhere to the latest design conventions together with a responsive layout framework.

It also provides that the development speed will be fast with the help of fast build and hot module replacement.

4.3 Implementation Plan

Here's Phase 1: I chose the topic Planning and Design from the weekly options available on the course, so, Week 1 and Week 2 revolve around this theme.

Collect and write information about project needs.

Design mockups for the user interface and the back end and admin interface.

The table structure for product, user, order and offer should be defined and finalized.

Phase 2: Frontend development plan (Week 3-6)

We decided to configure the basic HTML, CSS, JavaScript environment using the Vs code tool for faster development. Utilize the user interface with Bootstrap for its responsive functionality.

Develop key pages: Home, Find Doctor, Doctor Profile, Patient Profile, and Booking Appointment and Messaging Section, Admin Panel.

From this lesson, I implemented JavaScript Ajax to handle navigation between different pages. Phase 3: Backend Development (Week 7-10).

Install the Python server and the Django framework. Include modules of authentication, products, orders, and offers.

Integrate it with the PostgreSQL for storage of data back end to the PostgreSQL database.

Ensure to validate admin authentication by use of custom hooks.

Phase 4: Integration and Testing phase outlines in this section of the paper includes weeks 11 to 13.

Make frontend integrate with backend APIs for it to work as intended.

Test user features: web services like registration, user login, updating user profile, searching doctor and booking appointment.

Test admin features: for doctor management, user management, specialist management, languages. Perform performance, security and cross-browser test.

Phase 5: Deployment and maintenance are the two key strategies that are scheduled for the sixteenth week of the plan.

host the platform on a cloud hosting service

Such performance will allow detecting and solving the problems related to post-deployment, which is critical for any IT project.

Of course, you keep on collecting feedback from the user for future improvement of your application.

Chapter 5

Project Plan

5.1 Work Break Down structure

The work breakdown structure for Hospital Management is to segregate the project into five major phases. First, the Planning and Design phase is the process of requirement gathering, joining a number of wireframes, and establishing the final database schema. Secondly, the Frontend Development phase concerns layout and interface creation based on the user interface design utilizing HTML, CSS, and Bootstrap; navigation structure. The Backend Development can be divided into such steps as the setup of and Python Django, JavaScript setting, API implementation and usage of PostgreSQL for data management. Thus, there is Integration and Testing to guarantee that all the parts work together, Deployment and, at last, Maintenance for post-deployment enhancements. That is why our project has been divided into few subtasks and we have worked out a strict schedule of which part of the project will be given how much time. It enables us to see the entire project at the same timetable.

Task name	Duration (Days)	Start	End
Introduction	8	14 May-2025	21 May-2025
Initial study	5	22 May-2025	26 May-2025
Literature Review	3	27 May-2025	29 May-2025
Methodology	5	30 May-2025	03 Jun-2025
Project Plan	8	04 Jun-2025	11 Jun-2025
Feasibility Study	7	12 Jun-2025	18 Jun-2025
Foundation	7	19 Jun-2025	25 Jun-2025
Exploration	9	26 Jun-2025	04 Jul-2025
Engineering	12	05 Jul-2025	16 Jul-2025
UI/UX Development	5	17 Jul-2025	21 Jul-2025
Development	37	22 Jul-2025	27 Aug-2025
Testing	7	28 Aug-2025	03 Sep-2025
Implementation	5	04 Sep-2025	07 Sep-2025
Critical Appraisal	2	08 Sep-2025	09 Sep-2025
Lesson Learned	4	10 Sep-2025	13 Sep-2025
Conclusion	4	14 Sep-2025	17 Sep-2025
	Total: 130 days		

Figure 5-1: Work Break down Structure

5.2 Resource Allocation

Resource management is defined as the distribution and application of resources for executing project activities and achieving the defined goals of the project.

Task name	Duration (Days)	Resource Allocation
Introduction	8	Analyst, User
Initial study	5	Analyst
Literature Review	3	Analyst, Team Leader
Methodology	5	Analyst, Developer, Project Manager
Project Plan	8	Analyst, Project Manager, Team Leader
Feasibility Study	7	Analyst, Project Manager, Team Leader, User
Foundation	7	Analyst, Team Leader
Exploration	9	Analyst, Developer, Team Leader, Designer
Engineering	12	Project Manager, Team Leader
UI/UX Development	5	Designer, Developer, Team Leader, User
Development	37	Developer, Analyst, Tester
Testing	7	User, Developer, Tester, Team Leader
Implementation	5	Project Manager, Developer, Tester
Critical Appraisal	2	Analyst, User, Developer
Lesson Learned	4	Developer, Analyst
Conclusion	4	Analyst
	Total: 130 days	

Figure 5-2: Resource Allocation

5.3 Time Boxing

Time boxing can be described as project management technique or approach that presupposes that time is divided into certain fixed time segments called time boxes. It assists in the development of a conscientious work attitude, work culture and time - sensitive working norms within specified working hours. We have certain time boxing which is mentioned in the below part of this project.

Time Boxes	Task name	Duration (Days)	Resource Allocation
TB1	Introduction	8	Analyst, User
	Initial study	5	Analyst
TB2	Literature Review	3	Analyst, Team Leader
	Methodology	5	Analyst, Developer, Project Manager
TB3	Project Plan	8	Analyst, Project Manager, Team Leader
	Feasibility Study	7	Analyst, Project Manager, Team Leader, User
	Foundation	7	Analyst, Team Leader
TB4	Exploration	9	Analyst, Developer, Team Leader, Designer
	Engineering	12	Project Manager, Team Leader
	UI/UX Development	5	Designer, Developer, Team Leader, User
TB5	Development	37	Developer, Analyst, Tester
TB6	Testing	7	User, Developer, Tester, Team Leader
	Implementation	5	Project Manager, Developer, Tester
TB7	Critical Appraisal	2	Analyst, User, Developer
	Lesson Learned	4	Developer, Analyst
	Conclusion	4	Analyst
		Total: 130 days	

Figure 5-3: Time Boxing

5.4 Gantt chart

A Gantt chart is a graphical project management tool which displays the schedule of your tasks or activities and their dependencies, over time. It lists when tasks begin and end, as well as their dependencies and milestones.

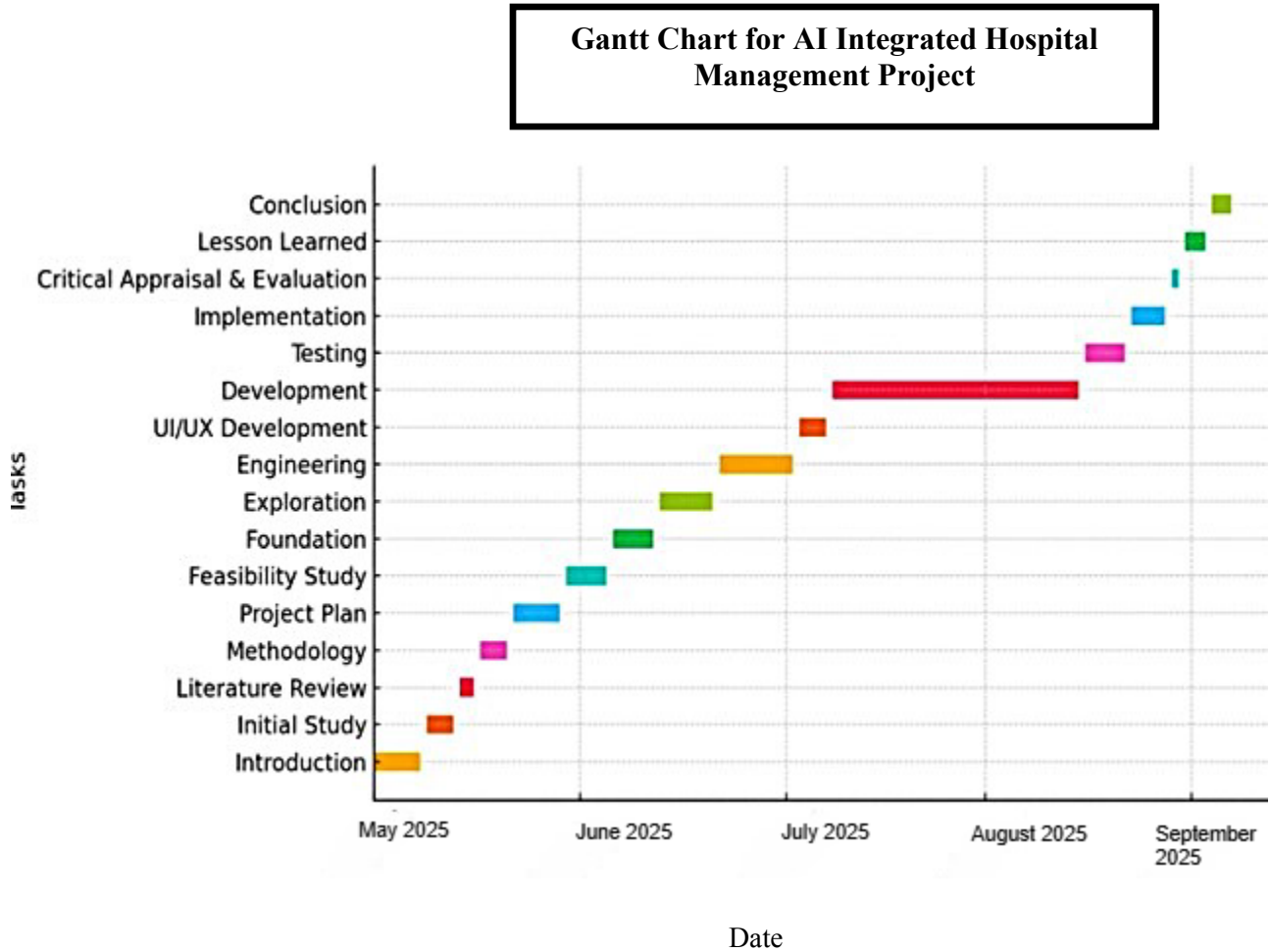


Figure 5-4: Gantt chart

Chapter 6

Feasibility Study

6.1 All possible types of feasibility study:

The common types of feasibility studies are technical feasibility that is checking on the available technology and resources, economic feasibility that focuses on the cost/benefit analysis and return on investment, operational feasibility that checks whether the project meets the users' requirement, legal feasibility that checks whether the project complies with the law and lastly schedule feasibility that looks at the viability of completing the project within the set timeframe.

6.2 Technical Feasibility:

The technical capabilities and existing resources, which will be need to deliver the project. The JavaScript, Python Django, PostgreSQL and Bootstrap, CSS technologies provide scalability, efficient performance, and up-to-date approach in the Hospital Management plan.

6.3 Economic Feasibility:

This study examines of Hospital Management platform and determines whether the project can yield repeatable returns to make the investments viable. The initial development cost comprises of technology (platform and languages used: HTML, CSS, Bootstrap, JavaScript, Python Django, and PostgreSQL), physical infrastructure (servers and hosting service), and human resource (developers, testers, designers). The main cost operation costs are related to server maintenance, advertising, and consumers' service. This is because the platform's revenue generating model will rely on sales, promotions, and subscriptions for the premium account. A breakeven calculation and ROI forecast suggests that further growth into functional domains backed by the systematic implementation of advertising, acquisition, and other promotional techniques presents appealing and sustainable acquisitive and profitable opportunities in the long term.

6.4 Operational Feasibility:

There are three key subtypes in analyzing Operational Feasibility for the Hospital Management platform, to do with the question of whether the system can be effectively kept running and managed in the long run. As for the usability, the platform is non-complex from patient's perspective and does not require complex backend administration. Some of these are staff training and documentation to facilitate easy operation of the organization. Therefore, the availability of large cloud-based structure facilitates management of extra traffic and booking is facilitated.

6.5 Legal & Compliance Feasibility:

Legal & Compliance Feasibility for the Hospital Management platform checks the compatibility of each proposed system in meeting legal requirements. This coupled with other Data protection laws that tackle issues like GDPR (General Data Protection Regulation) when it comes to handling patients' data. Also, the platform has to adhere to rules of intellectual property for the patients and doctors' information and content as well. Legal compliance is imperative in as it reduces risks related to the firm as well as increase the confidence of the users and stakeholders in the company.

Chapter 7

Foundation

7.1 The problem area identification:

Hospital Management is that existing best doctor's appointment solutions often offer a subpar user experience, are inefficient for doctor appointment in short time and payment, appointment management, booking are not always safe. The adopted approach of slow performance results in challenging usability for patients, and administrators are often limited when trying to manage patients, doctors, booking and payment transactions and promotions effectively. Furthermore, few platforms have individualization and compatibility, making patient interactions less enjoyable and organizational processes less effective. In order to address these challenges this project sets for itself a goal of implementing an easily searchable, expandable and secure healthcare solution appropriate for patients, doctors and administrative use.

7.2 Interview:

Surveys include getting information from patients, doctors, entrepreneurs and managers, which would be interested to use Hospital Management platform. They revolve around choices, drawbacks with existing Hospital Management models, required features, and functional issue. This feedback also helps to define or clarify requirements, and keep a project on track.

7.3 Questionnaire:

An assessment form for the project includes all necessary information about patients, doctors and administrators to improve the features of the Hospital Management site. From the patients, it is important to understand their choice in best doctor categories, mode of booking, payment and the support services they expect. For administrators, questions are concerned with the difficulties in handling booking for doctor in short time, patients, booking and needed analytics. The questionnaire also aims at getting an opinion on issues to do with security, mobile compatibility, and convenience. It also helps in the identification of the platform that meets the stakeholders' needs and offer the best and efficient solution.

7.4 Requirement Specification:

The Requirement Specification for Hospital Management identifies the features that are functional and the un-functional of the intended platform. Functional requirements are the registration of patients, doctors, verification of qualified doctors, management of profiles, view doctor in short time, secure login with face detection in real time , payment via mobile wallet like Bikash or in cash and booking appointment with automatic diagnosis system. Non-functional requirements include scalability, performance, security of data and the ability to provide with work on a mobile interface. The platform has to cater for the ease of making purchase, the organization's backend processes as well as analytics in real-time. The return to modern solutions and healthcare goals is guaranteed by such technologies as JavaScript, Python Django, PostgreSQL and Bootstrap, CSS.

Chapter 8

Exploration

8.1 Activity diagram:

Activity diagrams are most useful for modeling the behavior of a system or a process, documenting the activities that are necessary and identifying the utilize pale or opportunities for improvement. They offer an illustration of complex methodologies, and the relative positioning of different activities can be easily ascertained by the different stakeholders.

