



ULAB

UNIVERSITY OF LIBERAL ARTS
BANGLADESH

Final Report

Submitted to:

Suravi Akhter

Lecturer

Department of CSE

University of Liberal Arts Bangladesh

Submitted By:

Name: Sami Hossain

ID: 223014146

Name: Abu Saleh MD Araf

ID: 231014047

Name: Fahim Shihab Touhid

ID: 223014025

Name: Md Hafiz Ullah

ID: 223014199

Course Title: Design Project 02

Course Code: CSE 3200

Section: 03

Date of Submission: 28 August, 2025

Table of Contents

Table of Contents -----	2
Introduction: -----	3
Software Requirement Collection : -----	4
Questionnaire :-----	4
Patient Experience Survey Questions:-----	4
Doctor Experience Survey Questions:-----	7
Management Experience Survey Questions:-----	9
Literature Review: -----	11
Proposed Solution: -----	16
Software Requirement Specification (SRS): -----	18
Functional Requirements:-----	18
Admin Role:-----	18
Doctor Role:-----	19
Patient Role:-----	19
Non-Functional Requirements:-----	19
Use Case Diagram -----	22
Activity Diagram -----	23
ER Diagram -----	26
Data flow Diagram -----	28
Data Flow Diagram level 0:-----	28
Data Flow Diagram level 1:-----	29
Class Diagram -----	30
Class Diagram Part 01:-----	30
Class Diagram Part 02:-----	31
Sequence Diagram -----	33
Sequence Diagram Part 01:-----	33
Sequence Diagram Part 02:-----	35
Methodology: -----	35
Project Management: -----	41
Detailed Project Management Plan: -----	41
Contribution to Society: -----	45
Conclusion: -----	47
Reference: -----	48
Website Prototype: -----	49
Code in Appendix: -----	61
Contribution Table: -----	83

Introduction:

In today's fast-paced and digitized world, the healthcare sector in developing countries faces numerous challenges due to outdated and fragmented management systems. Hospitals, especially in regions like Bangladesh, often suffer from inefficiencies such as long patient wait times, lost medical records, lack of communication between departments, and manual data entry errors. These inefficiencies not only reduce the quality of healthcare services but also increase the workload on hospital staff and compromise patient safety. The COVID-19 pandemic further highlighted these gaps, showing how manual systems can collapse under increased demand.

To address these issues, the development of a Hospital Management System (HMS) has become essential. The HMS is a web-based application designed to streamline and digitize the hospital's internal operations such as appointment scheduling, patient record keeping, prescription management, and staff coordination. The primary goal of the system is to offer a centralized platform that enhances the efficiency, transparency, and accessibility of healthcare services.

Built using open-source technologies like PHP, MySQL, HTML, CSS, and JavaScript, the system offers robust functionality at a low cost, making it ideal for hospitals in developing regions. The HMS provides different portals for Admins, Doctors, and Patients—each with role-specific features to support their responsibilities.

Would it be better if you could see which doctors are available before going to the hospital? *

- Yes
- No

Is there any part of the hospital service or system that feels confusing or hard to understand? *

Long answer text

How easy is it for you to find the right type of doctor (specialist) near your area? *

- Very Rare
- Rare
- Easy
- Very Easy

What are 3 main problems you face when getting treatment? (Ex : waiting time, unclear communication) *

Long answer text

Doctor Experience Survey Questions

1. How many patients do you see daily? What digital tools can help you manage better in less time?--- (*Comment Section*)
2. Do you often miss patient history or old reports? Will a digital record system help your treatment?--- (*Comment Section*)
3. If patients got reminder messages or calls, would it help reduce missed appointments? Would a follow-up message system help you?
4. Is it hard to work smoothly with lab staff or get test results on time?
5. Without a digital system, is it difficult to track how the patient's treatment is progressing over time?

Reimagining Online Hospital Management in Bangladesh: Doctor Experience Survey



We're building a better online hospital management system for Bangladesh.
To do that, we want to hear from patients, doctors, and hospital staff about what works and what doesn't.

Your opinion matters.
This survey takes less than 5 minutes.

Let's improve healthcare together!

Email *

Valid email

This form is collecting emails. [Change settings](#)

How many patients do you see daily? What digital tools can help you manage better in less time? *

Long answer text

Do you often miss patient history or old reports? Will a digital record system help your treatment? *

Long answer text

If patients got reminder messages or calls, would it help reduce missed appointments? Would * a follow-up message system help you?

- Yes
- No
- Maybe

Is it hard to work smoothly with lab staff or get test results on time? *

- Very Important
- Sometimes
- Neutral
- Not Important at all

Without a digital system, is it difficult to track how the patient's treatment is going over time? *

- Yes
- No
- Neutral
- Not need at all

Management Experience Survey Questions

1. What are the biggest problems in running hospital services? Which tech tools can help fix them?--- (*Comment Section*)
2. Is it easy to track hospital income and expenses? What reporting system would make this clearer?--- (*Comment Section*)
3. Is it hard to monitor staff attendance and work? Will automatic systems help manage better?
4. How do you check patient satisfaction and service quality now? What tech can help you track it better?--- (*Comment Section*)
5. Would a real-time dashboard help in decision-making?
6. If the number of patients increases or new services are added, can your current system handle it easily?--- (*Comment Section*)

Reimagining Online Hospital Management in Bangladesh: Management Experience Survey

We're building a better online hospital management system for Bangladesh. To do that, we want to hear from patients, doctors, and hospital staff about what works and what doesn't.

Your opinion matters—even if you've just visited a hospital once. This survey takes less than 5 minutes.

Let's improve healthcare together!

What are the biggest problems in running hospital services? Which tech tools can help fix them? *

Long answer text

Is it easy to track hospital income and expenses? What reporting system would make this clearer? *

Long answer text

Is it hard to monitor staff attendance and work? Will automatic systems help manage better? *

1. Yes

2. No

3. Neutral

By digitizing the entire operational workflow, the HMS significantly reduces the dependency on paper-based records, increases accountability, and ensures that critical health data is always available when needed. Moreover, it is designed to be user-friendly, secure, scalable, and maintainable, ensuring long-term value to healthcare institutions of any size.

Software Requirement Collection:

Questionnaire

To better understand the needs, challenges, and expectations related to hospital services, we conducted a stakeholder survey targeting three key groups: Patients, Doctors, and Hospital Authorities. The aim of this survey was to collect feedback that would guide the design and development of an improved hospital management system. Below is the full list of questions categorized by stakeholder type.

Patient Experience Survey Questions

1. Are you happy with how you book appointments at your hospital? What can make it easier for you?--- (*Comment Section*)
2. How important is it for you to see your reports, prescriptions, and medical history on your phone or online?
3. Would you be happier if you could see which doctors are available online before going to the hospital?
4. Is there any part of the hospital service or system that feels confusing or hard to understand?--- (*Comment Section*)
5. How easy is it for you to find the right type of doctor (specialist) near your area?
6. What are the 3 main problems you face when getting treatment? How can technology (like apps or online help) solve them?--- (*Comment Section*)

Reimagining Online Hospital Management in Bangladesh: Patient Experience Survey

B I U ↵ X

We're building a better online hospital management system for Bangladesh. To do that, we want to hear from patients, doctors, and hospital staff about what works and what doesn't.

Your opinion matters—even if you've just visited a hospital once. This survey takes less than 5 minutes.

Let's improve healthcare together!

Email *

Valid email

This form is collecting emails. [Change settings](#)

Are you happy with how you book appointments at your hospital? What can make it easier for * you?

Long answer text

How important is it for you to see your reports, prescriptions, and history on your phone or online? *

- Very important
- Somewhere important
- Neutral
- Not very important
- Not important at all

How do you check patient satisfaction and service quality now? What tech can help you track it better? *

Long answer text

Would a real time dashboard help in decision-making? *

Yes

No

If the number of patients increases or new services are added, can your current system handle it easily? Please share your thoughts. *

Long answer text

We collected responses ranging from 9 to 11 participants across questions aimed at understanding the perspectives of doctors, patients, and hospital authorities regarding the implementation of digital systems in healthcare.

Literature Review:

This research, focusing on the increasing global adoption of Electronic Health Records (EHR), proposes a secure, web-based Clinical Laboratory Management System designed by the system improves the archiving and retrieval of patient test results by restricting access to authorized personnel such as lab assistants and physicians. Developed using MySQL, PHP, HTML, CSS, and JavaScript, the system aims to enhance accuracy, speed, and reliability in clinical operations, reducing manual workload and minimizing errors in medical documentation.[1]

Developed using PHP, MySQL, Bootstrap, and Ajax technologies, the Clinic Management System described by offers a comprehensive web application to streamline healthcare clinic workflows. It provides robust user management for administrators, facilitates appointment scheduling for doctors and patients, and enables secure medical record access. By improving communication and operational efficiency, the system enhances patient satisfaction and quality of care through a user-friendly interface.[2]

Present a modular, web-based PACS that addresses the need for efficient management of medical imaging such as X-rays, CT scans, and MRIs in hospital environments. The system integrates with Hospital Information Systems (HIS) and Radiology Information Systems (RIS) to provide electronic requesting, delivery, storage, and retrieval of images and patient data. Utilizing open-source platforms like Apache, MySQL, and PHP, the design emphasizes security through user authentication and flexibility via pluggable modules, allowing hospitals to tailor and expand functionality as needed.[3]

Aimed at overcoming inefficiencies at Anver Ismail Memorial Hospital, this project by introduces a web-based Hospital Management System that automates critical processes including patient registration, ward admission, discharge, transfers, and investigations. The system is intended to replace the hospital's entirely manual operations with an integrated digital platform to enhance staff productivity and patient care efficiency.[4]

In response to persistent criticisms of Nigeria's healthcare service delivery, developed a secured web-based platform to manage medical records across healthcare providers. The system incorporates strong authentication and authorization techniques to safeguard patient privacy and data security. Developed using WAMP Server with HTML, CSS, and MySQL, it aims to restore public confidence, promote digital healthcare adoption, and improve service accessibility and quality across Nigeria's public health sector.[5]

This project, by delivers a comprehensive web-based platform that addresses the core needs of clinics: user management, appointment scheduling, and medical record handling. Using PHP and MySQL for backend development, the system empowers administrators, doctors, and patients with easy access to essential functionalities. The intuitive user interface facilitates smooth workflows, better communication, and improved healthcare delivery.[6]

This research outlines the development of an E-Medical Management platform designed to replace traditional paper prescriptions with a digital solution. The system supports patient registration, appointment booking, and provides doctors and management with instant access to medical records. Its web-based nature allows for ubiquitous, cost-effective access, enhancing communication between patients, doctors, and hospital staff, and increasing overall healthcare efficiency.[7]

The system features an online hospital management platform that facilitates remote appointment booking and doctor information access for patients. Additionally, it provides hospital administrators with a complete admin dashboard to oversee day-to-day operations. This centralized control improves operational workflow, reduces patient wait times, and ensures better resource management within the hospital.[8]

This paper, by introduces a web-based real-time monitoring system for hospital medical equipment, focusing on maximizing equipment utilization and minimizing unnecessary purchases and idle time. By tracking usage and income data, hospital administrators can make informed decisions to increase revenues and reduce operational costs, thereby improving the overall financial and functional efficiency of healthcare facilities.[9]

Asher Fawwad, Mansoor Ahmed Siddiqui, and Amna Mansoor (2024) provide a comprehensive overview of web-based Hospital Management Systems (HMS) in developing nations. Their commentary discusses the significant benefits such as real-time information access, telemedicine capabilities, and system integration. They also highlight challenges like infrastructure deficits, high initial costs, security concerns, and the necessity for extensive user training, emphasizing that overcoming these barriers is essential to revolutionize healthcare delivery in resource-limited settings.[10]