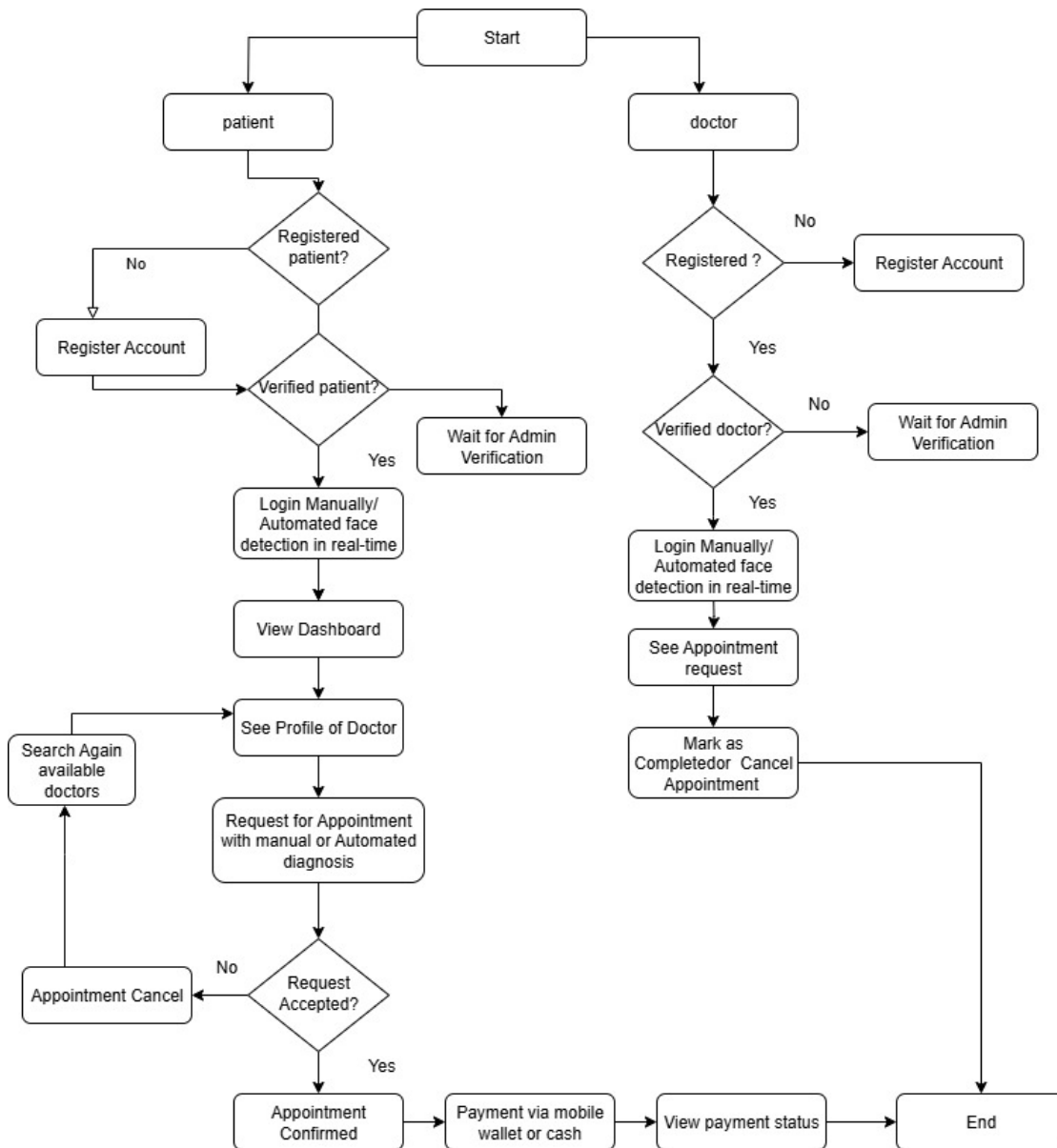


Figure 8-1: Activity Diagram

8.2 Use case of Dashboard:

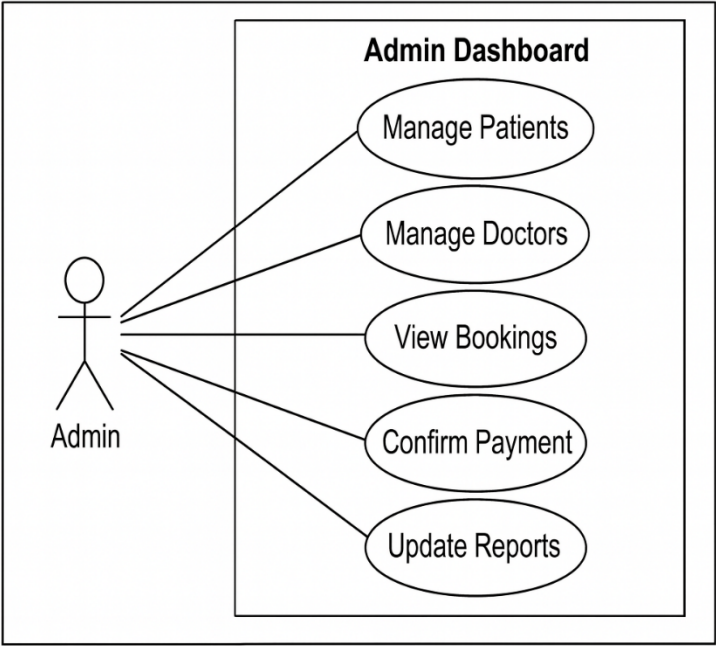


Figure 8-2: Use case of Dashboard

8.3 Process of AI- integrated Face Detection Model:

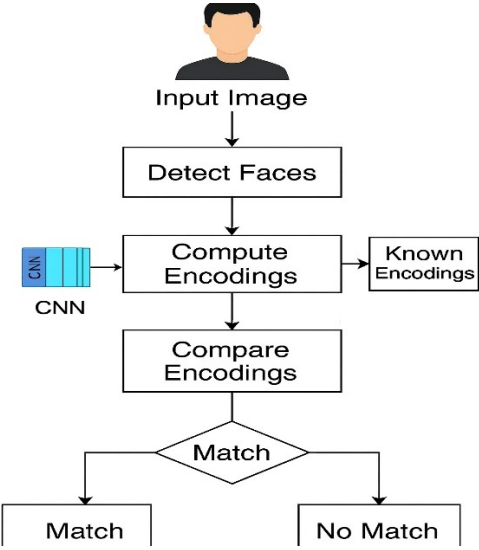


Figure 8-3: Face Matching process of AI Model

Chapter 9

Exploration

9.1 Module of the system:

Serial of Action	User Action	System Interaction
1	Patient opens the platform	System loads the homepage with searching option in google map near to patients.
2	Patient register / login manual or automated	System validates credential or real-time face detection and grants access.
4	Patient requests for booking	The system records the booking request and notifies the selected doctor in real-time through the doctor's dashboard.
5	Patient proceeds to payments	The system generates a secure payment page showing booking details, total cost, and available payment methods.
6	Patient completes payments	The system verifies the payment, updates the transaction status, and confirms booking upon success.
7	Patient views transaction status	The system retrieves payment data from the database and displays real-time transaction history with status (successful, pending, or failed).
8	Patient views booking status	The system fetches the booking details and displays the doctor's confirmation, appointment time, and booking progress.
11	Doctor register / login manual or automated	The system validates the doctor's credentials or authenticates the account using real-time face detection, and loads their dashboard showing appointments and availability.
12	Doctor views the appointments	The system retrieves and displays all current, pending, and past bookings for that doctor.
13	Doctor mark as confirm / cancel appointments	The system updates the booking status (confirmed/cancelled) and notifies the patient of the doctor's response.
15	Admin manages patients and doctors	The system loads the admin dashboard showing lists of users, doctors, and related data for monitoring and management.
16	Admin reviews patient details	System fetches and displays patients' data for management
17	Admin confirm the payment	The system updates the transaction record as "verified" and allows further processing of the booking.
18	Admin confirm the booking	The system updates the booking record as "approved" and notifies both the doctor and patient.

19	Doctor updates the records	The system applies the new records or updates to the database and refreshes the diagnosis options available to patients .
----	----------------------------	---

Table 8.1.3-1: Table of Module system

9.2 Class diagram of the system:

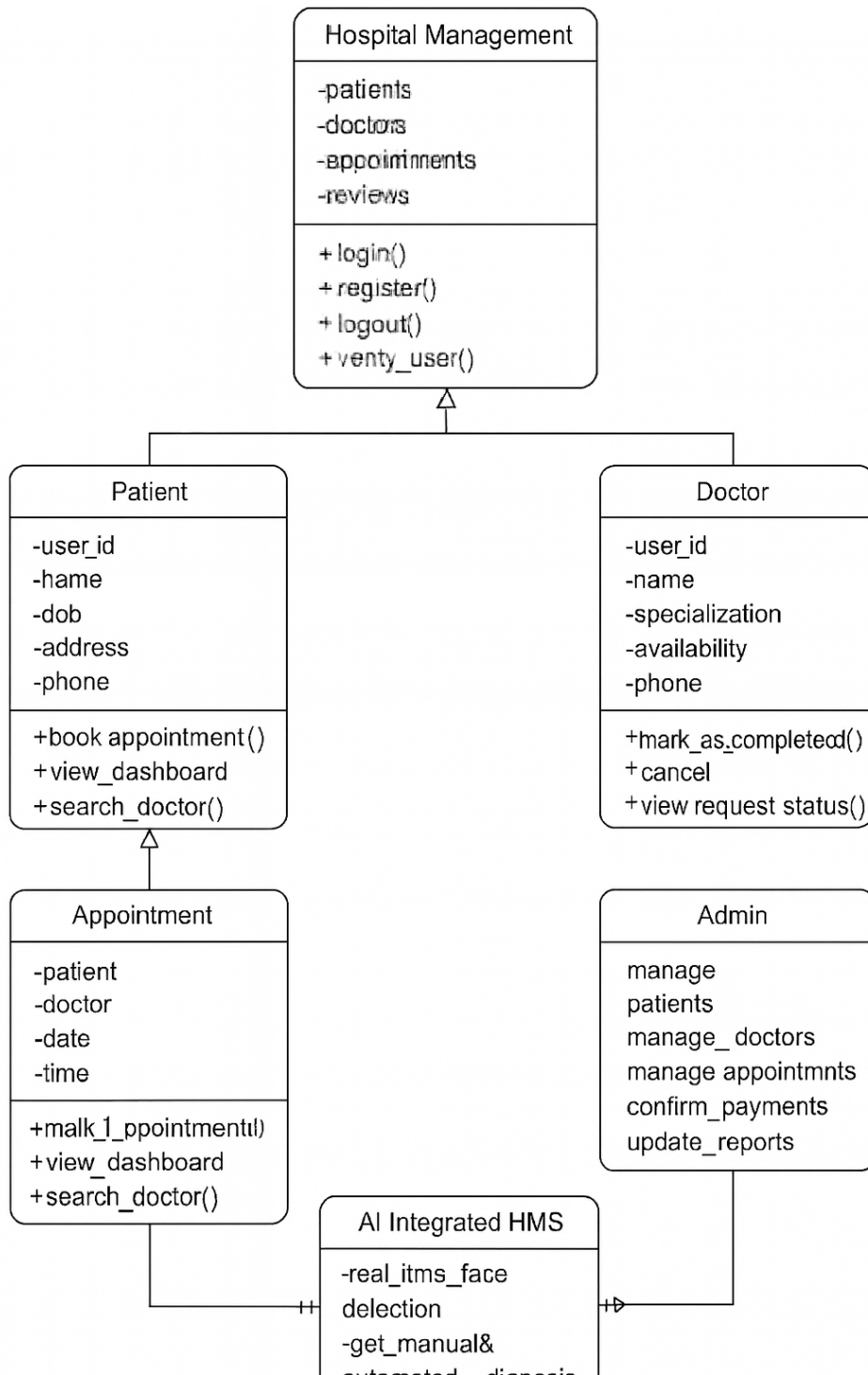


Figure 9-1: Class Diagram

9.3 High-fidelity prototype of the system:



Figure 9-8: High Fidelity prototype (Home page)

📅 Book Your Appointment

Select your preferred doctor and schedule your visit

👤 Select Doctor

📅 Appointment Date

🕒 Preferred Time

📌 Appointment Status

🕒 Pending Approval
Will be reviewed by admin

🗨️ How would you like to share your symptoms?

Write Symptoms (Manual) ▼

-- Select an option --

Write Symptoms (Manual)

Automatic Symptoms Analysis

Please describe your symptoms or reason for the appointment.

📌 Important Information

- Your appointment request will be reviewed by our administration team
- You will receive confirmation once approved
- Please arrive 15 minutes early for your appointment
- Bring a valid ID and insurance card if applicable

← Back to Dashboard

📅 Submit Appointment Request

Figure 9-9: High fidelity prototype (Doctor Appointment Page)

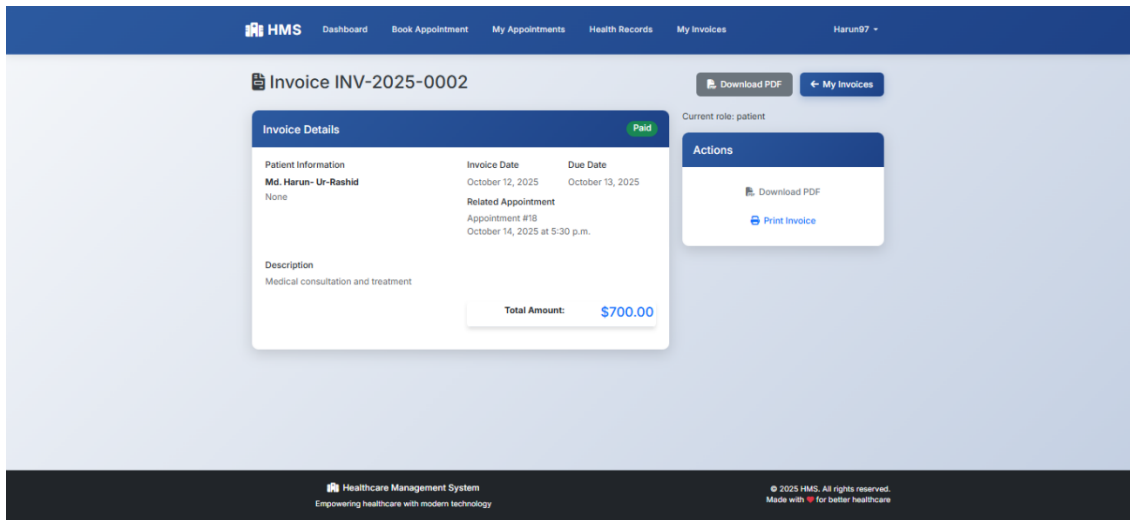


Figure 9-10: High fidelity prototype (Invoice page)

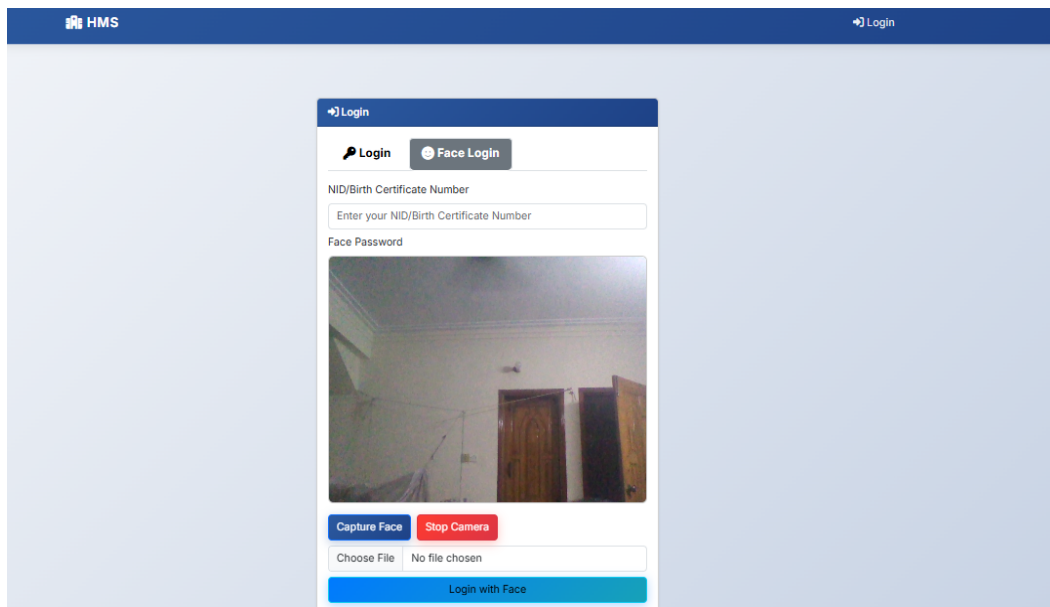


Figure 9-11: High fidelity prototype (Face Login Page)

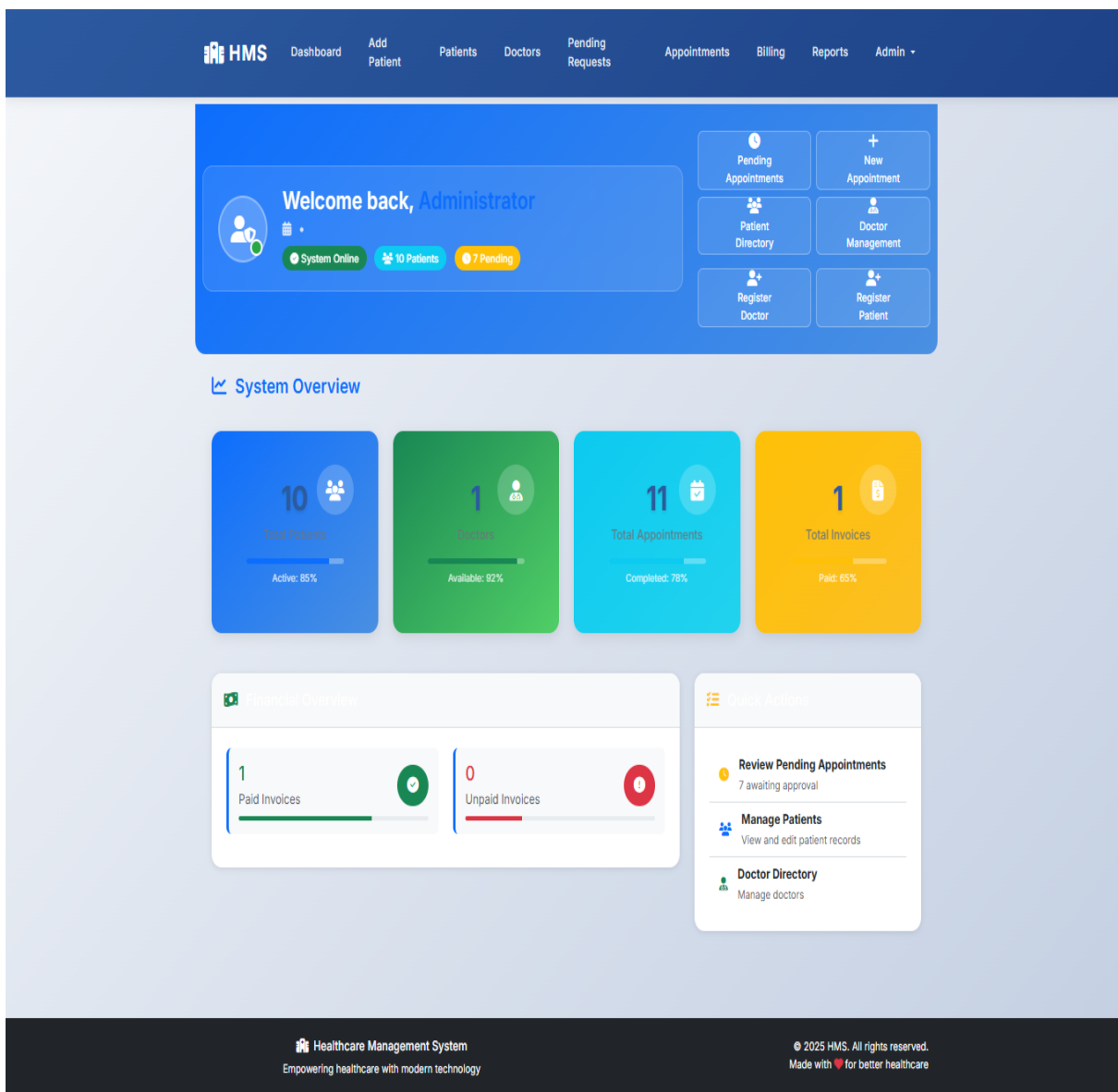


Figure 9-12: High fidelity prototype (Admin dashboard page)

Chapter 10

Development

10.1 Folder structure of the system:

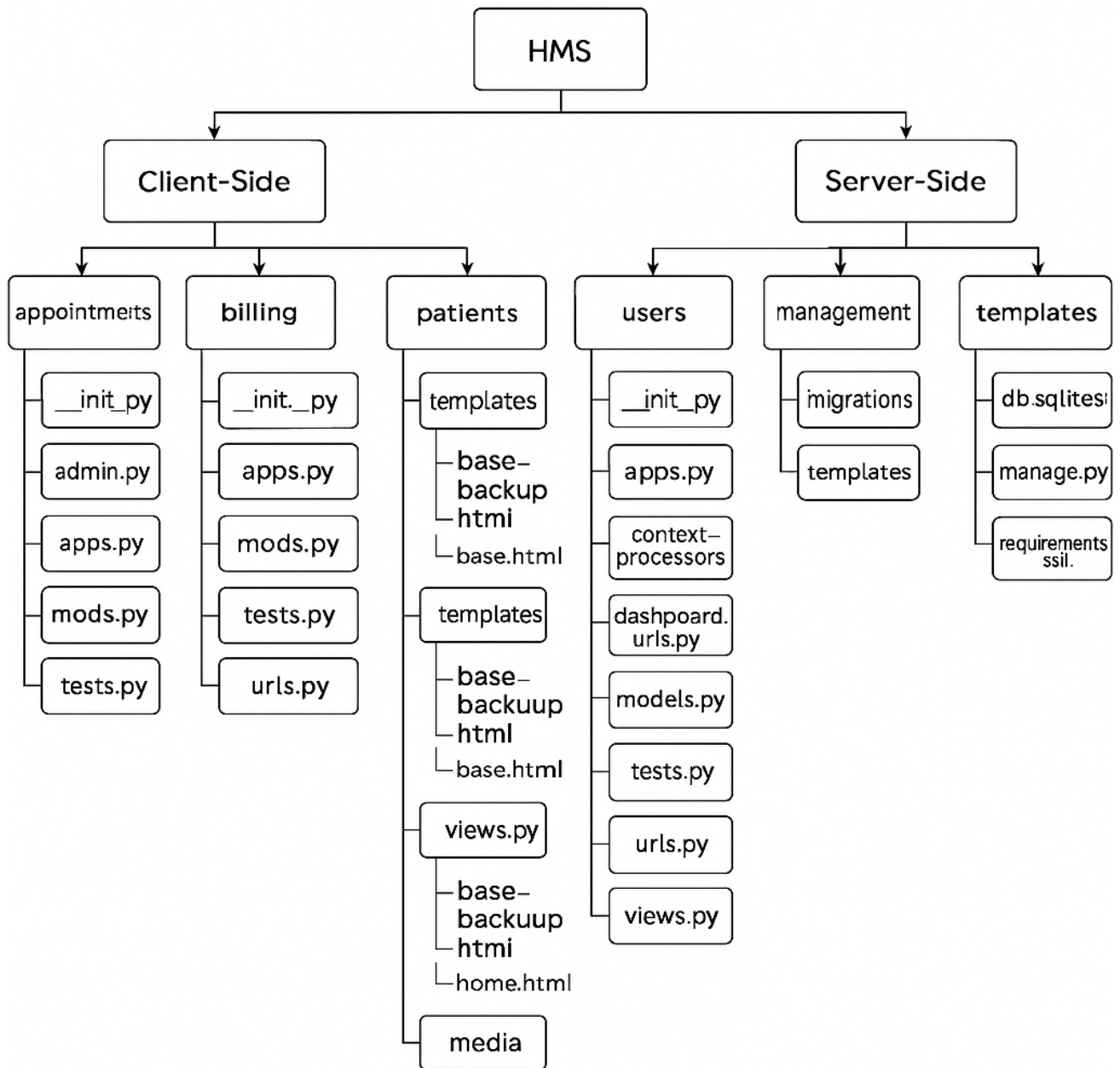


Figure 10-1: Client-side and Server-side Folder structure

Core module output sample:

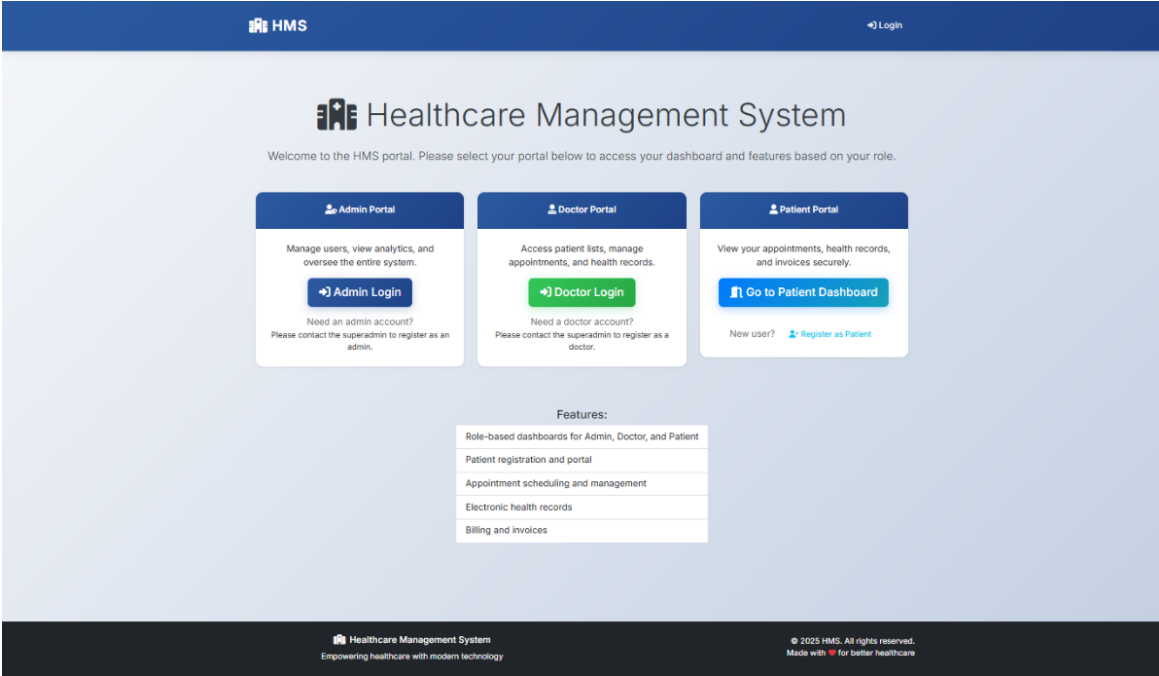


Figure 10-2: Output of Home page

Book Your Appointment

Select your preferred doctor and schedule your visit

Select Doctor

Appointment Date

Preferred Time

Appointment Status

Pending Approval
Will be reviewed by admin

How would you like to share your symptoms?

Important Information

- Your appointment request will be reviewed by our administration team
- You will receive confirmation once approved
- Please arrive 15 minutes early for your appointment
- Bring a valid ID and insurance card if applicable

Figure 10-3: Patient (Request for Booking Appointment Doctor)

Book Your Appointment

Select your preferred doctor and schedule your visit

Select Doctor

Appointment Date

Preferred Time

Appointment Status

Pending Approval
Will be reviewed by admin

How would you like to share your symptoms?

Please describe your symptoms or reason for the appointment.

Important Information

- Your appointment request will be reviewed by our administration team
- You will receive confirmation once approved
- Please arrive 15 minutes early for your appointment
- Bring a valid ID and insurance card if applicable

Figure 10-4: Patient (Request for Booking Appointment Doctor with manual and automatic diagnosis options)

The screenshot displays a web form titled "Book Your Appointment" with the subtitle "Select your preferred doctor and schedule your visit". The form is organized into several sections:

- Select Doctor:** A dropdown menu showing "Dr. Abdur Rahman - Dermatology".
- Appointment Date:** A date picker set to "10/14/2025".
- Preferred Time:** A time selection field currently showing "--:-- --".
- Appointment Status:** A section with a yellow "Pending Approval" label and the text "Will be reviewed by admin".
- How would you like to share your symptoms?:** A dropdown menu set to "Write Symptoms (Manual)".
- Write Symptoms (Manual):** A large text area with the placeholder text "Any additional notes or symptoms".
- Important Information:** A light blue box containing a list of instructions:
 - Your appointment request will be reviewed by our administration team
 - You will receive confirmation once approved
 - Please arrive 15 minutes early for your appointment
 - Bring a valid ID and insurance card if applicable

At the bottom of the form, there are two buttons: "Back to Dashboard" and "Submit Appointment Request".

Figure 10-5: Patient (Request for Booking Appointment Doctor with manual diagnosis)

Book Your Appointment

Select your preferred doctor and schedule your visit.

Select Doctor

Dr. Abdur Rahman - Dermatology

Appointment Date

10/14/2025

Preferred Time

---:-- --

Appointment Status

Pending Approval

Will be reviewed by admin

How would you like to share your symptoms?