This project, by seeks to automate essential hospital workflows including patient registration, appointment scheduling, ward allocation, and medical investigations through a comprehensive web-based system. By digitizing these processes, it aims to eliminate manual inefficiencies, reduce errors, and improve the overall quality and efficiency of healthcare service delivery, thus benefiting both hospital staff and patients.[11]

Paper's Title	Author's Name	Contribution	Research Gap
Web-Based Hospital Management System	Ismaeel A. Sikiru ¹ , Rafiat A. Oyekunle ²	This research designed to address the inefficiencies in hospitals operating with manual or semi-automated systems, the proposed HMS enhances communication among stakeholders, improves service quality, and ensures better time and resource management.	However, to the best of our knowledge, most of the hospitals found in our survey territory, Ilorin metropolis of Kwara state, Nigeria, are either operating HMS manually or partly automated. This deployment has posed a lot of challenges to this sector.
Hospital Management System using Web Technology	Jharana Paikray ¹ , Laxmikanta Sahoo ² , Krishna Kumar Patra ³ , Prasangsita Khara ⁴ , Priyanka Shit ⁵	This paper proposes a web-based E-Medical Management System to digitize hospital processes, replacing paper prescriptions with centralized digital records.	Some basic algorithms like ID3 have been used for the allotment of doctors and prediction of disease. Further some complex algorithms can be used to improve the performance of the system.
Development of Web-Based Clinic Management System	Nur Syafinaz Zamri ¹ , Yana Mazwin Mohmad Hassim ^{1*}	enhances efficiency for both patients and clinic staff by enabling easy scheduling, data retrieval, and report generation.	Absence of a web-based system at Medi Ehsan clinic, causing difficulties in appointment scheduling, data management, and patient record accessibility.

<p>A Web-Based Information System Applied on Utilization/Benefit Management of Medical Equipment in Hospital.</p>	<p>Chia-Hung Chien¹ , Man-Hsiang Chang² and Yi-You Huang¹</p>	<p>a web-based information system for real-time monitoring of medical equipment usage, aimed at maximizing utility, increasing revenue, and reducing unnecessary costs through data-driven management.</p>	<p>lack of real-time monitoring systems in hospitals to evaluate the utilization, effectiveness, and cost-efficiency of medical equipment.</p>
<p>Web Based Hospital Management System</p>	<p>Rahman, Asifur</p>	<p>presents an Online Hospital Management System that enables patients to book appointments and access doctor information remotely, while providing administrators with a complete dashboard to manage hospital operations efficiently.</p>	<p>lack of integrated online systems that allow patients to conveniently book appointments and enable hospital management to efficiently oversee operations through a centralized platform.</p> <p>Ask ChatGPT</p>
<p>Hospital management system using web technology is a demand of time</p>	<p>Mansoor Ahmed ,Siddiqui Asher ,Fawwad Amna Mansoor</p>	<p>highlights the urgent need for adopting web-based HMS in healthcare facilities, emphasizing the benefits of real-time data access, system integration, telemedicine, and improved patient care</p>	<p>many hospitals in developing countries still rely heavily on inefficient, error-prone paper-based processes due to inadequate infrastructure, lack of trained personnel which hindering effective healthcare delivery.</p>

<p>A WEB BASED APPLICATION FOR CLINICAL LABORATORY INFORMATION MANAGEMENT SYSTEM</p>	<p>Zinah Jaffar Mohamed Ameen1, Sama Salam Samaan2</p>	<p>contributes a secure, web-based Clinical Laboratory Management System designed to streamline the archiving and retrieval of patient records and test results</p>	<p>Existing clinical laboratory processes are slow, manual, and prone to errors, lacking a secure, web-based system that ensures fast, accurate, and authorized access to patient records and test results.</p>
<p>Design and Development of a Web Application for Medical Clinic Management</p>	<p>BOUSSAID Alaeddine ,HAFFERSSAS Aymen</p>	<p>Presents a comprehensive web-based Clinic Management System that integrates user management, appointment scheduling, and secure medical record handling into a single, user-friendly platform,</p>	<p>Many healthcare clinics still rely on fragmented or manual systems for managing users, appointments, and medical records, leading to inefficiencies, errors, and poor coordination between staff and patients.</p>
<p>Implementation of Web-based Management System for Pacs Network</p>	<p>Nasser N. Khmiss 1 , Nehad Hameed Hussein 2</p>	<p>A modular PACS management system that integrates with hospital information systems, supports secure user access, and allows easy addition of new features, enhancing flexibility, reliability, and workflow efficiency.</p>	<p>Existing hospital systems often lack a flexible, integrated, and secure Picture Archiving and Communication System (PACS) that efficiently manages medical images and related data with modular expandability.</p>

WEB BASED HOSPITAL MANAGEMENT SYSTEM FOR ANVER ISMAIL MEMORIAL HOSPITAL SAMMANTHURAI	FARSANA. LTI	Proposes a comprehensive web-based Hospital Management System to automate key hospital processes such as patient registration, ward allocation, discharge, transfer, and investigations, improving operational efficiency and staff productivity.	Relies on fully manual processes, leading to inefficiencies and delays in patient registration, ward management, and other hospital operations.
A WEB-BASED MEDICAL RECORDS MANAGEMENT SYSTEM FOR NIGERIA	USMAN, Nuradeen Umar	Develops a secured web-based platform for managing medical records with robust authentication and authorization, enhancing privacy and security, and demonstrates user readiness to adopt digital solutions	Nigeria's healthcare system suffers from poor service quality and lack of secure, unified medical record management, leading to loss of patient confidence and increased medical tourism.

Proposed Solution:

To solve the challenges of inefficient healthcare administration, this project proposes a fully functional Hospital Management System that centralizes and automates hospital operations. The system is web-based and accessible via standard internet browsers, making it usable on computers and mobile devices alike.

Key Objectives:

1. Streamline Hospital Workflow: Automate routine processes like appointments, patient admissions, discharge, and medical history updates.

2. Ensure Data Security and Accuracy: Eliminate human errors in record-keeping through validated inputs and secure storage.
3. Role-Based Access: Provide separate dashboards for Admin, Doctor, and Patient, each with relevant functionalities.
4. Real-Time Updates: Enable instant access to patient information, appointment schedules, and prescriptions.
5. Improve Patient Satisfaction: Reduce wait times and enhance the accuracy of diagnosis and treatment through better information flow.

System Features:

Admin Panel

- Register and manage doctors and departments.
- Monitor appointments and system users.
- View logs and generate reports.
- Maintain system security and data backups.

Doctor Panel

- View upcoming appointments.
- Access patient history and medical records.
- Write and update digital prescriptions.
- Communicate with the administrative team.

Patient Portal

- Register and manage profiles securely.
- Book appointments with preferred doctors.
- View past prescriptions and treatment history.
- Receive system notifications and alerts.

Benefits:

- Transparency: All actions are logged and auditable.
- Efficiency: Eliminates paper-based redundancy.
- Scalability: New modules can be added as needed (e.g., pharmacy, billing).
- Accessibility: Accessible from any device with an internet connection.
- Inclusiveness: Can be adapted to multilingual and offline SMS notifications in the future.

This proposed HMS ensures that hospital operations are conducted in a controlled, secure, and efficient manner while maintaining the integrity and confidentiality of patient data.

Software Requirement Specification (SRS):

1. Functional Requirements

Admin Role:

- Login/logout with secure credentials.
- Add/edit/delete doctors and departments.
- Manage patient records.
- Approve/reject appointment requests.
- Generate usage and appointment reports.

Doctor Role:

- Login securely to their portal.
- View appointments and patient records.
- Enter diagnoses and write prescriptions.
- Update patient treatment status.

Patient Role:

- Register/login with secure credentials.
- Book appointments with available doctors.
- View medical history and prescriptions.
- Update contact or profile information.

2. Non-Functional Requirements

- **Security:**

- Role-based access control.
- Passwords hashed before storage.
- Secure login sessions.

- **Performance:**

- Fast data retrieval with optimized SQL queries.
- Capable of handling hundreds of users concurrently.

- **Usability:**

- Intuitive UI for non-technical users.
- Responsive layout for mobile and tablet users.

- **Maintainability:**

- Modular code written in PHP.
- Easily extensible with clear file structure.

- **Scalability:**

- New roles or modules (e.g., pharmacy, blood bank) can be added.

- **Backup & Recovery:**

- Data stored in MySQL database with regular backup options.

3. Technical Specifications

- Frontend: HTML5, CSS3, JavaScript, Bootstrap
- Backend: PHP
- Database: MySQL
- Server: Apache 2.4 (XAMPP)
- OS Compatibility: Windows
- Browser Compatibility: Chrome, Firefox, Edge

4. System Architecture

- **Three-Tier Architecture:**
 - Presentation Layer (UI/UX)
 - Application Logic Layer (PHP scripts and logic)
 - Data Layer (MySQL database)
- **Database Schema (Core Tables):**
 - users (id, name, email, password, role)
 - patients (id, user_id, dob, blood_group, history)
 - doctors (id, user_id, department, specialization)

- appointments (id, patient_id, doctor_id, date, status)
- prescriptions (id, appointment_id, description, issued_by)

Use Case Diagram:

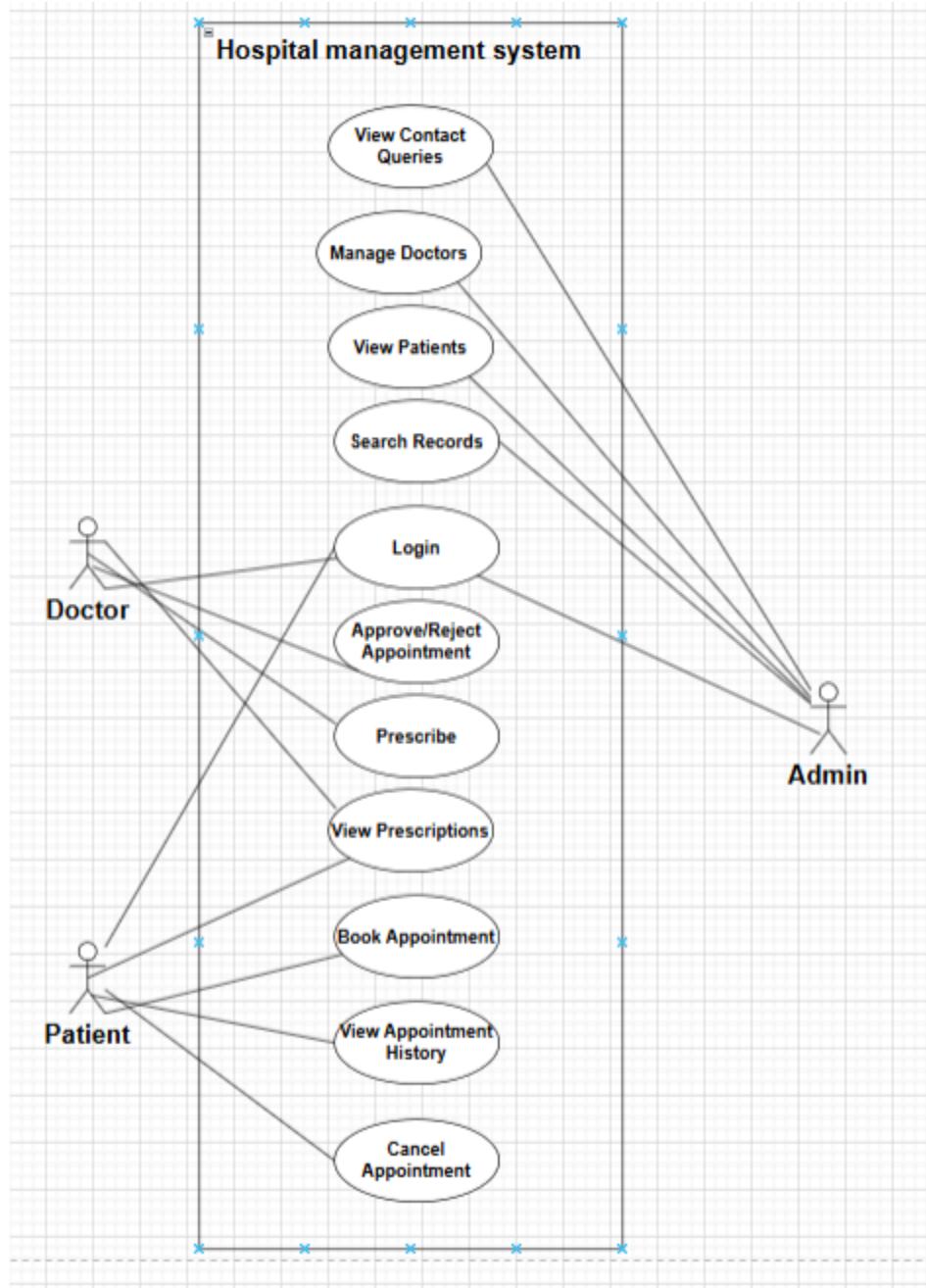


Figure- Use Case Diagram of Hospital Management System

This use case diagram illustrates the interactions between the primary actors—Admin, Doctor, and Patient—and the hospital management system. The Admin is responsible for managing doctors, viewing contact queries, searching records, and overseeing patient data. Doctors can log in, approve or reject appointments, prescribe medications, and view prescription histories. Patients can book or cancel appointments, view appointment history, and log in to the system. This diagram helps visualize the system's functional requirements and user roles in a structured way.

Activity Diagram:

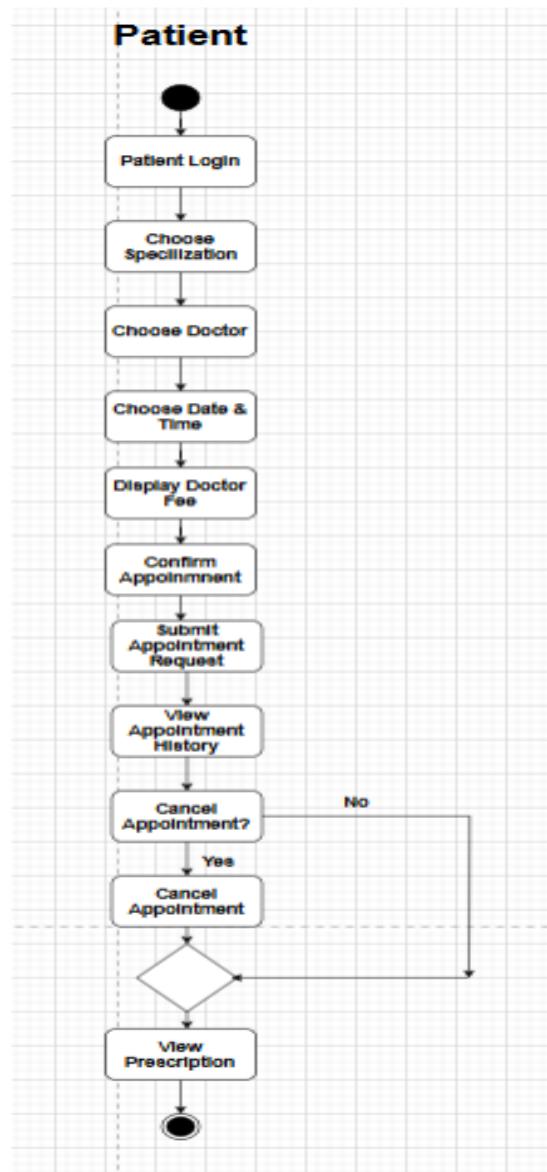


Figure-Activity Diagram for Patient

This activity diagram shows the sequence of actions taken by a **Patient** in the hospital management system. The process begins with patient login, followed by selecting a medical specialization and choosing a specific doctor. The patient then selects a date and time for the appointment, views the doctor's fee, and confirms the appointment.

After submitting the appointment request, the patient can view their appointment history. The diagram also includes a decision point to cancel an appointment, after which the patient can proceed to view the doctor's prescription. This workflow ensures an organized and user-friendly booking and consultation process for the patient.

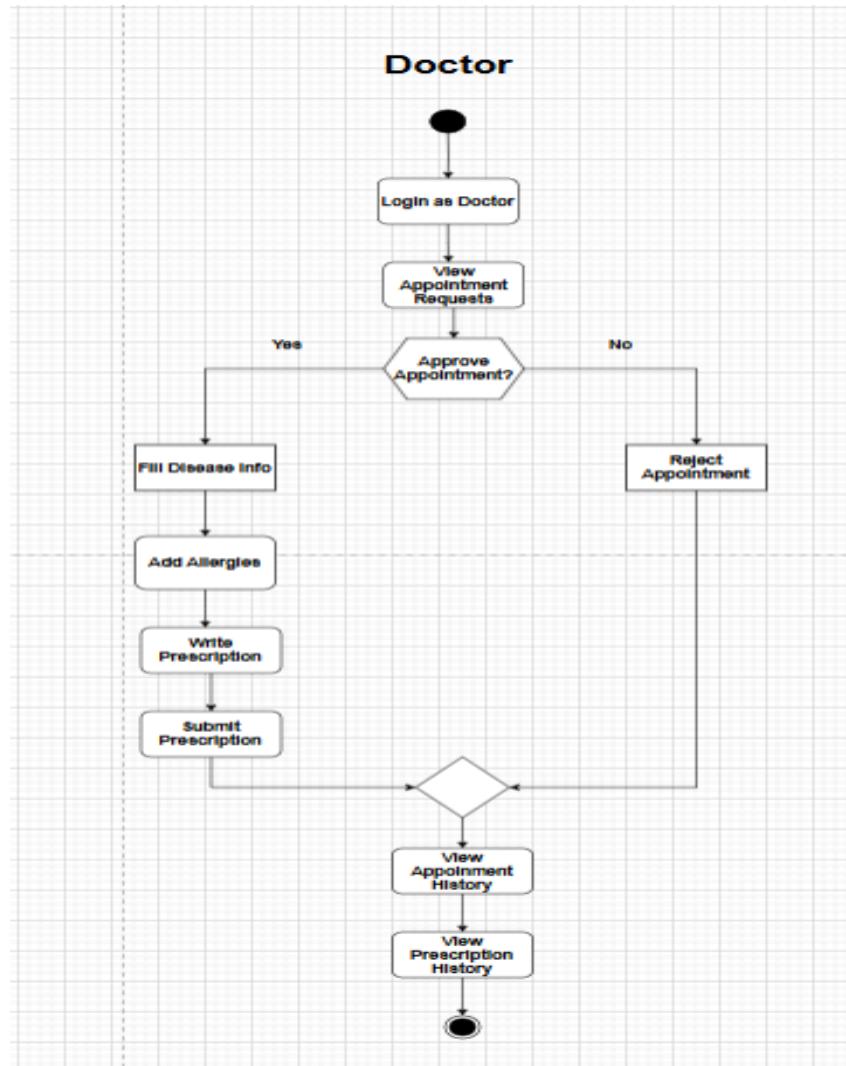


Figure-Activity diagram for Doctor

This activity diagram outlines the steps a **Doctor** follows after logging into the system. The doctor first views incoming appointment requests. Based on availability or other factors, the doctor can approve or reject the request. If approved, the doctor proceeds to fill in the patient's disease details, add any allergies, and write a prescription.

Once the prescription is submitted, the doctor can view both the appointment and prescription histories. This workflow supports an efficient and structured consultation and documentation process for healthcare providers.

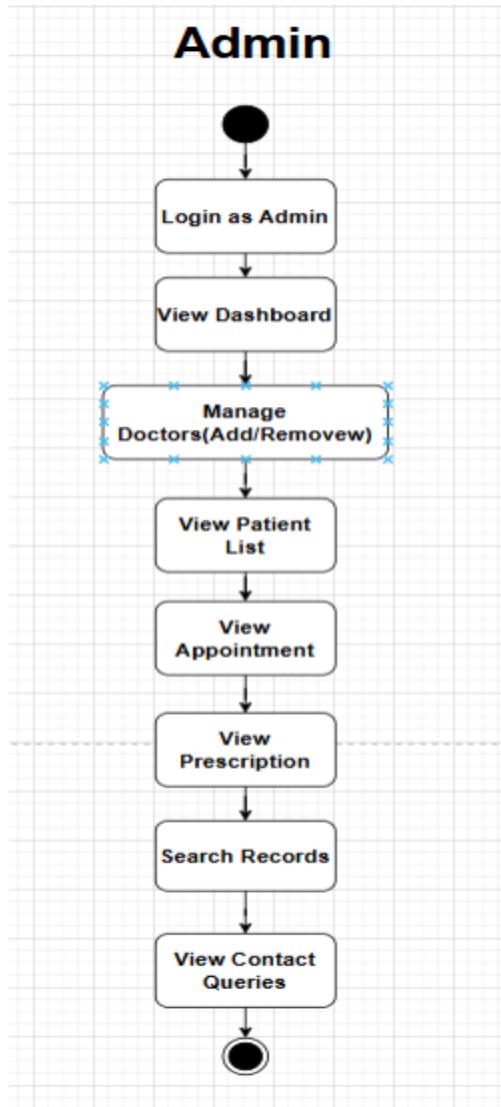
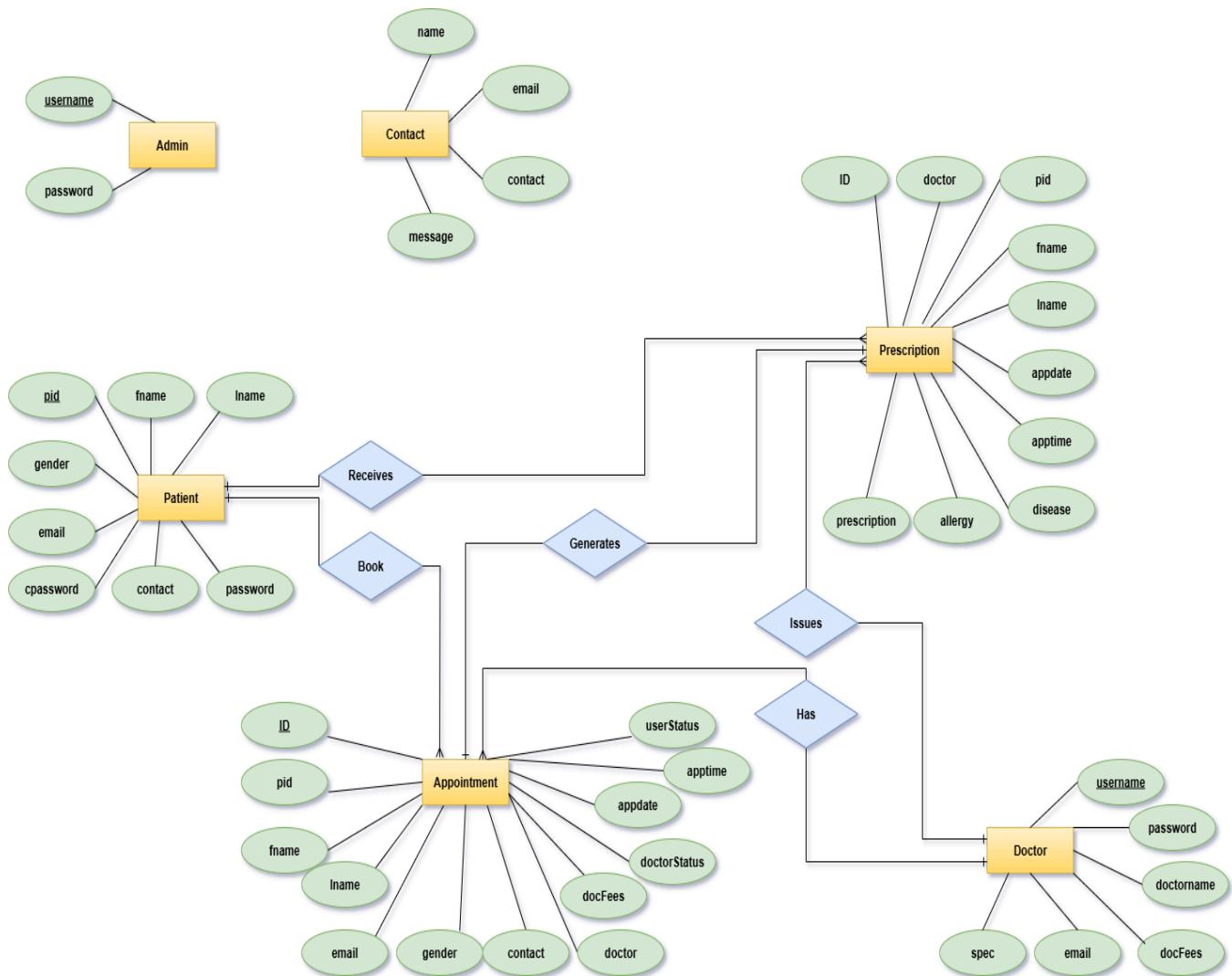