Automatic Symptoms Analysis

Automatic Symptoms Analysis

<p>What is your gender?</p> <p>Select...</p>	<p>What is your age?</p> <p>Enter your age</p>
<p>What is your height?</p> <p>Enter your height</p>	<p>What is your weight?</p> <p>Enter your weight</p>
<p>Do you have Diabetes?</p> <p>Select...</p>	<p>How much do you have Diabetes?</p> <p>Enter amount</p>
<p>Do you smoke?</p> <p>Select...</p>	<p>How much water do you drink daily?</p> <p>Select...</p>
<p>How often do you have physical activity?</p> <p>Select...</p>	<p>Time using technological devices?</p> <p>Select...</p>
<p>How often do you drink alcohol?</p> <p>Select...</p>	<p>Transportation you usually use?</p> <p>Select...</p>
<p>Eat food between meals?</p> <p>Select...</p>	<p>Eat vegetables in meals?</p> <p>Select...</p>
<p>Eat high-caloric food frequently?</p> <p>Select...</p>	

Run Analysis

Important Information

- Your appointment request will be reviewed by our administration team
- You will receive confirmation once approved
- Please arrive 15 minutes early for your appointment
- Bring a valid ID and insurance card if applicable

← Back to Dashboard

Submit Appointment Request

Figure 10-6: Patient (Request for Booking Appointment Doctor with automatic diagnosis)

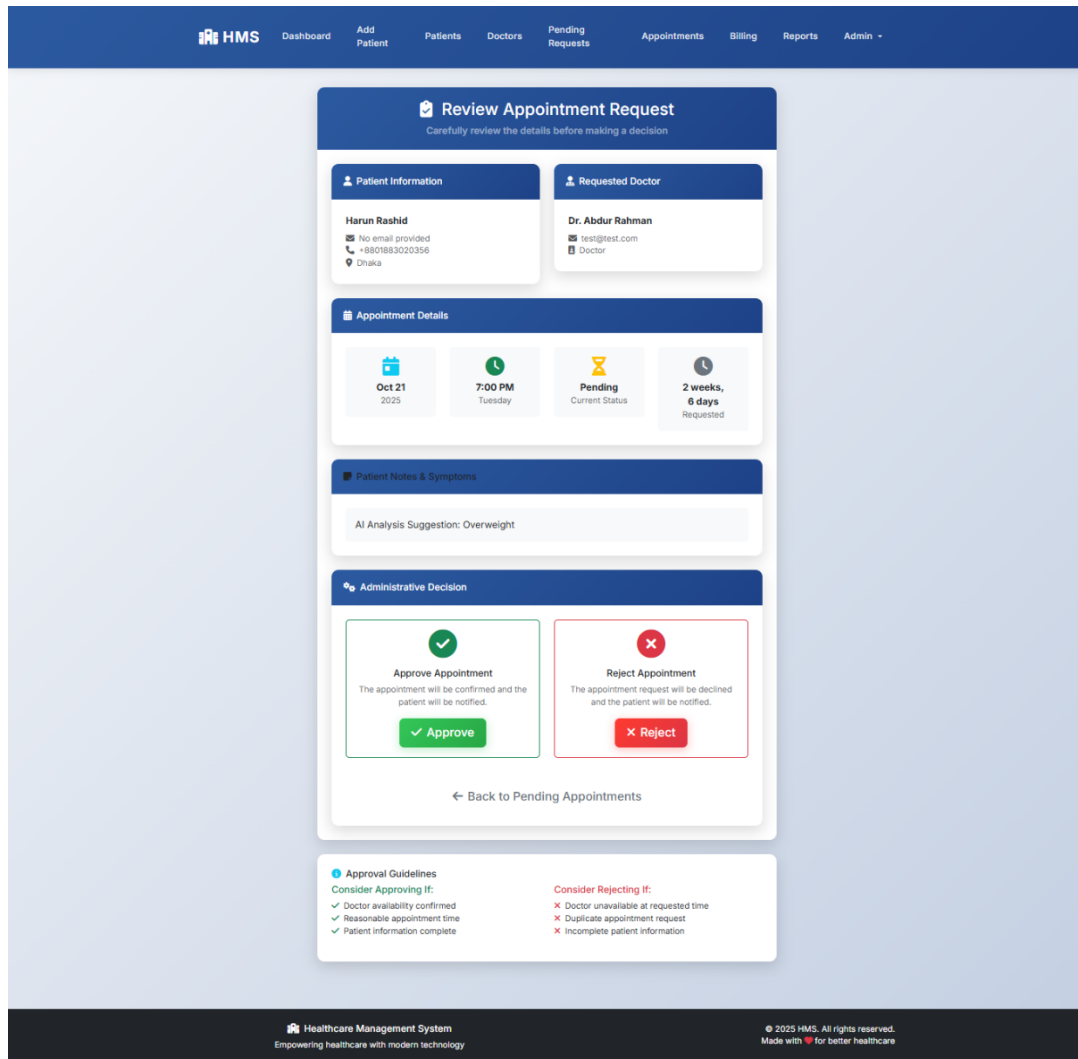


Figure 10-7: Admin (Approve or reject Appointment Doctor)

The screenshot displays the 'Edit Appointment' form within the HMS Admin interface. The form is titled 'Edit Appointment' and is located in the 'Dashboard / Edit Appointment' section. The form fields are as follows:

- Patient***: Harun Rashid
- Doctor**: testdoctor (Doctor)
- Date***: 10/21/2025
- Time***: 07:00 PM
- Status***: Pending Approval
- Notes**: AI Analysis Suggestion: Overweight
- Analysis json**: Overweight

An 'Update' button is located at the bottom of the form. The footer of the page contains the text: 'Healthcare Management System Empowering healthcare with modern technology' and '© 2025 HMS. All rights reserved. Made with ❤️ for better healthcare'.

Figure 10-8: Admin (Update information for Booking Appointment)

HMS Dashboard Add Patient Patients Doctors Pending Requests Appointments Billing Reports Admin

Pending Appointments

Review and approve appointment requests from patients [All Appointments](#)

Appointment Request #12 2 weeks, 6 days ago

Patient Details
 Harun Rashid
 No email
 +8801883020356

Requested Doctor
 Dr. Abdur Rahman
 test@test.com

Appointment Details
 Monday, September 08, 2025
 7:00 PM

Status
Pending

[Review & Decide](#)

Appointment Request #11 2 weeks, 6 days ago

Patient Details
 Harun Rashid
 No email
 +8801883020356

Requested Doctor
 Dr. Abdur Rahman
 test@test.com

Appointment Details
 Tuesday, September 23, 2025
 6:33 PM

Status
Pending

[Review & Decide](#)

Appointment Request #9 2 weeks, 6 days ago

Patient Details
 Harun Rashid
 No email
 +8801883020356

Requested Doctor
 Dr. Abdur Rahman
 test@test.com

Appointment Details
 Tuesday, September 23, 2025
 7:00 PM

Patient Notes
 Fever

Status
Pending

[Review & Decide](#)

Appointment Request #10 2 weeks, 6 days ago

Patient Details
 Harun Rashid
 No email
 +8801883020356

Requested Doctor
 Dr. Abdur Rahman
 test@test.com

Appointment Details
 Tuesday, September 23, 2025
 7:00 PM

Status
Pending

[Review & Decide](#)

Appointment Request #15 2 weeks, 6 days ago

Patient Details
 Harun Rashid
 No email
 +8801883020356

Requested Doctor
 Dr. Abdur Rahman
 test@test.com

Appointment Details
 Thursday, September 25, 2025
 7:00 PM

Patient Notes
 AI Analysis Suggestion: Normal Weight

Status
Pending

[Review & Decide](#)

Appointment Request #18 4 minutes ago

Patient Details
 Md. Harun-Ur-Rashid
 No email
 +8801701085147

Requested Doctor
 Dr. Abdur Rahman
 test@test.com

Appointment Details
 Tuesday, October 14, 2025
 5:30 PM

Patient Notes
 Diabetes: Yes Diabetes Amount: 7.9 AI Analysis Suggestion: Overweight

Status
Pending

[Review & Decide](#)

Appointment Request #16 2 weeks, 6 days ago

Patient Details
 Harun Rashid
 No email
 +8801883020356

Requested Doctor
 Dr. Abdur Rahman
 test@test.com

Appointment Details
 Tuesday, October 21, 2025
 7:00 PM

Patient Notes
 AI Analysis Suggestion: Overweight

Status
Pending

[Review & Decide](#)

Need to manage appointments?

[View All Appointments](#)
[Schedule New Appointment](#)
[Back to Dashboard](#)

Healthcare Management System
 Empowering healthcare with modern technology

© 2025 HMS. All rights reserved.
 Made with for better healthcare

Figure 10-9: Admin (Pending lists for Booking Appointment)

HMS Dashboard Add Patient Patients Doctors Pending Requests Appointments Billing Reports Admin

All Appointments Pending Appointments

Patient	Doctor	Date	Time	Status	Notes	Approved By	Actions
Harun Rashid	Dr. Abdur Rahman	Oct. 21, 2025	7 p.m.	Pending	AI Analysis Suggestion: Overweight	-	Review Delete Edit
Md. Harun- Ur-Rashid	Dr. Abdur Rahman	Oct. 14, 2025	5:30 p.m.	Pending	Diabetes: Yes Diabetes Amount: 7.9 ...	-	Review Delete Edit
Harun Rashid	Dr. Abdur Rahman	Sept. 28, 2025	6:50 p.m.	Completed	AI Analysis Suggestion: Overweight	admin Sep 26, 12:53	Edit Delete
Harun Rashid	Dr. Abdur Rahman	Sept. 25, 2025	7 p.m.	Pending	AI Analysis Suggestion: Normal Weight	-	Review Delete Edit
Harun Rashid	Dr. Abdur Rahman	Sept. 23, 2025	9:08 p.m.	Scheduled	AI Analysis Suggestion: Overweight	admin Sep 22, 13:25	Edit Delete
Harun Rashid	Dr. Abdur Rahman	Sept. 23, 2025	7 p.m.	Pending	Fever	-	Review Delete Edit
Harun Rashid	Dr. Abdur Rahman	Sept. 23, 2025	7 p.m.	Pending	No notes	-	Review Delete Edit
Harun Rashid	Dr. Abdur Rahman	Sept. 23, 2025	6:33 p.m.	Pending	No notes	-	Review Delete Edit
Harun Rashid	Dr. Abdur Rahman	Sept. 23, 2025	9 a.m.	Scheduled	AI Analysis Suggestion: Overweight	admin Sep 22, 16:01	Edit Delete
Harun Rashid	Dr. Abdur Rahman	Sept. 8, 2025	7 p.m.	Pending	No notes	-	Review Delete Edit
Mashrafi Mia	Dr. Abdur Rahman	Aug. 8, 2025	3:31 p.m.	Scheduled	fever	admin Aug 05, 19:20	Edit Delete

Healthcare Management System Empowering healthcare with modern technology © 2025 HMS. All rights reserved. Made with for better healthcare

Figure 10-10: Admin (All Appointments details with manual review, edit and delete options)

Book Your Appointment

Select your preferred doctor and schedule your visit

Select Doctor

Dr. Abdur Rahman - Dermatology

Appointment Date

10/14/2025

Preferred Time

05:30 PM

Appointment Status

Pending Approval

Will be reviewed by admin

How would you like to share your symptoms?

Automatic Symptoms Analysis

Automatic Symptoms Analysis

What is your gender?

Male

What is your age?

20

What is your height?

120

What is your weight?

100

Do you have Diabetes?

Yes

How much do you have Diabetes?

7.9

Do you smoke?

No

How much water do you drink daily?

0-1 liter

How often do you have physical activity?

None

Time using technological devices?

3-5 hours

How often do you drink alcohol?

sometimes

Transportation you usually use?

Automobile

Eat food between meals?

Frequently

Eat vegetables in meals?

Sometimes

Eat high-caloric food frequently?

Yes

Run Analysis

Health Analysis Result

Overweight

Important Information

- Your appointment request will be reviewed by our administration team
- You will receive confirmation once approved
- Please arrive 15 minutes early for your appointment
- Bring a valid ID and insurance card if applicable

← Back to Dashboard

Submit Appointment Request

Figure 10-11: Patient (Automatic Diagnosis result in Appointment)

HMS Dashboard Add Patient Patients Doctors Pending Requests Appointments Billing Reports Admin

Billing Dashboard

Manage invoices and payments

1
Total Invoices

1
Paid Invoices

\$500.00
Total Revenue

\$0.00
Pending Amount

Quick Actions

+ Create Invoice

All Invoices

Unpaid (0)

Overdue (0)

Recent Invoices

Invoice #	Patient	Amount	Status	Date	Actions
INV-2025-0001	Mashrafi Mia	\$500.00	Paid	Aug 05, 2025	View Print

Healthcare Management System
Empowering healthcare with modern technology

© 2025 HMS. All rights reserved.
Made with ❤️ for better healthcare

Figure 10-12: Admin (Billing Dashboard)

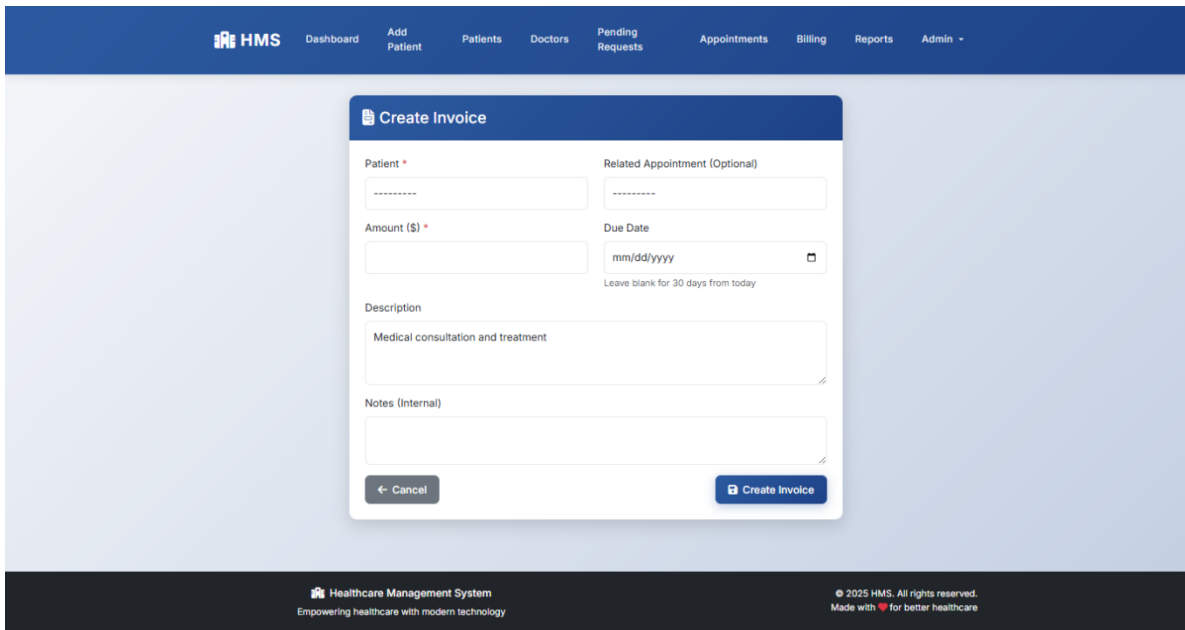


Figure 10-13: Admin (Create Invoice)

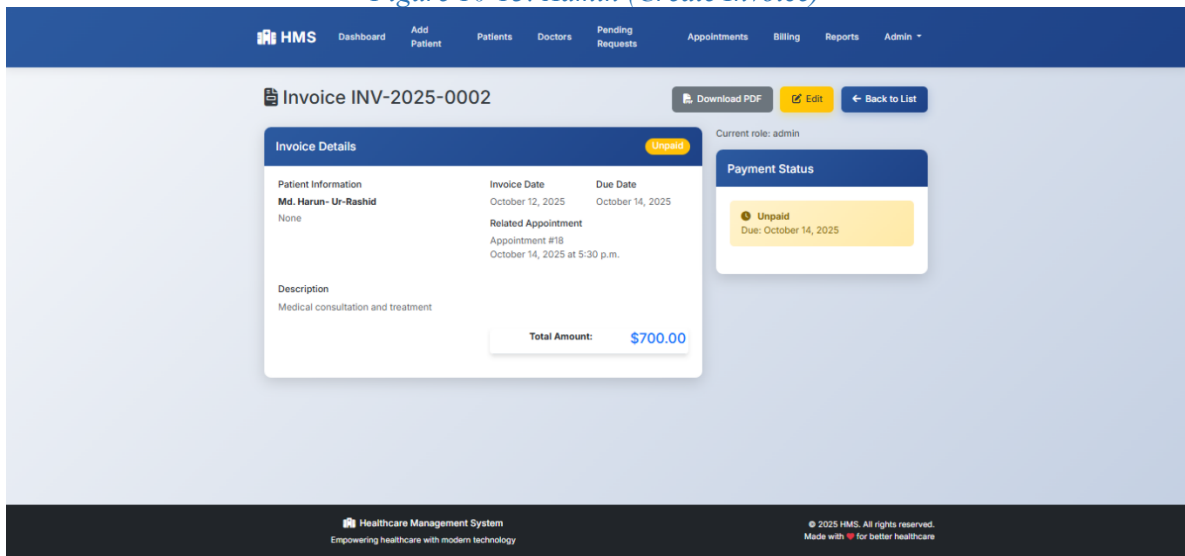


Figure 10-14: Admin (Invoice details with edit and download pdf options)

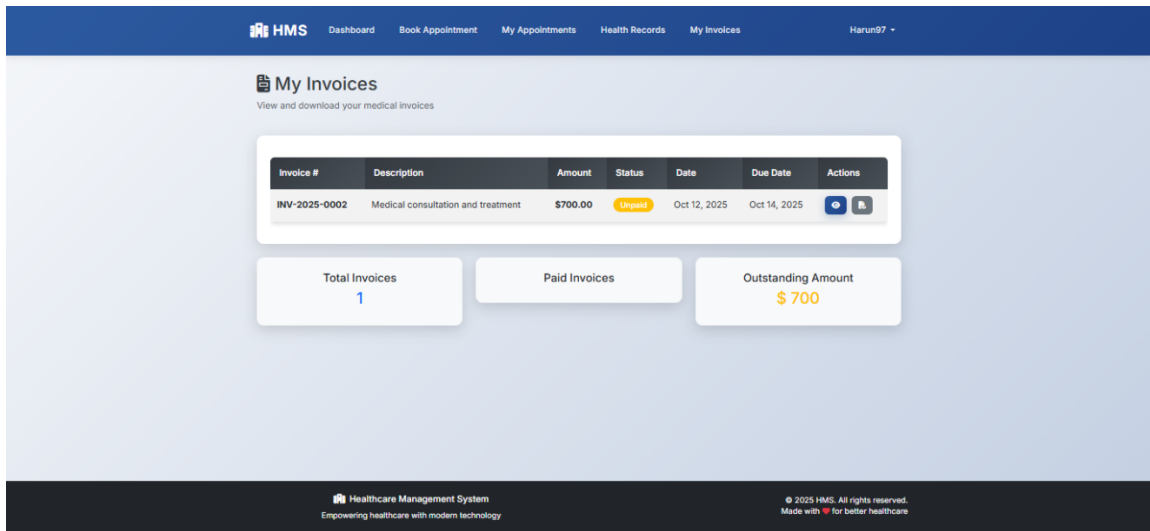


Figure 10-15: Patient (Invoice details with several actions)

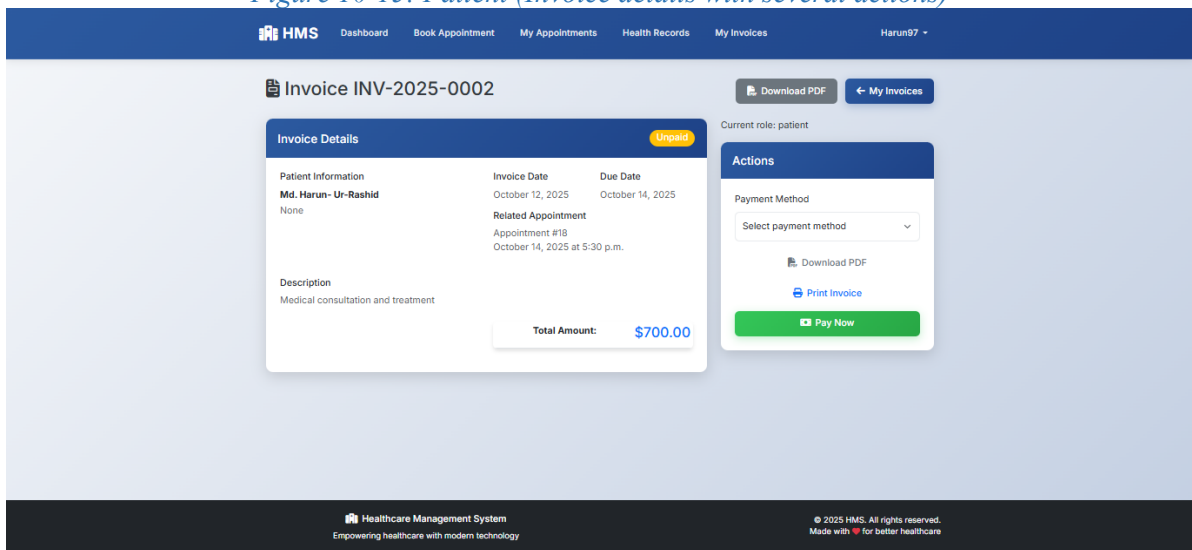


Figure 10-16: Patient ((Invoice details with payment method, download and print pdf)

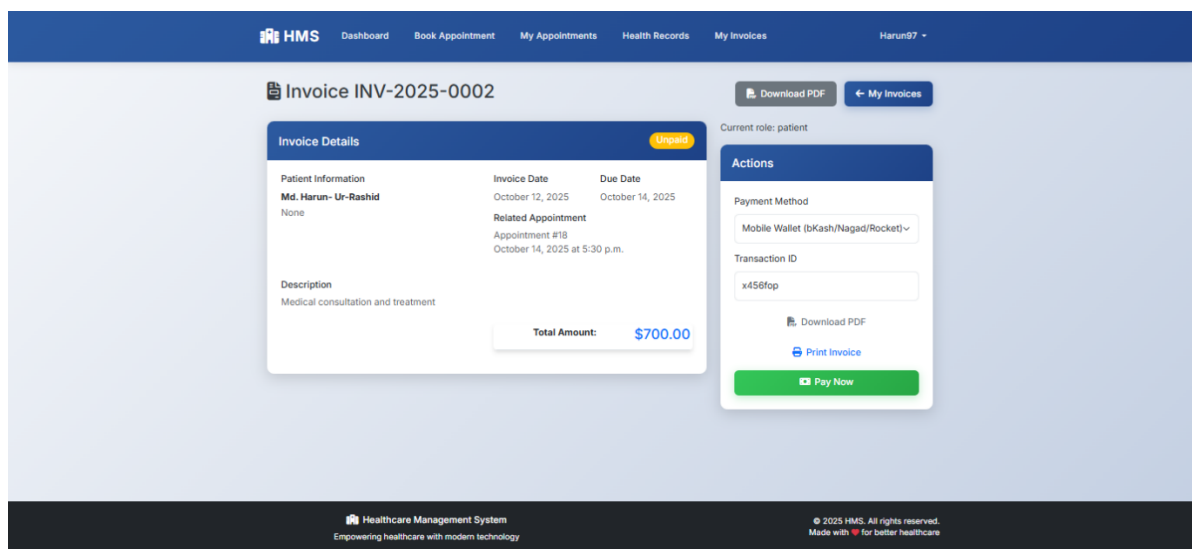


Figure 10-17: Patient ((Invoice details with mobile wallet payment method, download and print pdf)

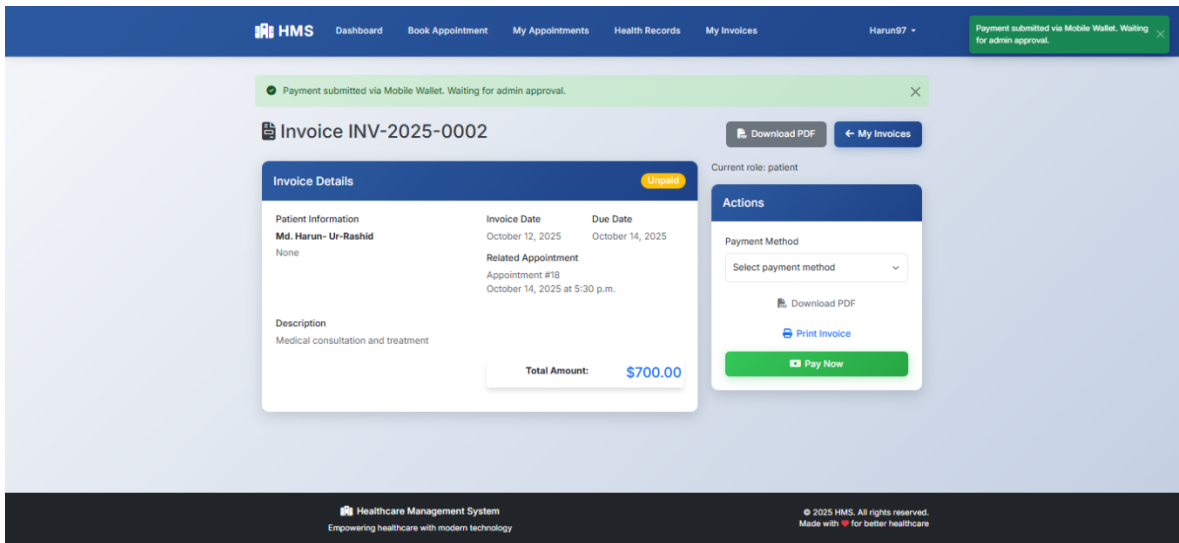


Figure 10-17: Patient ((Invoice details with pending mobile wallet payment)

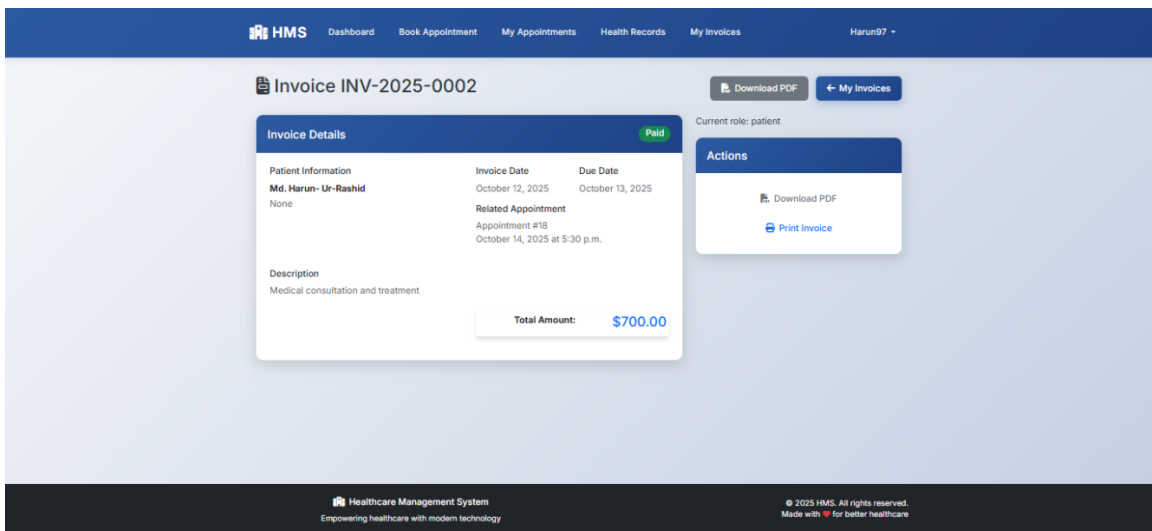


Figure 10-18: Patient ((Invoice details with paid status of mobile wallet payment)

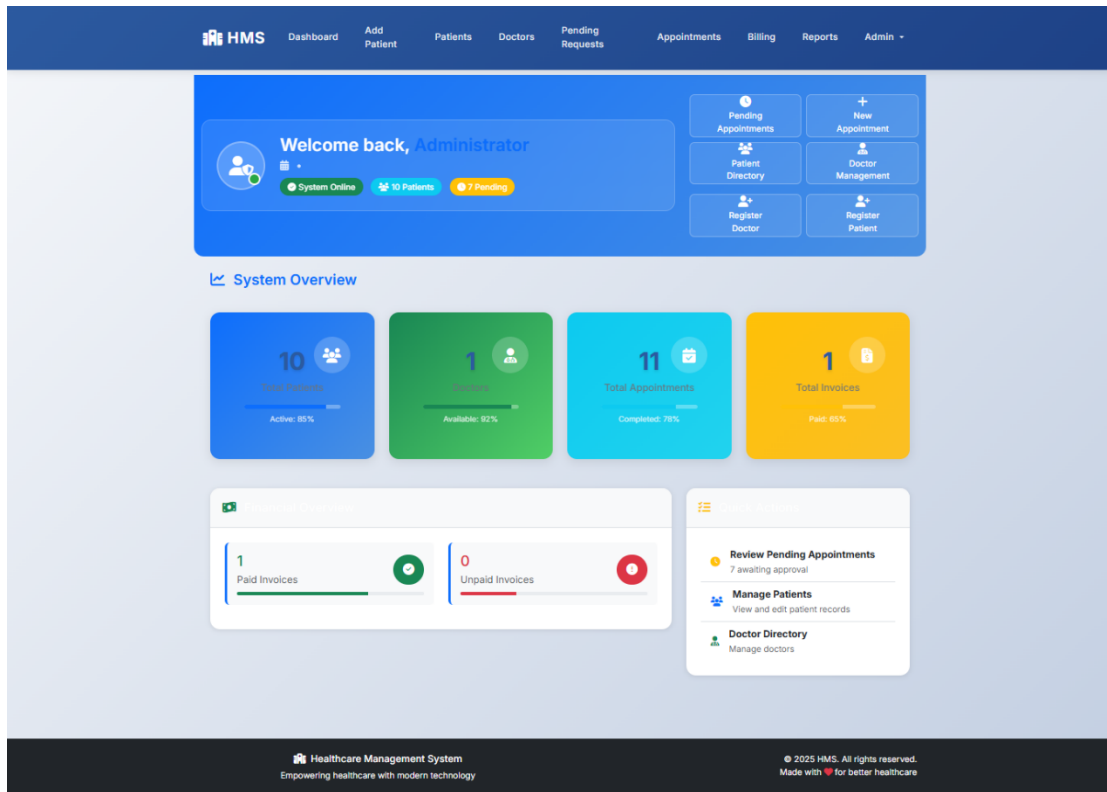


Figure 10-18: Admin Dashboard

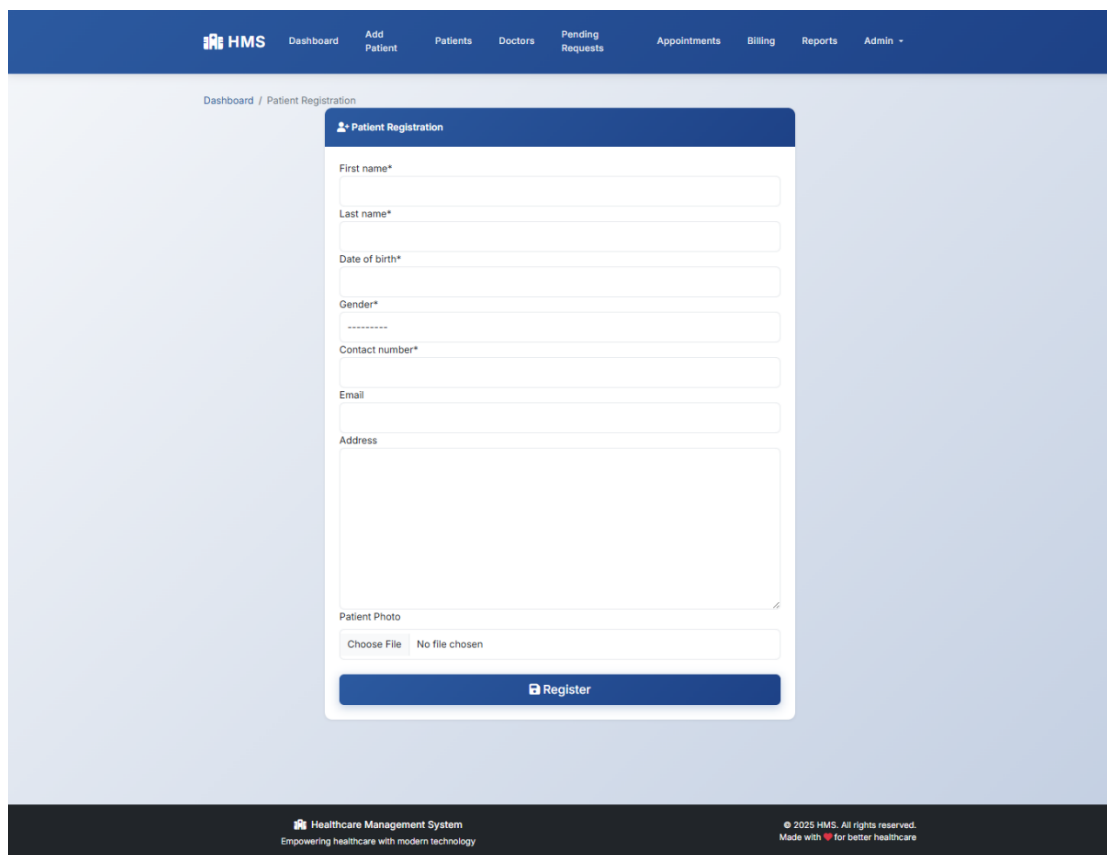


Figure 10-19: Admin (patient register)