Figure-Activity Diagram for Admin

This activity diagram outlines the operational flow for the **Admin** in the hospital management system. The process begins with the admin logging into the system and

accessing the dashboard. From there, the admin can manage doctor information by adding or removing profiles, view the complete list of patients, check all appointment records, and review submitted prescriptions. Additionally, the admin can search specific records and respond to contact queries from users. The process ends when the admin logs out or exits the system. This workflow ensures that the admin maintains control and oversight over all key system functions.

ER Diagram:

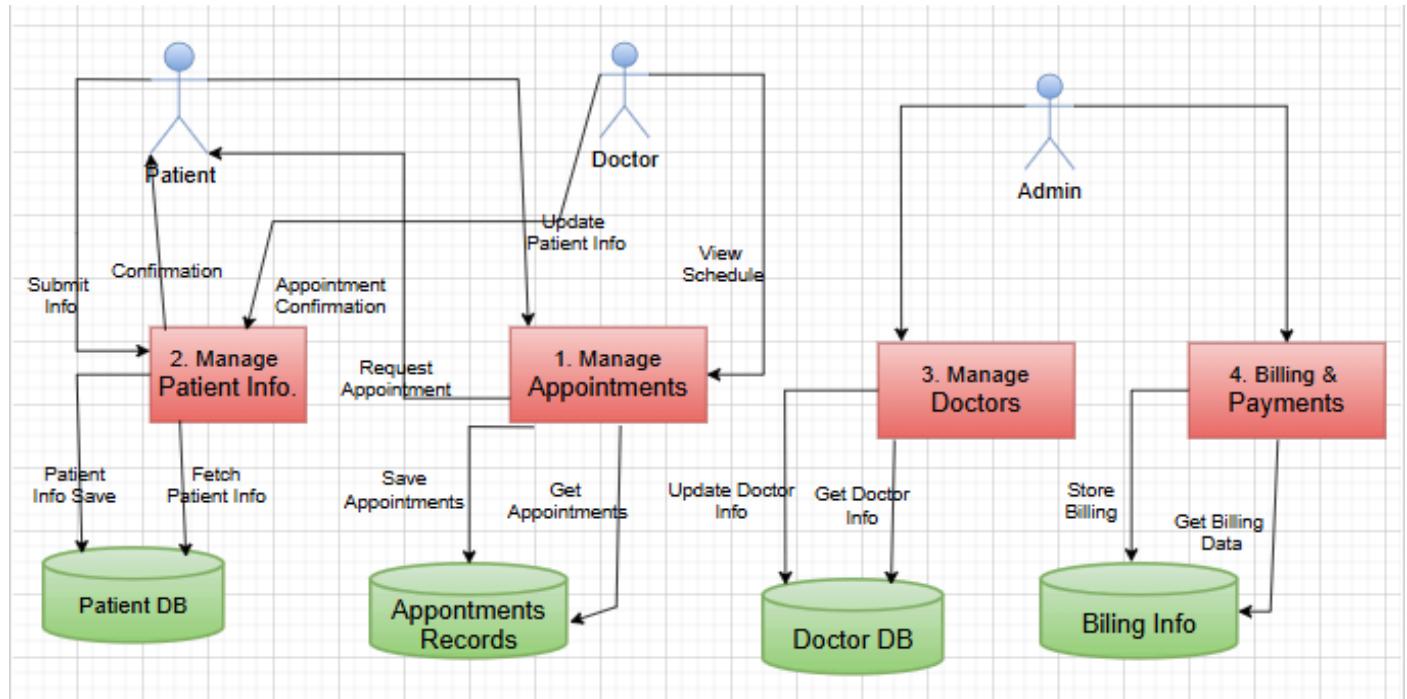


This ER diagram represents the core data structure of our Hospital Management System, showing entities, their attributes, and the relationships between them. The primary entities include Admin, Contact, Patient, Appointment, Prescription, and Doctor.

The Admin entity stores administrator credentials such as username and password, allowing secure access to system management functions. The Contact entity captures inquiries or messages from users, containing fields for name, email, contact number, and message content. The Patient entity holds essential patient information, including patient ID, first and last name, gender, email, contact number, and both chosen and confirmed passwords. Patients can perform two key actions: they receive appointments and book them. The Appointment entity records details such as appointment ID, patient details, date, time, doctor assigned, doctor's fee, and status fields for both the patient and doctor. Appointments can Generate prescriptions. The Prescription entity stores data such as prescription ID, doctor's name, patient details, appointment date and time, diagnosed disease, prescription notes, and any allergies. Appointments also issue prescriptions to patients. The Doctor entity contains credentials and professional information, including username, password, doctor name, specialization, email, and consultation fees. Each appointment has an associated doctor, linking patient care to the appropriate medical professional.

This ER model ensures clear representation of all entities, their relevant attributes, and the relationships that define hospital operations. The use of relationships like Receives, Book, Generates, Issues, and Has makes the workflow of data movement and linkage between entities explicit, enabling structured database design and smooth hospital system management.

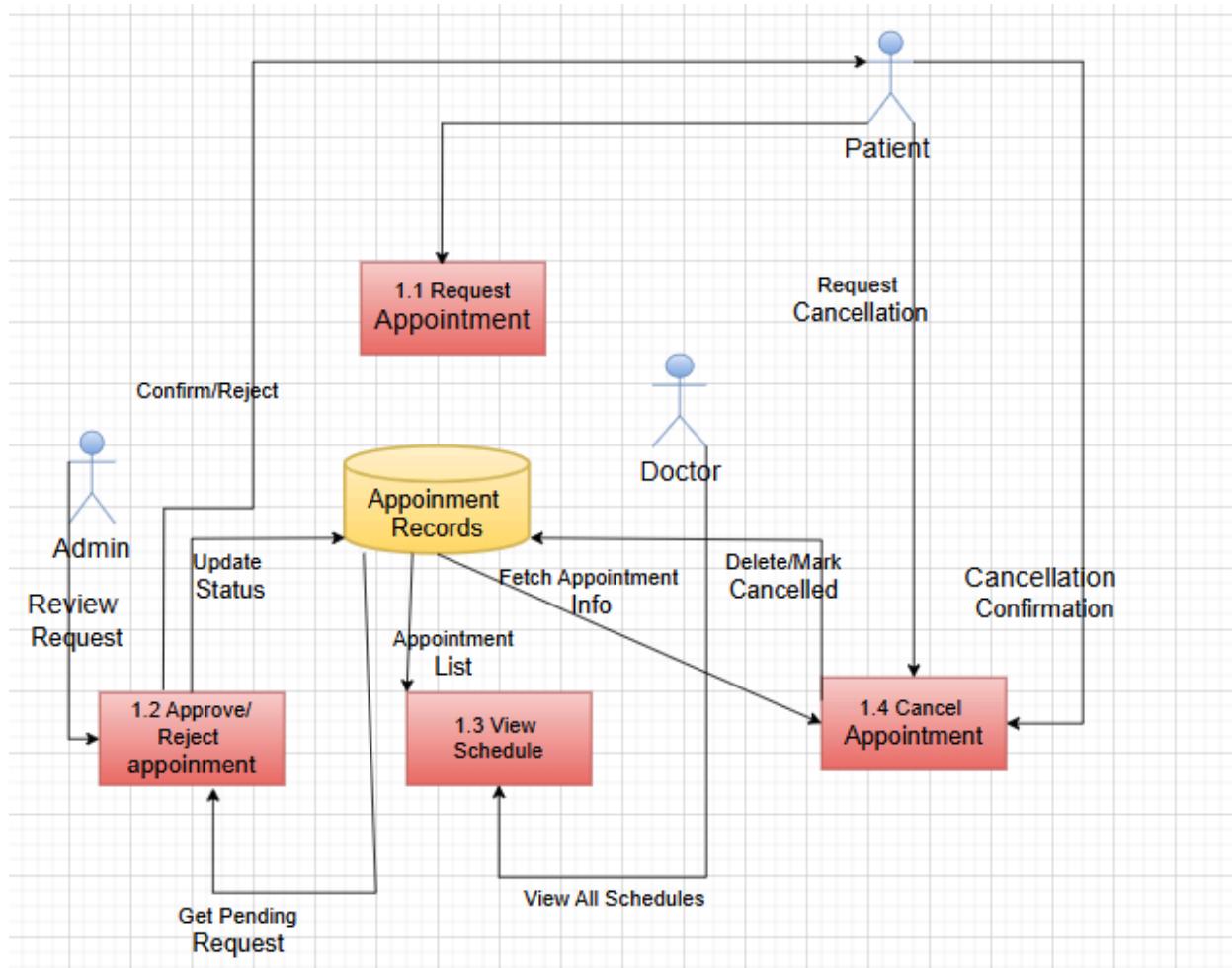
Data Flow Diagram (DFD Level-0):



The Level 0 DFD provides a high-level overview of a patient management system. The system's architecture is centered around four main processes: Manage Appointments, Manage Patient Info, Manage Doctors, and Billing & Payments. These processes are connected to three external entities—Patient, Doctor, and Admin—and four data stores—Patient DB, Appointments Records, Doctor DB, and Billing Info.

The Patient submits information and requests appointments, receiving confirmations in return. The Doctor can update patient information and view their schedule. The Admin manages doctor information and handles billing. The data stores act as central repositories: Patient DB stores patient details, Appointments Records holds appointment information, Doctor DB contains doctor profiles, and Billing Info stores all billing-related data. The data flows show the movement of information, such as the patient's submission of information to the "Manage Patient Info" process, which then saves it to the "Patient DB." The "Manage Appointments" process, for example, handles appointment requests from the patient and saves them to the "Appointments Records" data store, while also allowing the doctor to view their schedule by getting data from the same store.