Register New Doctor

User Account Information

Username *
admin

Email *

First Name *

Last Name *

Password *

Confirm Password *

Medical Information

Medical License Number *

Years of Experience *

Specialization *
General Practice

Department *
Emergency Department

Education *

Educational background and qualifications

Professional Details

Phone Number

Office Location

Consultation Fee (\$)
0.0

Available Days
Monday to Friday

Available Hours
9:00 AM - 5:00 PM

Professional Biography

Brief professional biography (optional)

[← Back to Dashboard](#) [Register Doctor](#)

Healthcare Management System
Empowering healthcare with modern technology

© 2025 HMS. All rights reserved.
Made with ❤️ for better healthcare

Figure 10-20: Admin (register doctor)

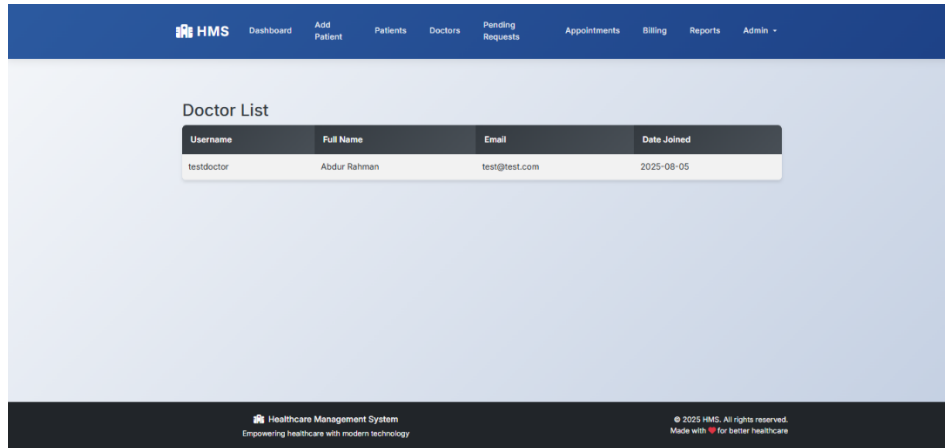


Figure 10-21: Admin (view doctor list)

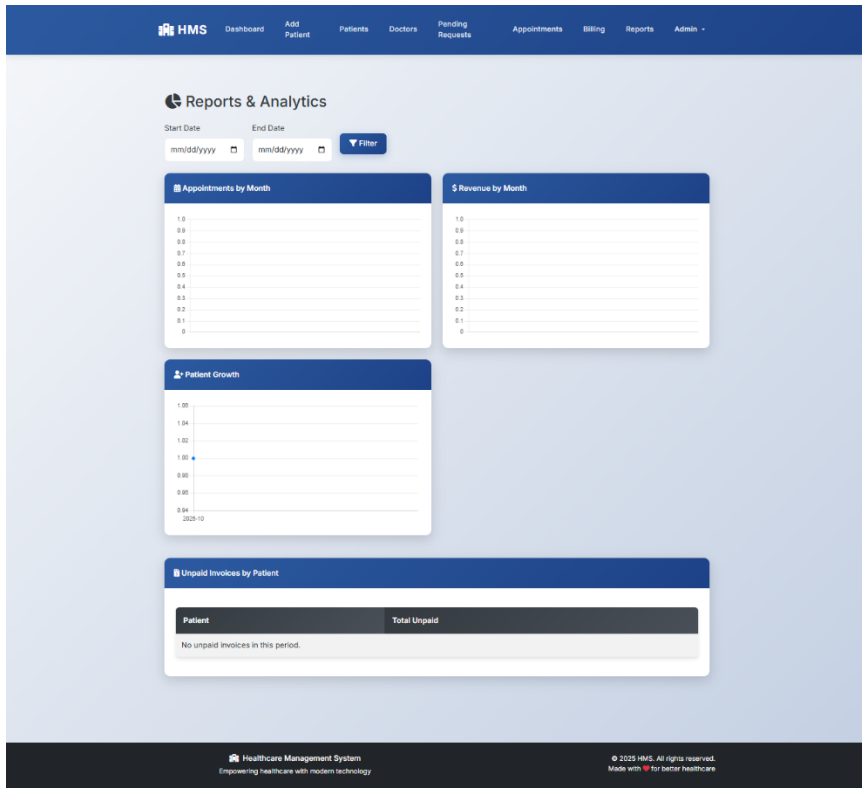


Figure 10-22: Admin (Reports and Analytics)

HMS Login

Patient Registration

Create your patient account to book appointments, view health records, and manage your healthcare.

Account Information

NID/Birth Certificate Number *

Email Address *

Password *

Confirm Password *

Personal Information

First Name *

Last Name *

Date of Birth *

Gender *

Document Type

Contact Information

Contact Number *

Address

Profile Image

Capture or Upload Image *

Choose File

This will be used for facial recognition login.

I agree to the Terms and Conditions and Privacy Policy *

Already have an account? [Login here](#)

Healthcare Management System
Empowering healthcare with modern technology © 2025 HMS. All rights reserved.
Made with ❤️ for better healthcare

Figure 10-23: Patient (Registration with capture and upload image options)

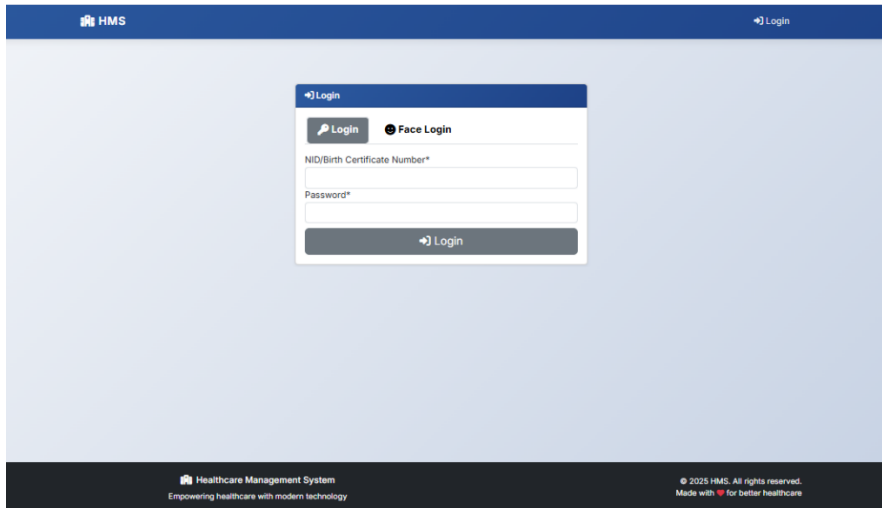


Figure 10-24: Login (All users with manual method)

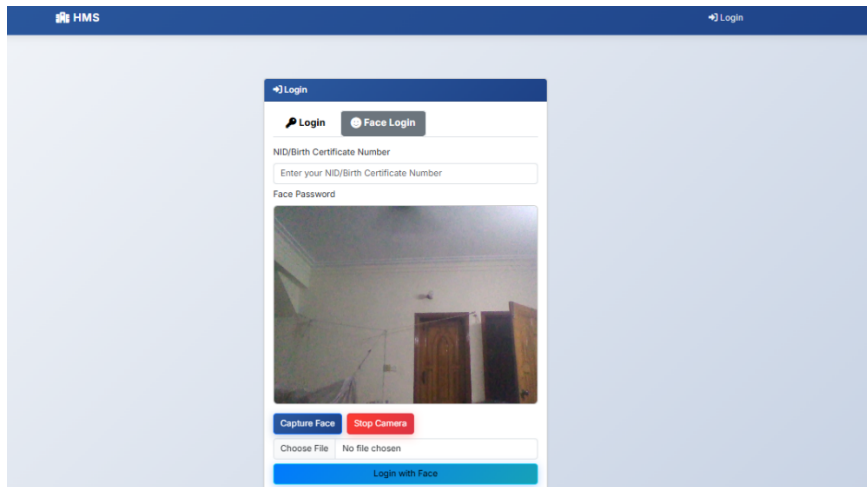


Figure 10-25: Login (All users with automatic face detection method)

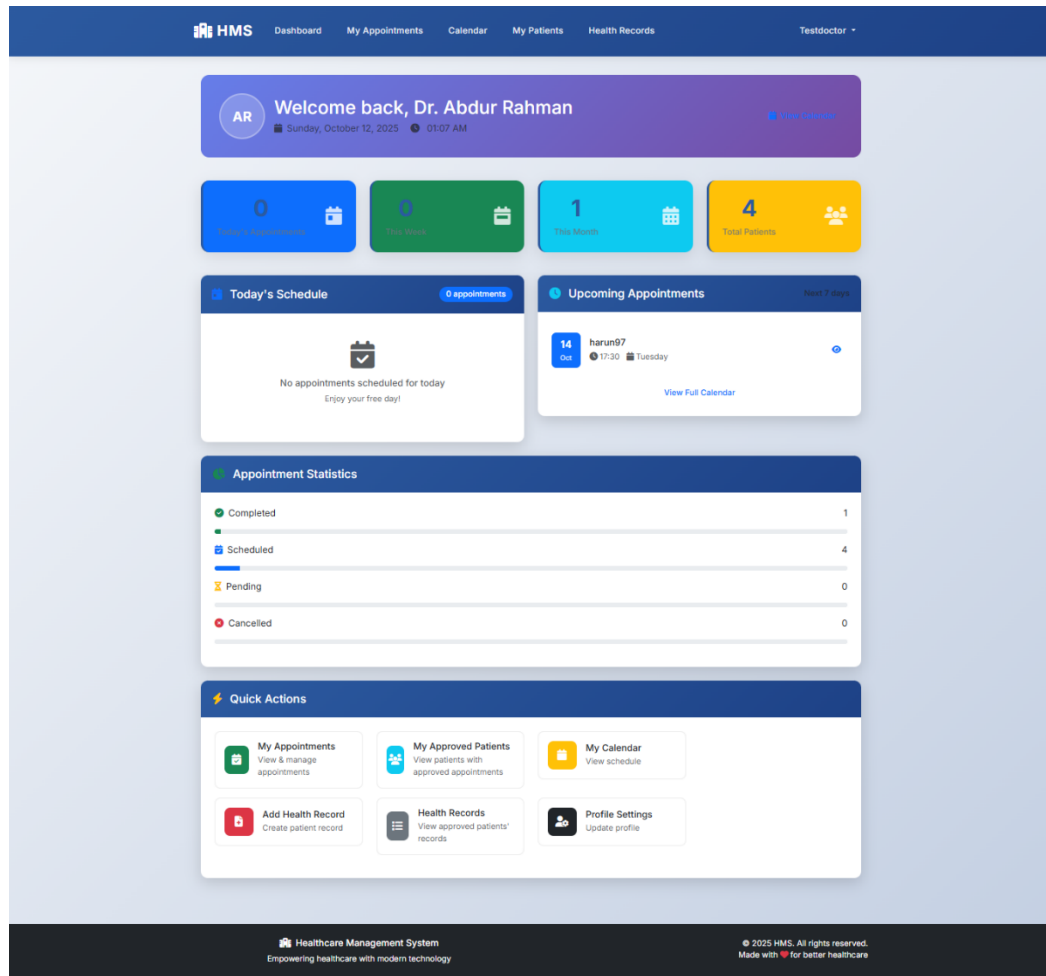


Figure 10-26: Doctor Dashboard

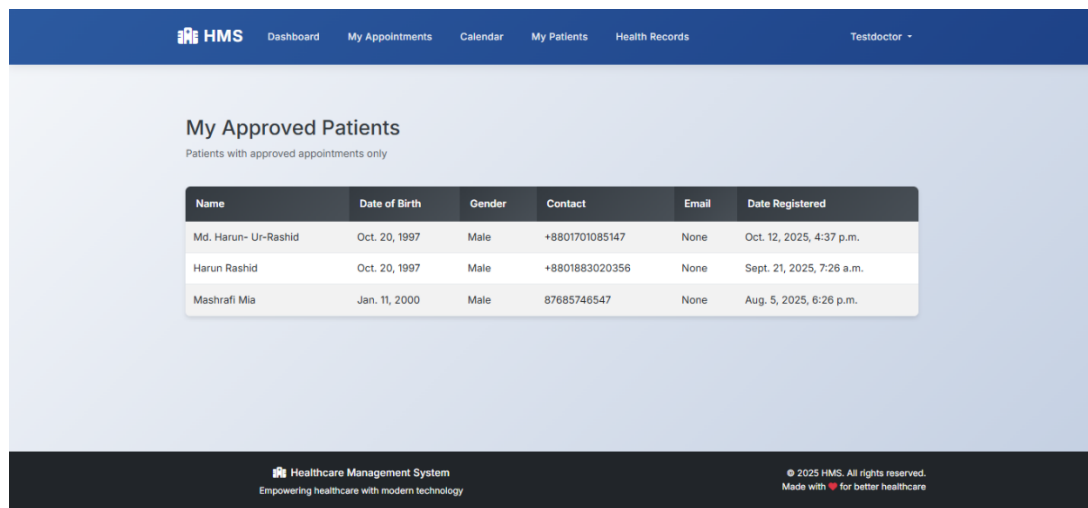


Figure 10-27: Doctor (Patient list)