Data Flow Diagram (DFD Level-1):

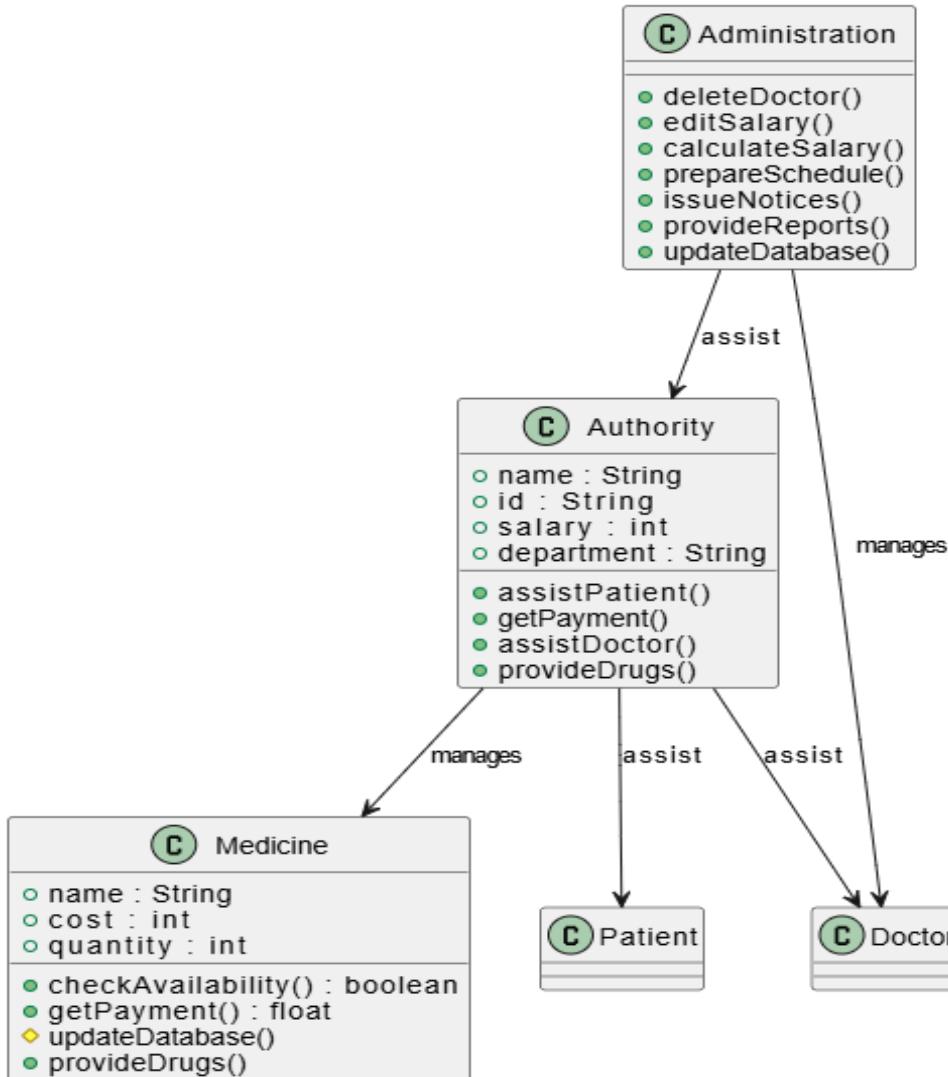


The Level 1 DFD, a decomposition of the "Manage Appointments" process from the Level 0 diagram, provides a detailed view of how appointments are handled. It illustrates four sub-processes: Request Appointment, Approve/Reject appointment, View Schedule, and Cancel Appointment. These processes interact with the Patient, Doctor, and Admin entities, and the central Appointment Records data store.

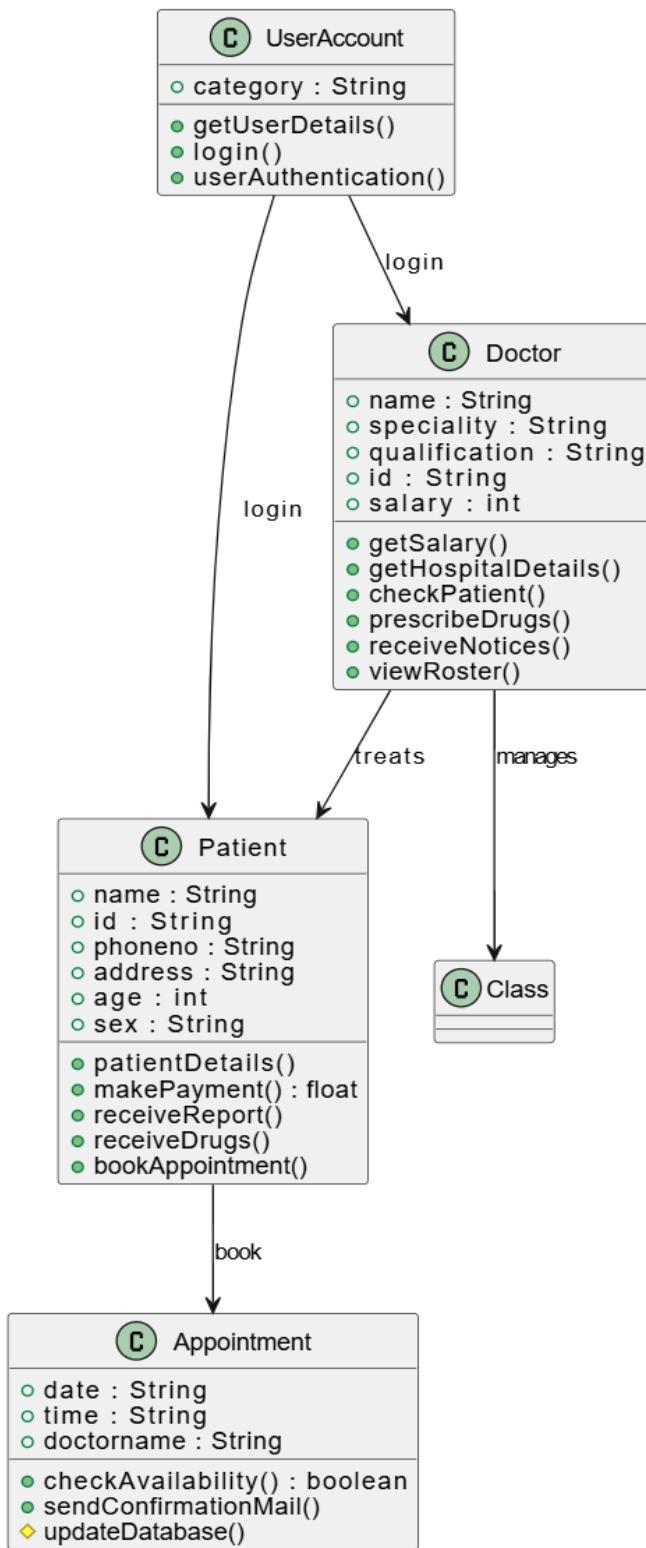
A Patient initiates the process by making a request through the Request Appointment sub-process, which saves the request to the Appointment Records data store. The Admin then reviews these pending requests via the Approve/Reject appointment sub-process, updating the status in the data store and sending a confirmation or rejection to the patient. The Doctor can view all schedules by fetching information from the data store through the View Schedule sub-process. If

a patient needs to cancel an appointment, they do so through the Cancel Appointment sub-process, which updates the status in the data store and provides a confirmation to the patient. The doctor can also mark or delete a cancelled appointment.

Class Diagram: (Part-1)



Class Diagram: (Part-2)

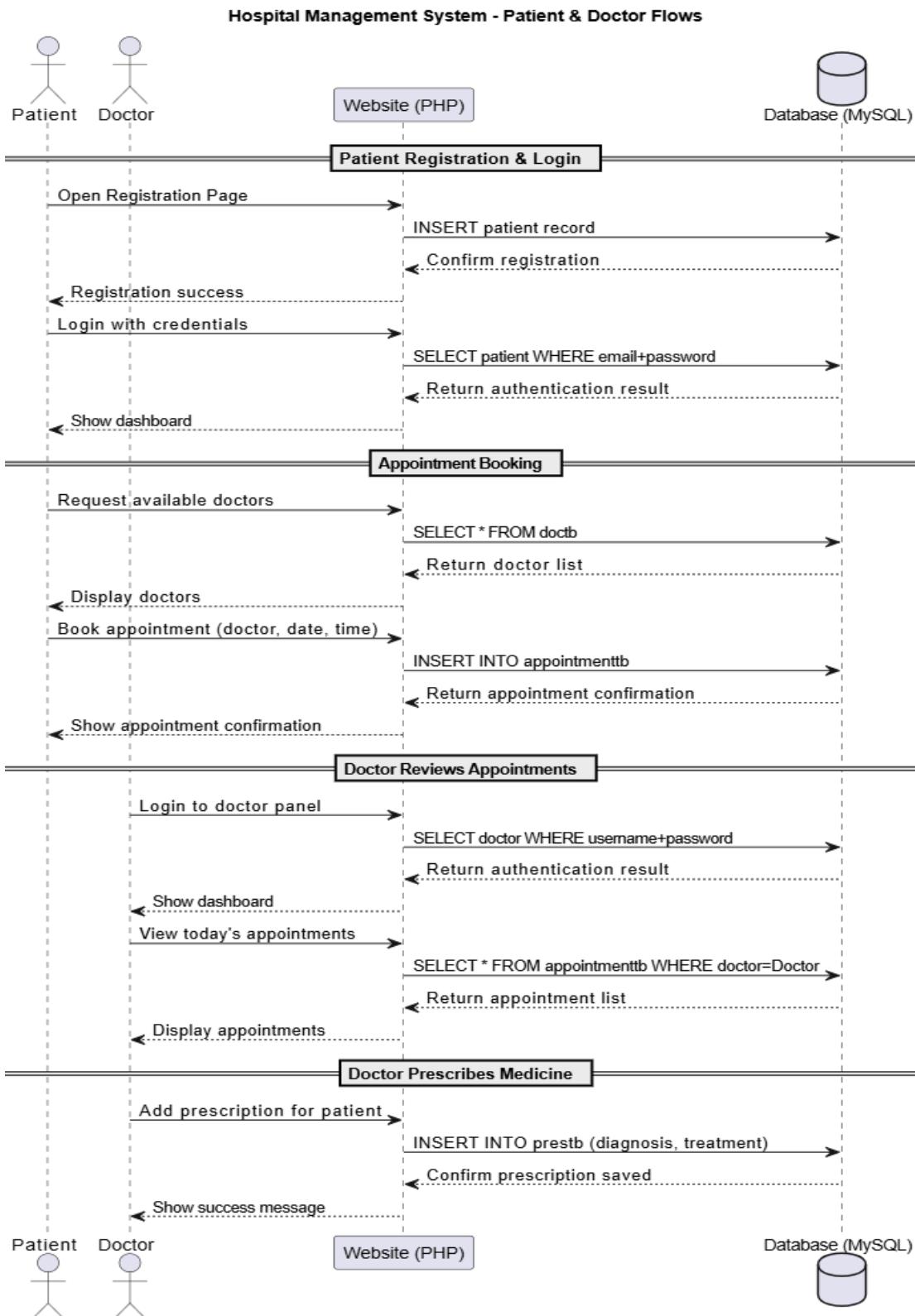


The class diagrams illustrate the architecture of our Hospital Management System, designed to manage patient care, doctor operations, appointments, administrative activities, and medicine distribution. The system supports multiple user roles, including Patients, Doctors, Administrative Staff, and Support Authorities, each with dedicated functions and responsibilities.