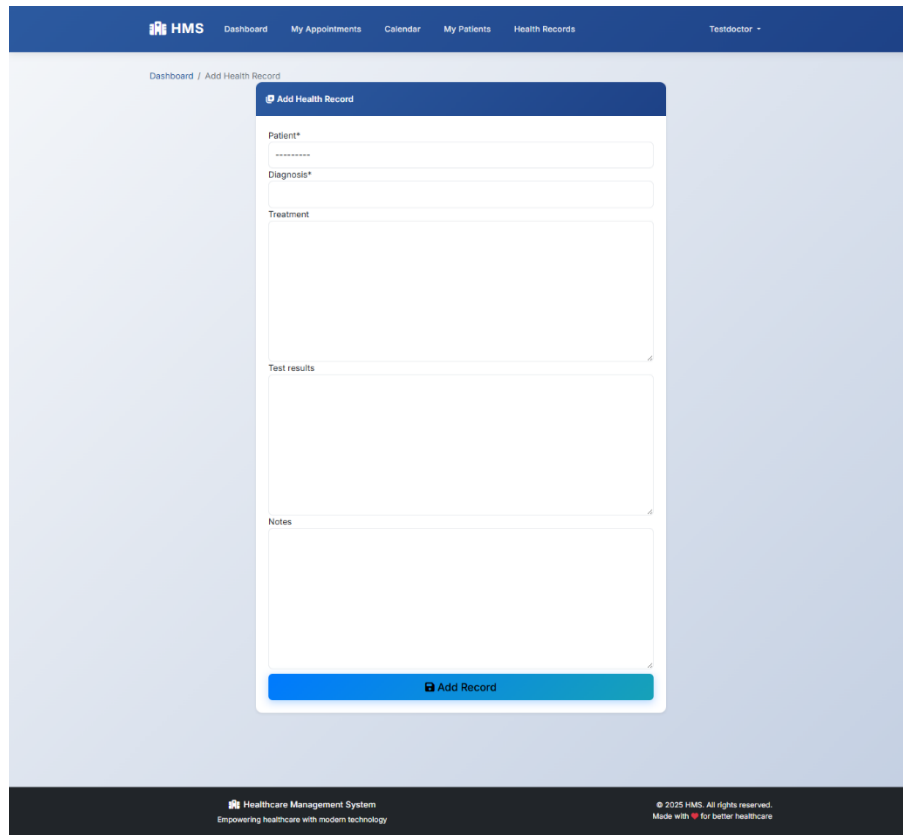


Figure 10-28: Doctor (Add Patient health record)

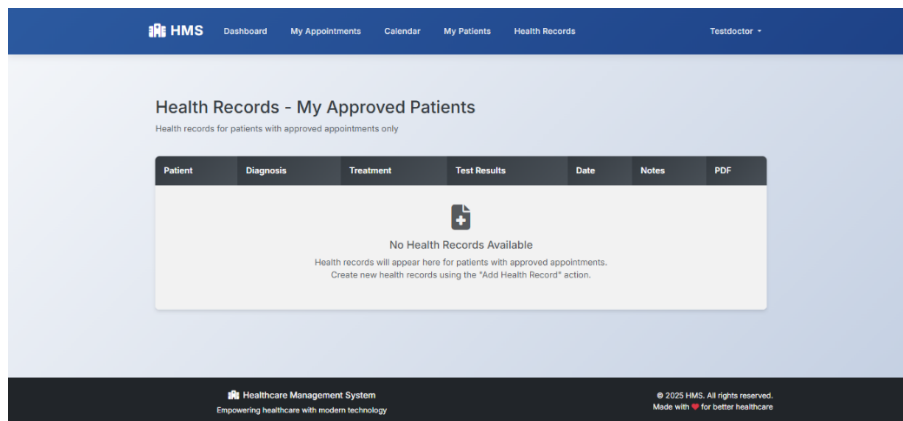


Figure 10-29: Doctor (View Patient health record list)

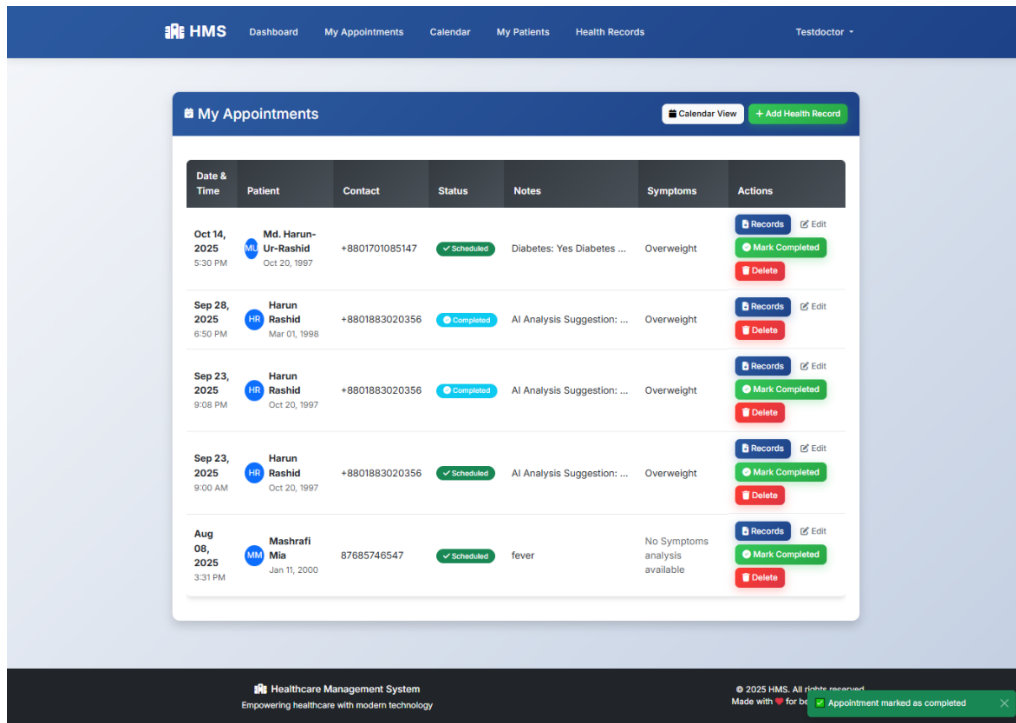


Figure 10-30: Doctor (Patient appointment list with recording, marking, editing and deleting options)

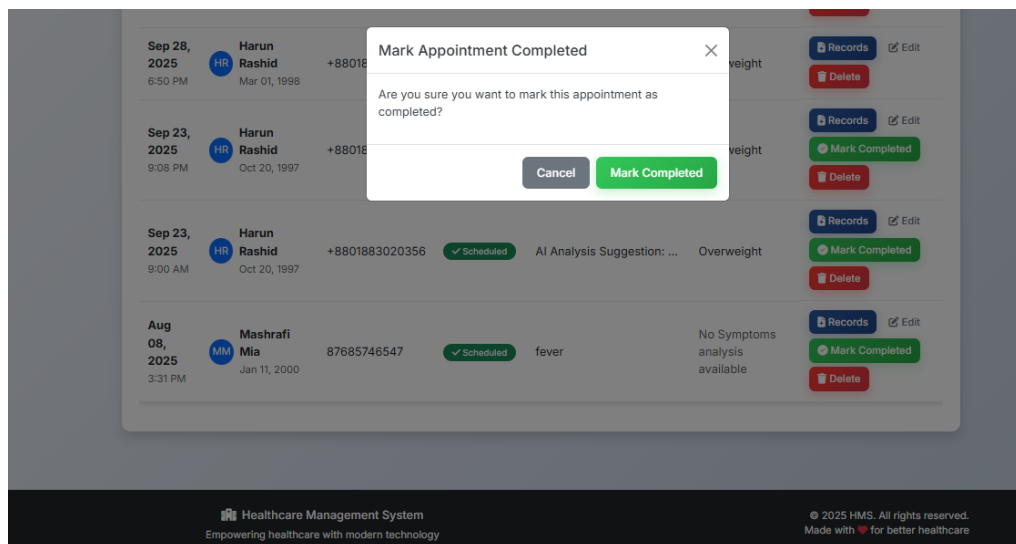


Figure 10-31: Doctor (Patient appointment list with marking completed)

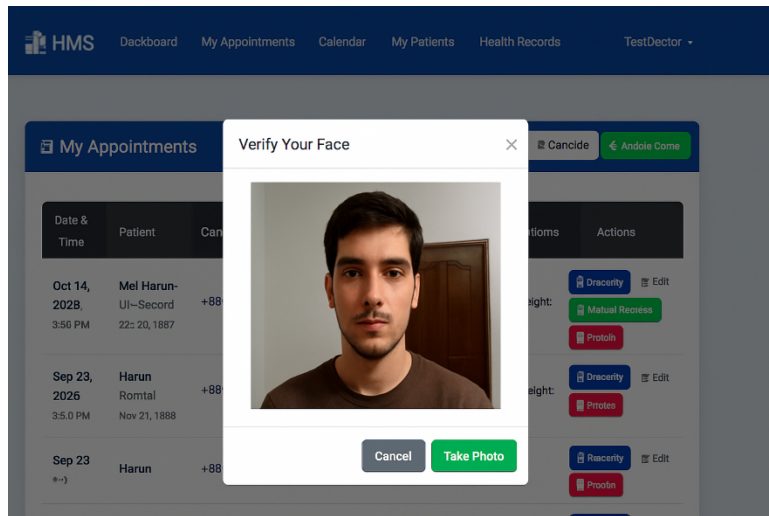


Figure 10-32: Doctor (View Patient records verifying face)

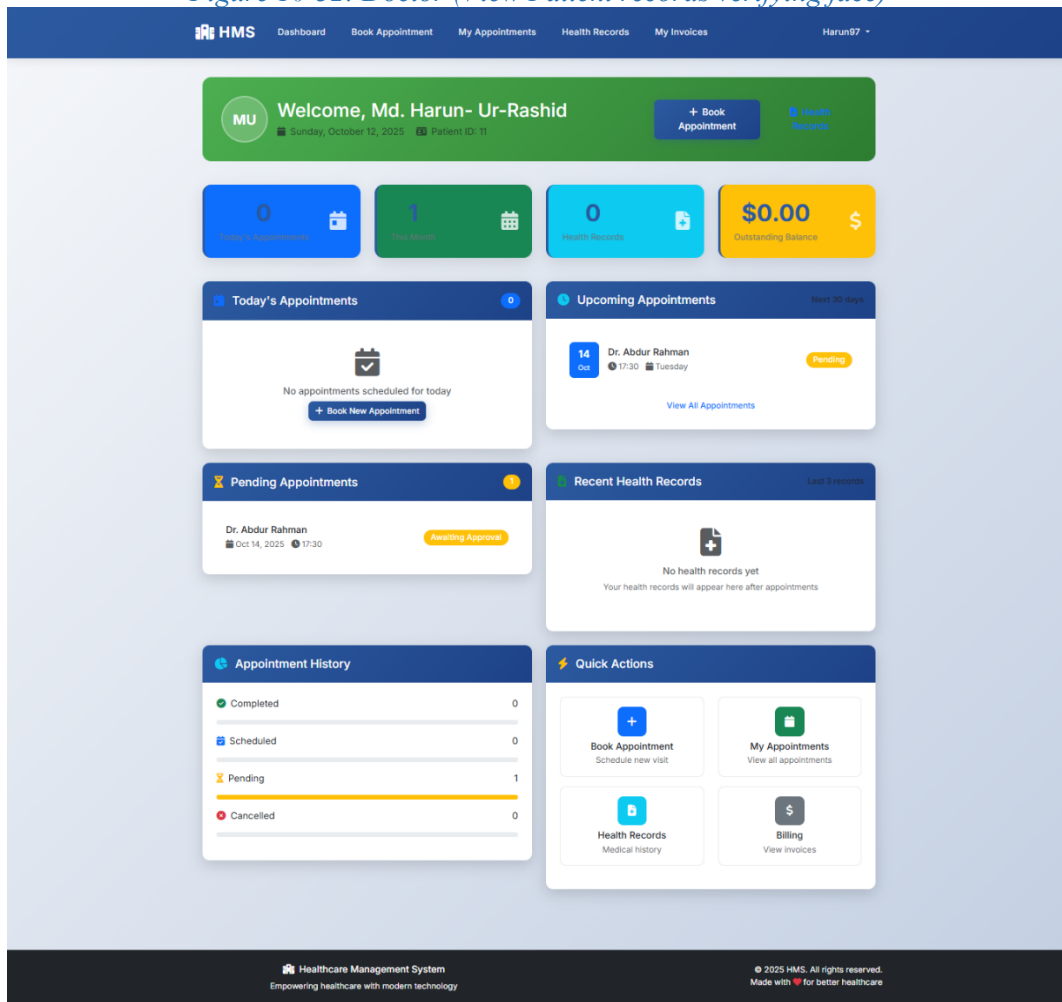


Figure 10-33: Patient Dashboard

HMS Dashboard Book Appointment My Appointments Health Records My Invoices Harun97

Your Appointments

Schedule Appointment

Date	Time	Status	Notes
Oct. 14, 2025	5:30 p.m.	Pending	Diabetes: Yes Diabetes Amount: 7.9 AI Analysis Suggestion: Overweight

Healthcare Management System
Empowering healthcare with modern technology

© 2025 HMS. All rights reserved.
Made with ❤️ for better healthcare

Figure 10-34: Patient (appointment status list)

10.2 Prioritization while developing:

Serial	Requirements
1	Patient registration and authentication with email or phone verification.
2	Doctor registration and authentication with admin approval.
3	Patient login and dashboard to access profile, bookings, and payments.
4	Doctor login and dashboard to manage availability and appointments.
5	Update reports of patients by doctor
6	Booking system for patients to request appointments with doctors.
7	Payment module for patients to make secure online payments for bookings.
8	Transaction management system for viewing and verifying payment history.
9	Real-time face detection system for patient's reports views at short time.
11	Admin login and authentication for secure system access.
12	Admin management of patients, doctors, and bookings (view, edit, delete).
13	Admin payment verification and booking confirmation.
14	Admin dashboard with analytics (total users, active doctors, completed bookings).
15	Service management by admin (add or update healthcare service types).
16	Notification system for booking confirmations and status updates.
17	Multi-device compatibility and responsive interface for mobile and desktop users.
18	Data security, privacy, and encryption for user and payment information.
19	AI-based diagnosis system in appointment

Table 10-8.1.3-1: Table of Module system

Chapter 11

Testing

11.1 Test Case:

Test Case ID	Test Scenario	Test Steps	Expected Result	Status
TC1	Patient Registration	<ol style="list-style-type: none"> 1. Navigate to the registration page. 2. Enter valid name, email, password, NID / Birth certificate number and capture/upload image. 3. Click on “Register”. 	Patient is registered successfully and verification email is sent.	Pending
TC2	Patient Login	<ol style="list-style-type: none"> 1. Navigate to the login page. 2. Enter valid credentials /real-time face detection. 3. Click on “Login”. 	Patient is logged in and redirected to the dashboard.	Pending
TC3	Request Manual / Automated Booking	<ol style="list-style-type: none"> 1. Select a doctor from the list. 2. Click on manual / automated options “Book Now”. 3. Enter preferred date, time and enter physical conditions for automatic diagnosis with booking. 4. Click “Confirm Request”. 	Booking request is sent to the selected Doctor.	Pending
TC4	Proceed to Payment	<ol style="list-style-type: none"> 1. After booking 	Payment loads with	Pending

		request, click “Proceed to Payment”. 2. Select payment method. 3. Enter payment details.	correct booking details.	
TC5	Complete Payment	1. Enter valid payment information. 2. Confirm payment. 3. Wait for transaction to process.	Payment is completed successfully and transaction is recorded.	Pending
TC7	View Transaction Status	1. Navigate to “My Invoice”. 2. Select a transaction to view.	Payment history is displayed with real- time transaction status.	Pending
TC8	View Booking Status	1. Navigate to “My appointments”. 2. Select the recent booking.	Booking status (Pending / Confirmed / Cancelled) is displayed.	Pending
TC9	Doctor Registration	1. Navigate to doctor registration page. 2. Enter name, email, password, specialty, location and put resume pdf. 3. Click “Register”.	Doctor is registered successfully and awaits admin approval.	Pending
TC10	Doctor Login	1. Enter valid credentials/ real-time face detection. 2. Click “Login”.	Doctor is logged in and redirected to their dashboard.	Pending
TC11	View Appointments	1. Go to “My Appointments”. 2. View current and past bookings.	List of appointments is displayed correctly.	Pending