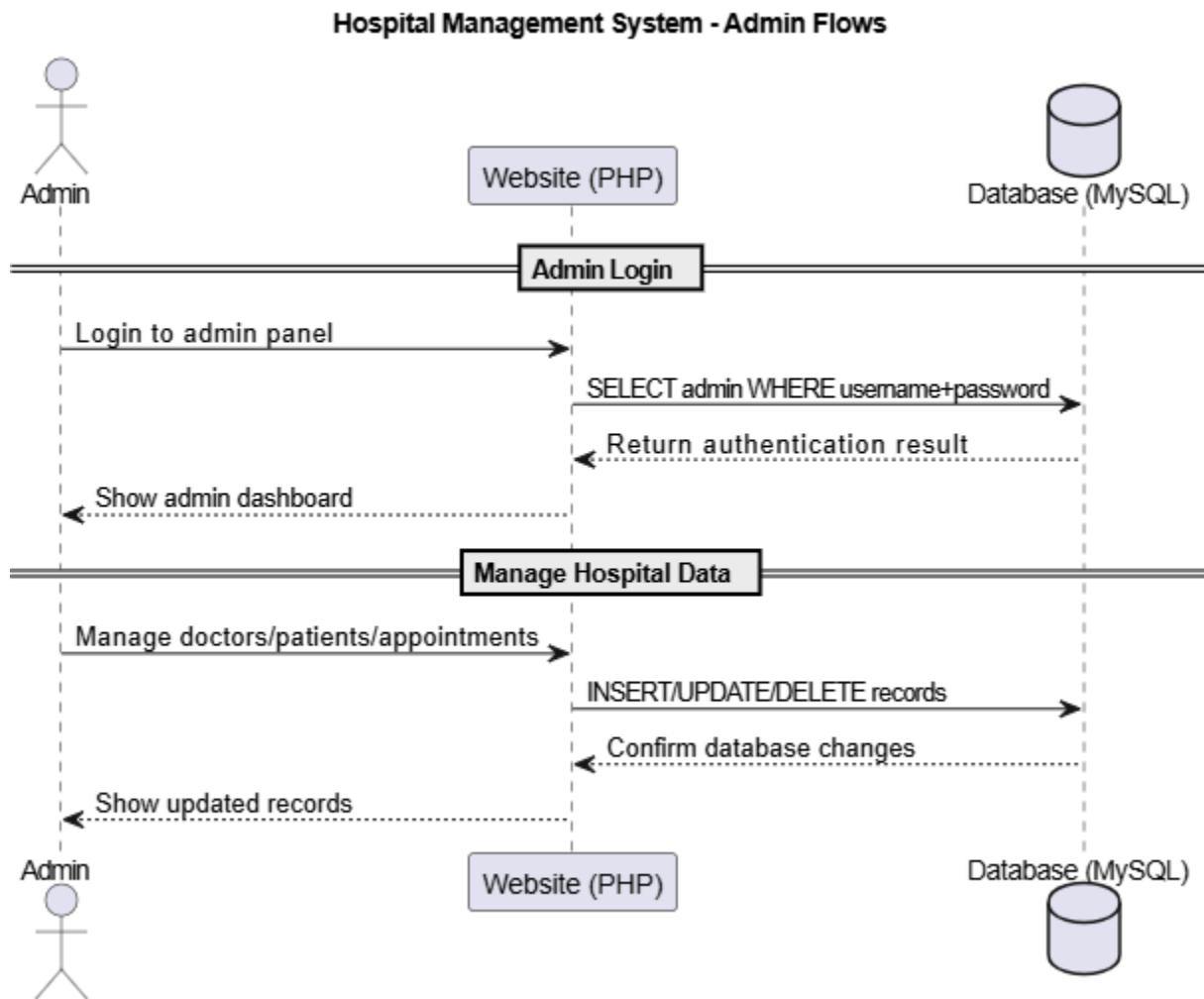
The primary entities include User Accounts, Patients, Doctors, Appointments, Administration, Authority, and Medicines. User Accounts manage authentication and role-based access for both patients and doctors. Patients maintain personal details, book appointments, make payments, receive medical reports, and collect prescribed medicines. Doctors hold professional and hospital-related information, perform patient checkups, prescribe drugs, view schedules, and receive notices. Appointments store details such as date, time, and assigned doctor, while ensuring slot availability and confirmation. The administrative module oversees staff management, salary processing, schedule preparation, report generation, and record updates. Authorities act as support staff, assisting patients and doctors, managing payments, and handling medicine distribution. The Medicine entity maintains inventory details such as name, cost, quantity, and availability, with capabilities for dispensing drugs and updating records. The interconnections between these entities are critical to system functionality. User Accounts link to Patients and Doctors for login and authentication. Doctors are associated with Patients through medical treatment, while Patients interact with Appointments for scheduling. Administrative staff manage Doctors and Authorities, who in turn assist both Patients and Doctors. Authorities also handle and manage Medicines.

This interconnected structure allows the Hospital Management System to efficiently coordinate patient care, manage hospital resources, handle administrative tasks, and ensure seamless communication between all operational units within the healthcare facility.

Sequence diagram: (Part-1)



Sequence diagram: (Part-2)



This sequence diagram shows the interaction of different actors with the Hospital Management System in order. The workflows include Patients, Doctors, and Admin, and their operations are described. First, a patient registers in the system by submitting their details, which are stored in the database and confirmed. After successful registration, the patient logs into the system, and their credentials are verified with the database. Once authenticated, the patient gains access to the dashboard.

The patient can then search for available doctors, with the system retrieving and displaying the list from the database. Upon selecting a doctor, date, and time, the patient books an appointment, which is stored in the database, and a confirmation is returned.

The Doctor workflow begins with the doctor logging into the system. Their login credentials are validated against the database, and upon success, the doctor's dashboard is shown. The doctor can view the list of today's appointments, retrieved from the database. After consulting with a patient, the doctor adds a prescription containing the diagnosis and treatment details, which is inserted into the database and confirmed. The Admin workflow starts with the admin logging into the system. Their credentials are checked in the database, and the admin dashboard is displayed upon successful authentication. The admin manages hospital data, including doctors, patients, and appointments. These operations involve inserting, updating, or deleting records in the database, and the changes are confirmed before being displayed on the admin panel.

All workflows share common concepts: each process begins with user interaction, is validated by the system against stored data, performs the necessary operations in the database, and returns confirmations. This ensures smooth coordination among all actors and reliable data management within the Hospital Management System.

Methodology:

Data Description:

An online survey was conducted to understand the necessity and effectiveness of a digital Hospital Management System. A total of 31 participants took part in the survey, including patients, doctors, nurses, and hospital administrative staff. Each group was asked specific questions relevant to their roles to assess their experience with the current hospital system and gather suggestions for improvement. The survey focused on issues such as appointment scheduling, access to medical records, doctor-patient communication, and overall satisfaction with hospital services. The collected data helped identify key challenges and guided the development of a more efficient and user-friendly management system.

Outcome from the survey:

Section 1: Doctors' Opinions

1. "If patients got reminder messages or calls, would it help reduce missed appointments? Would a follow-up message system help you?"

- Total Responses: 11
- Yes: 72.7% of doctors believe reminder messages or calls would help reduce missed appointments, and a follow-up system would be beneficial.
- No: 18.2% of doctors do not think so.
- Maybe: 9.1% of doctors are unsure.

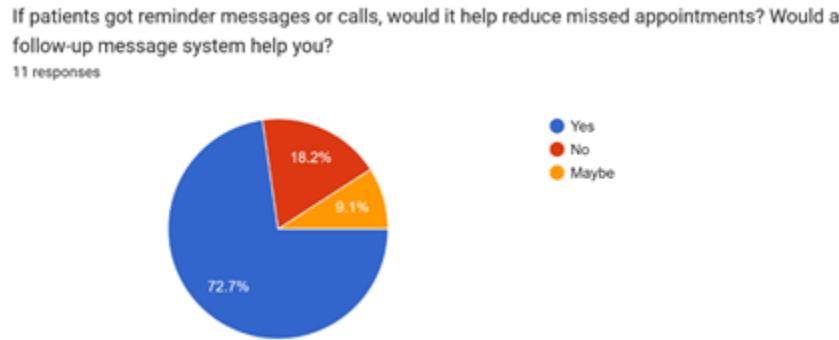


Figure-1

2. "Is it hard to work smoothly with lab staff or get test results on time?"

- Total Responses: 11
- Very Important: 54.5% of doctors consider smooth collaboration with lab staff and timely test results to be very important.
- Sometimes: 36.4% of doctors find it sometimes hard.
- Neutral: 9.1% of doctors are neutral on this matter.
- Not Important at all: 0%

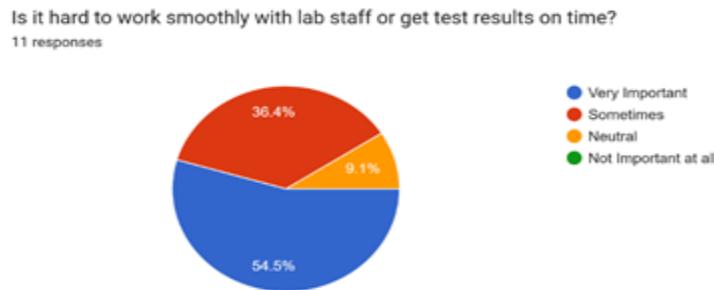


Figure-2

3. "Without a digital system, is it difficult to track how the patient's treatment is going over time?"

- Total Responses: 11
- Yes: 81.8% of doctors find it difficult to track patient treatment over time without a digital system.
- No: 0%
- Neutral: 9.1% of doctors are neutral.
- Not need at all: 9.1% of doctors believe it's not needed at all.

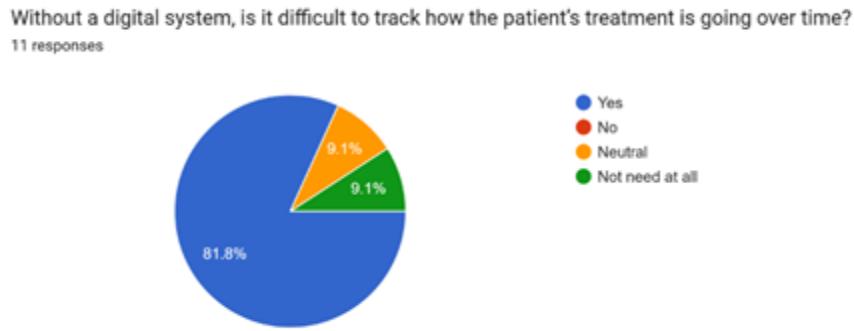


Figure-3

Section 2: Patients' Opinions

1. "Would it be better if you could see which doctors are available before going to the hospital?"

- Total Responses: 11
- Yes: 100% of patients agreed that it would be better to see doctor availability before going to the hospital.
- No: 0%

Would it be better if you could see which doctors are available before going to the hospital?
11 responses

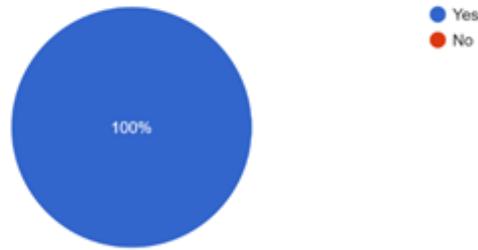


Figure-1

2. "How important is it for you to see your reports, prescriptions, and history on your phone or online?"

- Total Responses: 11
- Very important: 90.9% of patients consider it very important to access reports, prescriptions, and history on their phone or online.
- Somewhere important: 0%
- Neutral: 0%
- Not very important: 9.1% of patients consider it not very important.
- Not important at all: 0%

How important is it for you to see your reports, prescriptions, and history on your phone or online?
11 responses

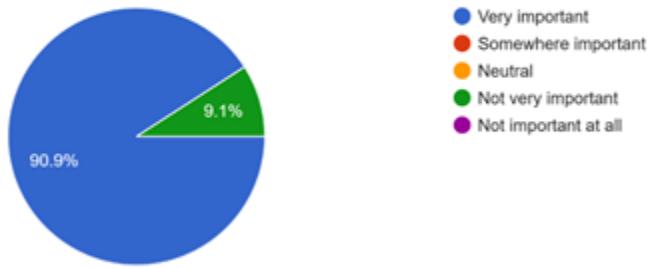


Figure-2

3. "How easy is it for you to find the right type of doctor (specialist) near your area?"

- Total Responses: 11

- Very Rare: 27.3% of patients find it very rare to find the right specialist.
- Rare: 45.5% of patients find it rare.
- Easy: 18.2% of patients find it easy.
- Very Easy: 9.1% of patients find it very easy.

How easy is it for you to find the right type of doctor (specialist) near your area?
11 responses

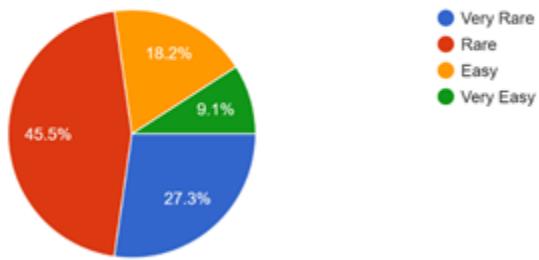


Figure-3

Section 3: Hospital Authority Opinions

1. "Is it hard to monitor staff attendance and work? Will automatic systems help manage better?"

- Total Responses: 9
- Yes: 66.7% of hospital authorities believe automatic systems would help manage staff attendance and work better.
- No: 0%
- Neutral: 33.3% of hospital authorities are neutral.

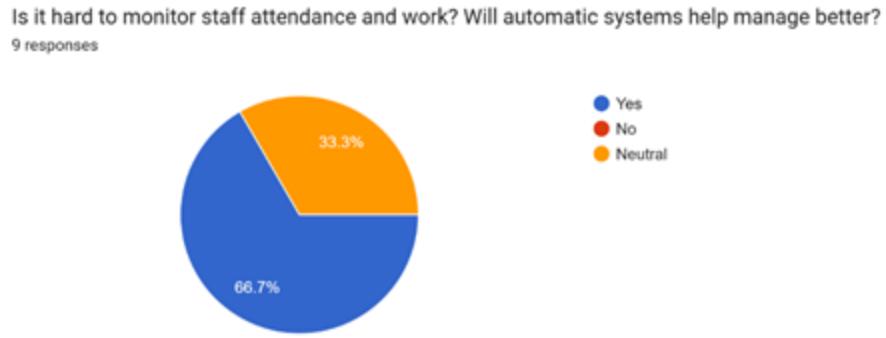


Figure-1

2. "Would a real time dashboard help in decision-making?"

- Total Responses: 9
- Yes: 100% of hospital authorities agreed that a real-time dashboard would help in decision-making.
- No: 0

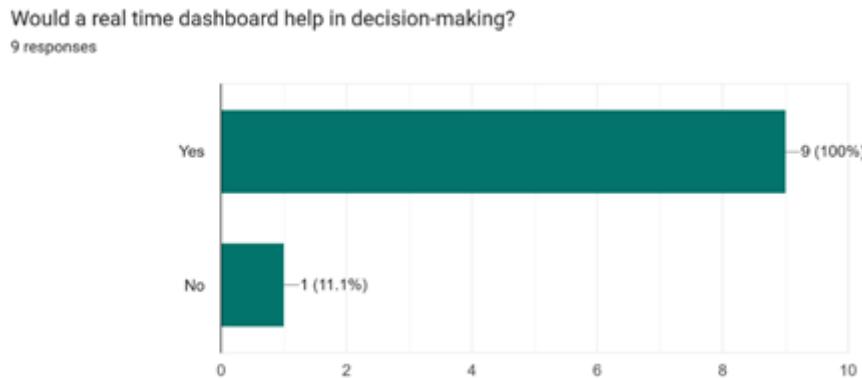


Figure-2

Stakeholder Feedback Report for Hospital Management System.

Project Management

Detailed Project Management Plan:

Project Objectives:

For our Hospital Management System project, my main goals are:

- To create a secure, scalable, and user-friendly web-based platform for hospital operations.
- To provide separate dashboards for Admins, Doctors, and Patients to ensure smooth and role-specific access.
- To improve hospital service delivery by replacing manual processes with automated and centralized data handling.

Key Phases:

Initiation:

- We will start by collecting requirements directly from hospital staff, doctors, and administrative teams.
- We will clearly define the scope and objectives of the system.
- We will ensure that all necessary resources (technical tools, skilled people, and budget) are ready before development.

Planning:

- We will prepare a detailed project timeline with all milestones.

- We will assign tasks to each team member according to their expertise.
- We will set up a risk management strategy so unexpected problems can be handled easily.

Execution:

- We will design the database schema and the overall system architecture.
- We will develop each role-based module (Admin, Doctor, Patient) with all required features.
- We will make sure that important features like appointment scheduling, medical records, and prescription management work smoothly.

Monitoring and Controlling:

- We will use project tracking tools to check progress regularly.
- We will compare the work with planned milestones and make adjustments if needed.
- We will collect feedback from users during the development phase.

Closure:

- We will finalize and deploy the system in the hospital environment.
- We will review performance, fix any remaining issues, and complete all documentation.
- We will officially hand over the system for daily use.

Resource Allocation:

- **Human Resources:** Myself as Project Manager & Developer, additional PHP developers if needed, a UI/UX designer, a database administrator, and testers.
- **Technical Resources:** VS Code, Apache Server, PHPMyAdmin, MySQL database, web hosting server, and testing tools.

Timeline:

Task	Duration
Requirement Collection	4 weeks
System Design	6 weeks
Development	12 weeks
Testing & Iteration	8 weeks
Deployment	4 weeks
Training & Documentation	4 weeks
Total	38 weeks

Risk Management:

- **Technical Risks:** We will reduce them by doing regular testing and reviewing the code.
- **User Adoption Risks:** We will arrange training sessions for hospital staff to make the transition easier.
- **Budget Risks:** We will keep a contingency fund to cover unexpected expenses.

Deliverables

- Fully functional Hospital Management System.

- User manuals and training materials.
- Post-deployment technical support.

Full Project Plan:

- **Phase 1:** Requirement Analysis – meet with stakeholders and gather requirements.
- **Phase 2:** System Design – create database schema, architecture, and UI mockups.
- **Phase 3:** Development – build front-end and back-end modules with secure authentication.
- **Phase 4:** Testing – functional, performance, and UAT testing.
- **Phase 5:** Deployment – install the system on hospital servers or hosting.
- **Phase 6:** Training & Documentation – train staff and prepare user guides.
- **Phase 7:** Scaling & Feedback – expand to more departments and apply user feedback.

Budget and Total Cost:

Cost Category	BDT Amount (Approx.)
Developer Salaries	2,500,000 BDT
UI/UX Designer Salary	500,000 BDT
Cloud Hosting	220,000 BDT
Testing Tools & Software	100,000 BDT
Training Sessions	80,000 BDT

Marketing & Awareness	150,000 BDT
Equipment & Tools	100,000 BDT
Contingency Fund	200,000 BDT
Total Cost	3,850,000 BDT

Contribution to Society:

The Hospital Management System developed in PHP plays a vital role in addressing some of the long-standing issues faced by the healthcare sector in Bangladesh. It provides a digital platform for managing patients, doctors, appointments, and prescriptions, which contributes to both efficiency and accessibility in medical services across the country. In a developing country like Bangladesh, where healthcare facilities are often overburdened and understaffed, the use of such systems has the potential to bring about significant positive changes.

1. Promoting Digital Healthcare (E-Health) in Line with “Digital Bangladesh”

The Government of Bangladesh has long emphasized the development of ICT-based public services under the vision of “Digital Bangladesh.” This hospital management system directly aligns with that vision by promoting digitization in the healthcare sector. It encourages clinics and hospitals, especially small and mid-level facilities, to shift from manual, paper-based processes to efficient digital solutions.

2. Reducing Administrative Burden and Overcrowding

One of the common problems in public hospitals and clinics in Bangladesh is overcrowding and long waiting lines for appointments. By introducing a system where patients can view doctors' availability and book appointments online, the project significantly reduces unnecessary hospital visits and improves patient flow. Doctors and administrative staff also save time by handling fewer manual records, thus improving overall service delivery.

3. Accessibility for Rural and Underserved Areas

Many rural healthcare centers in Bangladesh lack proper documentation and scheduling systems. This project, being lightweight and developed with open-source tools like PHP and MySQL, is cost-effective and scalable. It can be easily implemented even in small local clinics or NGOs operating in rural areas. With basic internet access, rural healthcare providers can benefit from organized patient records, appointment scheduling, and digital prescriptions.

4. Enhancing Doctor–Patient Communication

Through the system, patients can view their appointment history, prescription details, and other medical records at any time. This improves transparency and strengthens communication between doctors and patients. Doctors can also review past interactions and provide more accurate diagnoses based on recorded history, reducing the likelihood of misdiagnosis or medical errors.

5. Streamlining Hospital Operations

From the admin's perspective, the system centralizes control over doctors, patients, and medical records. It allows administrators to track hospital activities, manage medical staff, and handle patient data in a secure and efficient way. This improves data accuracy, ensures accountability, and helps in generating reports or statistics that can support healthcare planning and decision-making.

6. Encouraging Health Awareness and System Trust

As patients interact with structured systems that offer clarity, consistency, and access to their personal health data, it builds trust in the healthcare process. Better managed systems also promote health awareness as patients are more likely to follow up with appointments, prescriptions, and treatments when everything is organized and accessible.

7. A Step Toward Paperless Hospitals

Bangladesh is yet to fully adopt digital records in healthcare. With systems like this one, hospitals and clinics can gradually transition to paperless management, which not only saves time and costs but also helps preserve the environment and reduces risks of data loss.