TC12	Mark Completed	<ol style="list-style-type: none"> 1. Click on a pending appointment. 2. Mark “Completed” or “Cancel”. 	Booking status updates and patient is notified.	Pending
TC13	Admin Login	<ol style="list-style-type: none"> 1. Go to admin panel. 2. Enter valid credentials/ real-time face detection. 3. Click “Login”. 	Admin is logged in successfully.	Pending
TC14	Manage Patients	<ol style="list-style-type: none"> 1. Navigate to “Patients List”. 2. View or update patient information. 	Admin can view, edit, or remove patient records.	Pending
TC15	Manage Doctor	<ol style="list-style-type: none"> 1. Navigate to “Doctor List”. 2. Approve or deactivate Doctor profiles. 	Admin successfully manages Doctor accounts.	Pending
TC16	Confirm Payments	<ol style="list-style-type: none"> 1. Go to “Payments” section. 2. Verify payment transaction. 3. Click “Confirm”. 	Payment is verified and marked as completed.	Pending
TC17	Confirm Bookings	<ol style="list-style-type: none"> 1. View pending bookings. 2. Approve confirmed ones. 	Bookings are confirmed and system notifies both parties.	Pending
TC18	Update patient records	<ol style="list-style-type: none"> 1. Go to “Record Settings”. 2. Modify or add new reports. 3. Save changes. 	Updated reports appear immediately on the platform.	Pending

Chapter 12

Implementation

12.1 Scaling:

Hospital Management, the platform can efficiently and affordably handle a growing number of users, doctors, bookings, and data as the system expands. This is achieved through technologies like caching for faster data access, a Content Delivery Network (CDN) to support many users simultaneously, and database optimization to ensure smooth and reliable performance under heavy loads. Cloud infrastructure can play a very important role in the scalability of Hospital Management. In There are two ways the system can be built: by scaling vertically-upgrading the capacity of existing servers. horizontal scaling, adding more servers to distribute the workload. The frontend and backend can easily scale up to sudden increases in traffic. For instance, many users accessing the app simultaneously during health campaigns or emergencies. At the backend, Python Django provides strong support for scalability, including connection pooling and asynchronous request handling, and load balancing enable the platform to handle numerous requests. efficiently. Additionally, PostgreSQL has table partitioning, replication, which also enhances performance. It also supports query optimization to ensure fast and secure handling of large volumes of data. Combining HTML, CSS, Bootstrap, and JavaScript on the frontend helps create a lightweight and responsive interface. These technologies improve scalability by enabling client-side and reduction in the number of requests being transferred to the server for faster loading of the page. and smoother user interactions. Scalability is a requirement of Hospital Management, not just a feature. It ensures that platform can continue to perform well as the number of users, doctors, and bookings increase. This would ensure that the system scaled up smoothly without crashes or slowdowns and remained reliable while seamless healthcare experience for everyone involved.

12.2 Load Balancing:

Load balancing means distribution of incoming traffic or user requests across multiple servers so that no single server gets overloaded. This helps the system to run faster, be reliable, and always be it is available to users. In Hospital Management, load balancing helps in ensuring the platform can handle many users simultaneously. For instance, when many patients try to book doctors during busy hours or health campaigns. A few common ways to do load balancing include round-robin, least associations, and IP hash.

The Round-Robin Method sends requests to each server in turn in a continually repeating cycle.

This way all the servers receive equal amounts of work. The least connections method sends a new request to the server with the fewest active users.

This keeps things balanced and helps every server run efficiently. The IP hash method uses the user's IP address to decide which server should handle their requests. So, if the same user connects again, they'll usually reach the same server, which makes the user's experience quicker and easier. In case one server gets too busy or goes down, the System automatically redirects users to other available servers.

Load balancing helps the Hospital Management stay fast, stable, and dependable despite high traffic. It ensures there are no delays, either for the patients using the app or for the doctors' results.

Mission: Make online healthcare experience simple and reliable for everyone.

Chapter 13

Critical Appraisal and Evaluation

13.1 Objective that could be met:

The main objective of the Hospital Management is to make it easy and convenient for patients to find, book, and connect with best doctors while giving administrators strong tools to manage everything behind the scenes. The app provides a seamless experience to the patients with easy/automatic registration/ login, secure booking and payment options, personal profile management. For administrators, it includes tools to manage doctors, patients, and bookings, as well as real-time reports and analytics to track performance and activities. The app uses HTML, CSS, Bootstrap, and JavaScript for the frontend, Python Django for the backend, and PostgreSQL for the database. The objective of the platform is to be reliable, fast, and user-friendly with features like responsive design, secure login, efficient load balancing, and scalable architecture. The overall aim is to enhance accessibility to healthcare services and to make doctors management easier and more efficient for everyone.

13.2 How much better could have been done:

One of the earmarks of most Hospital Managements today is their online healthcare booking capabilities. We Could upgrade some critical areas to make it more powerful and appealing to find qualified doctors.

Example: Adding AI-based suggestions of doctors will help a patient in finding out those doctors who best match their needs and preferences. Some common multiple languages would make the app easier for people of different regions. Real-time location tracking, intelligent scheduling

In order to make the service more accurate and convenient, from a technical side, shifting from a monolithic System to a microservices architecture could improve performance and reliability, since the app grows.

Finally, the inclusion of advanced data analytics and predictive insights would help administrators understand trends, improve decision-making, and offer better healthcare services. These improvements would make the Hospital Management more flexible, smarter, ready for future growth in digital healthcare.

13.3 Which features could not be touched:

Some features of the Hospital Management are essential and should remain unchanged because they form the backbone of the system and ensure user trust. These include the secure login and the registration process that protects user data and privacy.

The booking and scheduling system must also remain robust and stable; it directly affects the way patients and doctors interact. The Admin Dashboard, which enables administrators to manage doctors, patients, and bookings, should also remain intact as it ensures smooth operations and monitoring. Additional features such as AI-based recommendations or predictive analytics can be included in future versions of this project; the core functionality of booking and managing services should remain untouched.

These core systems are in place to keep the app stable and functional, leaving room for innovative enhancements, such as AI-powered doctor matching, automated health suggestions, or data-driven performance insights to be layered on top later.

Chapter 14

Lesson Learned

14.1 Pre Project – Review – Closing:

Pre-Project Phase

The pre-project phase is about scope of the project, requirement analysis, and possibility of project implement. In other words, feasibility entails collecting doctor information, identifying goals and outlining a specific plan. Everyone affected by its implementation defines the issues that the project may face, the features are agreed, and the resources are assigned. Pre-feasibility (technical and operational) and post feasibility (financial) studies are conducted to check the project feasibility. There is also formation of a clear chart and time line that helps in the development process to take place.

Review Phase

This is routine checking as well as approval of the project at some intervals to ensure that it meets the laid down objectives or not. There are periodical progress meetings, unit, integration and system tests, and feedback from stakeholders. During this phase developers make sure that all sub-systems work rightly, all the laid down user interface is right and its back bone systems are efficient. Any questions or variances that come up, are resolved so the project does not experience any delay.

Closing Phase

The closing phase is a process in which the product is completed and a project is final, and deliver to the stakeholders. This is undertaken through the installation of the platform in a live environment being tested and confirmed if all features are working properly and offering necessary user and administrator training. For purpose of determining if the project was successful, and to conduct an assessment and get feedback of the project and the lessons learnt, the post project review is conducted. Preventive and support strategies are also developed to make certain that the system continues to be dependable and current when actually implemented.

14.2 The Problem I Have Faced:

Several challenges appeared during the development of the **Hospital Management**, especially when trying to make the platform smooth, fast, and reliable for patients and doctors. One major issue was handling many users simultaneously for example during busy hours or when many the patients were booking doctors at once. Maintaining this traffic without slowing down the System was a big concern. Another challenge was to keep the system fast and at low cost. High performance might easily rack up server costs, so striking a balance with caching, load balancing, and proper Handling requests took meticulous planning. Another tricky part was the implementation of secure online payments, as it required very strict data protection. and adherence to security standards. In addition, services such as real-time analytics, booking Tracking and AI-based doctor recommendations required complex database operations, which initially were making the app slower.

On the front-end side, ensuring the website looked and worked perfectly on These ranges of devices-from phones to desktops-required repeated testing and UI adjustments the other big challenge was to keep a balance between the patient side and the admin side. Both needed completely different features and interfaces but had to work together without any conflicts. Lastly, keeping the frontend, backend, and database teams in sync was sometimes tricky. Version mismatches and update conflicts slowed down the process of development, especially when several developers were working at the same time.

14.3 What Solutions Occurred:

In order to solve these challenges, numerous improvements were made during the development of the Hospital management; to handle high traffic, load balancer tools such as Nginx and cloud Load balancers were utilized to evenly distribute network requests, so that when many where users were active.

The Python Django-based backend has been optimized using asynchronous tasks, and Redis caching were employed to accelerate data processing and reduce the pressure on servers. Indexing in PostgreSQL and query optimization were applied to improve database performance. and make real-time analytics and search features faster.

This ensured smooth performance even When dealing with a lot of bookings or doctor profiles. HTML, CSS, Bootstrap, and JavaScript testing was performed on the frontend side using tools like Browser Stack to ensure the interface worked well on all screen sizes. Based on test feedback, the Design was refined for enhancing usability and loading speed. For better organization, the admin and user modules were developed separately to avoid code. conflicts and make updates easier. Clear version control practices and better team coordination were also introduced to ensure smooth communication between frontend, backend, and database teams.

Overall, these solutions helped make the **Hospital Management** more reliable, faster, and easier to manage both for users and administrators.

Chapter 15

Lesson Learned

15.1 Summary of the project:

Hospital Management project is a concept that started as an online site where patients want to find best qualified doctors in short time while providing adequate administrative tools that will allow for easy management of the site. Using JavaScript for the frontend, and Python Django for backend, supplemented by HTML, CSS, Bootstrap and PostgreSQL database this application includes such intuitive UI options as account creation, details information of doctor viewing, , booking for doctor, login or viewing records of patients in real time and personal profiles. The administrators have a privileged access that covers the management of patients, doctors, booking, payment transaction all in one easy to use dashboard. Verification and searching of qualified doctors, analytics, and scalability combine reliably with challenges for booking enhanced competitive online healthcare platform.

15.2 Goal of the project:

Hospital Management is the created project aimed to minimize the cost, sufferings and save time for patients especially who are incapacitated. Also making the **Hospital Management** more convenient on online platform for patients and effective for admin and management. It is supposed to offer convenient find doctor, booking, as well as customizable user tools like personal account and booking status indication. For administrators it can potentially provide convenient means of managing doctors, booking, messaging, confirming payment transaction, reviewing, rating and analytics. Through the adaptive features including integrated added technologies, secure booking, messaging, and responsive design, the project aims to assist healthcare development and bring value to its patients.

15.3 What I have done in Documentation:

This document of **Hospital Management** project includes every stage of the platform development from the planning point to the deployment point. I described the aims and characteristics of the project and provided the features for the patients, doctors and administrator interfaces. These are HTML, CSS, JavaScript, Python, PostgreSQL and using frameworks like Django on which I explained in detail on the parts they played in the creation of the system. Also, I prepared patient, doctor and admin documentation setting out the registration, login process, payment, and booking doctor management. I also created technical specifications, activity diagram, class diagram, and test cases to make it easier to come back to and modify. The documentation is intended for further evolution of the system, as well as for testing and staff training.

My Experience:

While working on the **Hospital Management** project, I learned about the peculiarities of web development and the management of the project. I was writing both frontend and backend code using JavaScript, Python, PostgreSQL and using frameworks like Django. I wrote interfaces and served as an interface between frontend and backend, created dynamic plots generated in respond to user interaction.

I also did some architectural tuning of the platform to make it run faster and to support the planned growth of user numbers through horizontal scaling techniques and by improving the database performance. Also, I work in cooperation with my colleagues, adhere to the process of the regular meetings' conducting, and take an active part in documentation. I was able to pick improvements in my problem-solving skills especially when it comes to understanding and handling numerous demands in different system aspects.

All in all, the project consolidated the knowledge of creating full-stack applications and improve the skills in overcoming problems in technical as well as in a team setting.

Works Cited

1. *Practo. "Book Doctor Appointments and Online Consultations." Practo, www.practo.com.*
2. *HealthTap. "Talk to a Doctor Online, 24/7." HealthTap, www.healthtap.com.*
3. *Zocdoc. "Find a Doctor, Book a Visit Online." Zocdoc, www.zocdoc.com.*
4. *Care.com. "Find Caregivers, Babysitters, and Doctors Near You." Care.com, www.care.com.*
5. *Home Instead. "In-Home Care for Seniors and Families." Home Instead, www.homeinstead.com.*
6. *Doctor Next Door. "Home Health Care Services." Doctor Next Door, www.doctornextdoor.com.*
7. *Carenow. "Urgent Care Clinics and Online Check-In." CareNow, www.carenow.com.*
8. *Healthline. "Telehealth and Virtual Care Explained." Healthline, www.healthline.com.*
9. *BetterHelp. "Online Counseling and Therapy Platform." BetterHelp, www.betterhelp.com.*
10. *Doctolib. "Book Appointments with Healthcare Professionals." Doctolib, www.doctolib.com.*
11. *Amwell. "Online Doctor Visits and Telemedicine Services." Amwell, www.amwell.com.*
12. *PlushCare. "Talk to a Doctor Anytime, Anywhere." PlushCare, www.plushcare.com.*
13. *MyDoctor. "Connecting Patients with Qualified Doctors." MyDoctor, www.mydoctor.ai.*
14. *Healthgrades. "Find Doctors and Hospitals Near You." Healthgrades, www.healthgrades.com.*
15. *WebMD. "Health Information and Online Medical Resources." WebMD, www.webmd.com.*

213-16-584

ORIGINALITY REPORT

15%	14%	1%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	10%
2	Submitted to Daffodil International University Student Paper	4%
3	Submitted to University of Greenwich Student Paper	<1%
4	Submitted to The University of Texas at Arlington Student Paper	<1%
5	Submitted to Institute of Technology, Nirma University Student Paper	<1%
6	Submitted to Queen's University of Belfast Student Paper	<1%
7	humber.ca Internet Source	<1%
8	www.coursehero.com Internet Source	<1%
9	Submitted to Aliat Universidades Student Paper	<1%
10	Hilman Jihadi, Dade Maulana, Harjo Baskoro. "Construction And Design of A Web Information System For Extracurricular Activities In High Schools In Bogor City-Indonesia", Digitus : Journal of Computer Science Applications, 2024 Publication	<1%

11	Submitted to Te Pūkenga trading as the Open Polytechnic Student Paper	<1%
----	--	-----