Conclusion:

The **Hospital Management System** developed using PHP is a robust, dynamic, and user-centric web application designed to enhance and digitalize the workflow of hospital operations. By incorporating a structured interface and a role-based access model, the system addresses the needs of three primary stakeholders—**patients**, **doctors**, and **administrators**—ensuring that each user group can interact with the system effectively and securely. For **patients**, the system simplifies the often-tedious process of hospital visits by allowing them to **book appointments online**, **view their medical history**, and **receive digital prescriptions**. This not only improves patient convenience and engagement but also reduces unnecessary queues and administrative delays. Patients can access their health records at any time, empowering them to take a more active role in managing their healthcare. **Doctors**, on the other hand, benefit from a centralized and organized platform to manage their daily consultations. The system allows doctors to **access patient records**, **update diagnoses**, and **issue prescriptions** efficiently. It ensures that vital patient data is readily available during appointments, which improves the quality of care and reduces errors caused by missing or inconsistent information. The **administrator module** plays a crucial role in maintaining the overall operation of the system. Admins have full control over **user registration**, **appointment scheduling**, **doctor allocation**, and **record management**. The system enables efficient **data retrieval and report generation**, allowing hospital staff to make data-driven decisions and maintain accountability. Furthermore, access controls and data validation mechanisms have been integrated to maintain security, privacy, and the integrity of sensitive information. This project serves as a significant demonstration of how **core web development technologies**—such as PHP, HTML, CSS, JavaScript, and MySQL—can be combined to create a practical solution for real-life challenges. Beyond technical execution, the system provides a platform to analyze and understand the operational gaps that exist in traditional hospital management processes. In particular, it sheds light on how digital systems can alleviate the workload on healthcare staff, reduce the scope for human error, and enhance the overall patient experience. In the context of **Bangladesh**, where many healthcare facilities—especially in rural and semi-urban areas—still rely on manual record-keeping and appointment handling, the implementation of such a system can be transformative. It aligns well with the government's vision of "**Digital Bangladesh**", which aims to leverage technology to improve public services, including healthcare. A digital hospital management system can help bridge the gap between urban and rural healthcare access, optimize hospital resources, and ensure timely and accurate treatment for patients across the country. So, overall, this project has been both a technical

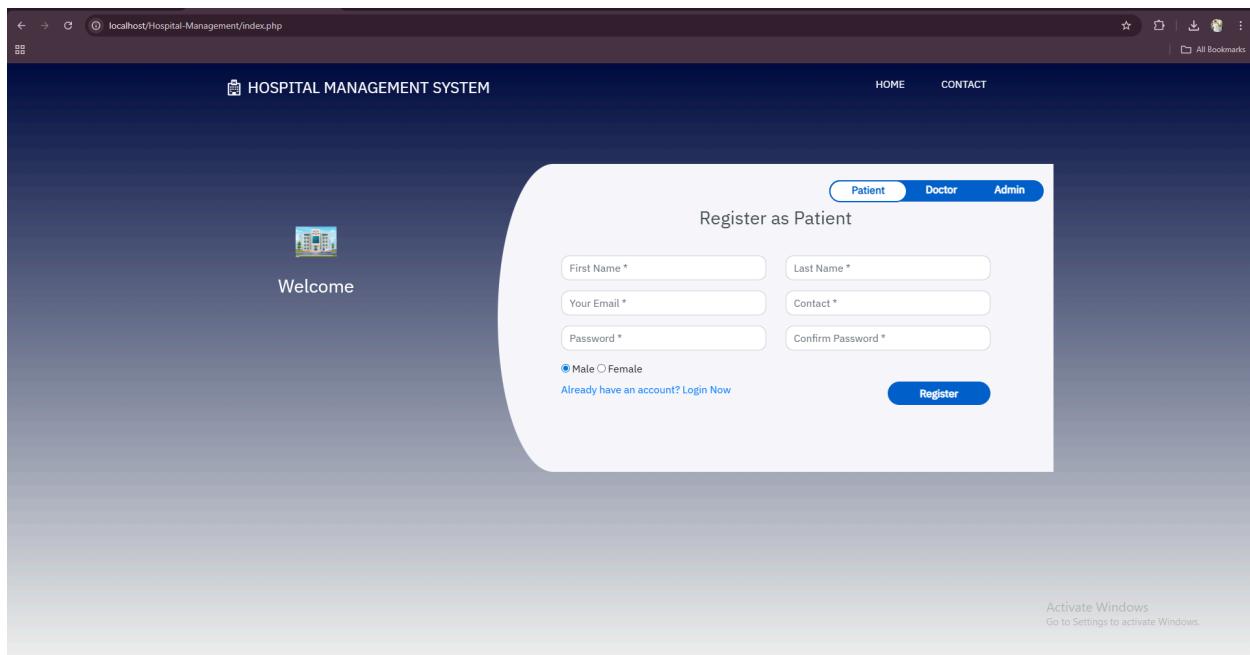
and social learning experience. It combines **software engineering principles** with **real-world application**, illustrating how even relatively simple digital tools can lead to substantial improvements in critical sectors like healthcare. Through this system, we've not only addressed technical challenges but also explored the broader impact of digital transformation in society.

Reference:

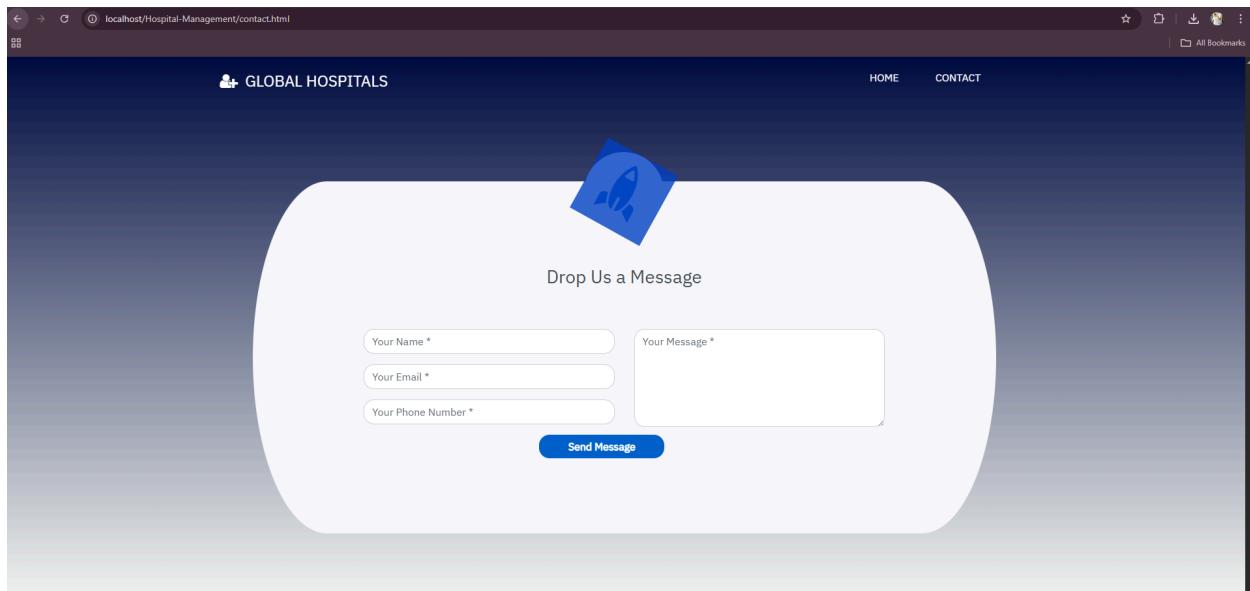
- [1] Web-Based Hospital Management System Ismaael A. Sikiru1 , Rafiat A. Oyekunle2
- [2] Hospital Management System using Web Technology Jharana Paikray1 ,Laxmikanta Sahoo2 , Krishna Kumar Patra3 ,Prasangsita Khara4 , Priyanka Shit5 1,2,3,4GIET, Baniatangi, Khorda 5RITE, Bhubaneswar
- [3] Development of Web -Based Clinic Management System Nur Syafinaz Zamri1 , Yana Mazwin Mohmad Hassim1*
- [4] A Web-Based Information System Applied on Utilization/Benefit Management of Medical Equipment in Hospital Chia-Hung Chien1 , Man-Hsiang Chang 2 and Yi-You Huang 1
- [5] Web Based Hospital Management System Rahman, Asifur Independent University, Bangladesh
- [6] Hospital management system using web technology is a demand of time Received: 19 Nov 2023 | Revised version received: 25 Mar 2024 | Accepted: 26 Mar 2024 | Published online: 30 Mar 2024
Responsible Editor: M Mostafa Zaman | Reviewer A: Anonymous Mansoor Ahmed Siddiqui Asher Fawwad Amna Mansoo
- [7] A WEB BASED APPLICATION FOR CLINICAL LABORATORY INFORMATION MANAGEMENT SYSTEM 1
Zinah Jaffar Mohamed Ameen
- [8] Design and Development of a Web Application for Medical Clinic Management. BOUSSAID Alaeddine HAFFERSSAS Aymen
- [9] Implementation of Web-based Management System for Pacs Network Nasser N. Khmiss 1 , Nehad Hameed Hussein 2 Information Engineering College, Al-Nahrain University, Iraq
- [10] WEB BASED HOSPITAL MANAGEMENT SYSTEM FOR ANVER ISMAIL MEMORIAL HOSPITAL SAMMANTHRUI FARSANA. LTI Registration No: SEU/IS/07/IT/049 Index No: IT049
- [11] A WEB-BASED MEDICAL RECORDS MANAGEMENT SYSTEM FOR NIGERIA (A CASE STUDY OF PILGRIMS WELFARE AGENCY USMAN, Nuradeen Umar 1153-04156-0216

Website Prototype:

Home Page:



Contract Page:



Patient Registration page if they do not have any account:

localhost/Hospital-Management/index.php

HOSPITAL MANAGEMENT SYSTEM

HOME CONTACT

Welcome

Register as Patient

Patient Doctor Admin

First Name * Last Name *

Your Email * Contact *

Password * Confirm Password *

Male Female

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

[Register](#)

Activate Windows
Go to Settings to activate Windows.

Login Page for Patient :

localhost/Hospital-Management/index1.php

HOSPITAL MANAGEMENT SYSTEM

HOME CONTACT

We are here for you!

Patient Login

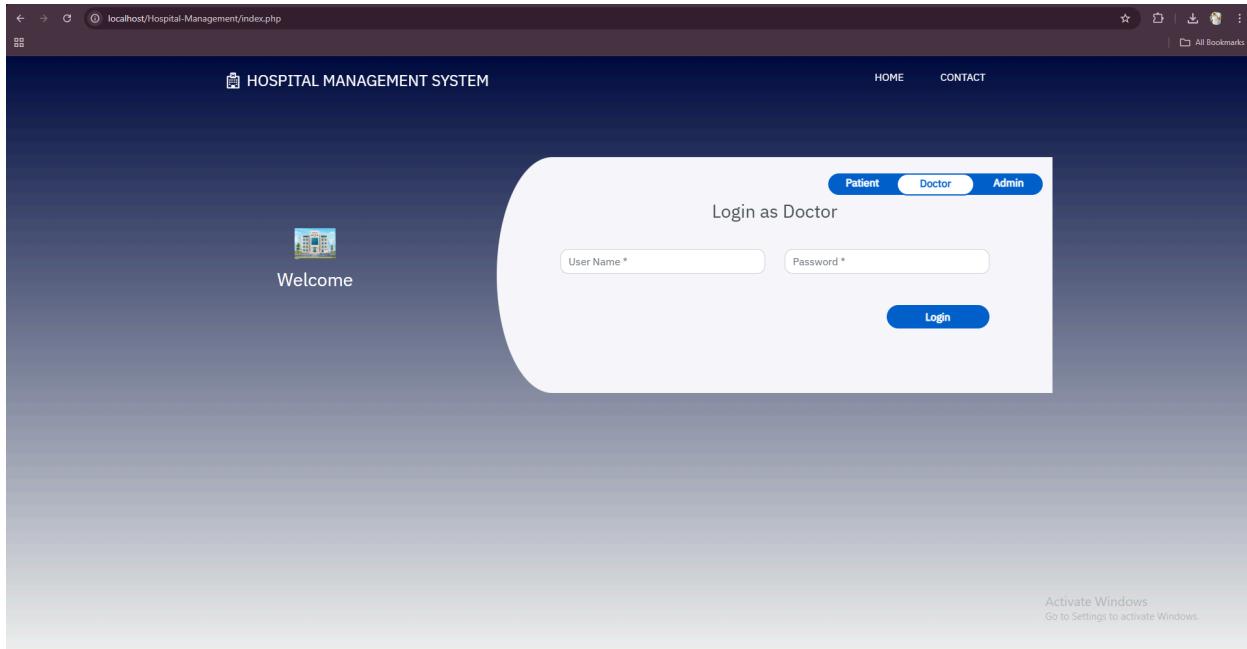
Email:

Password:

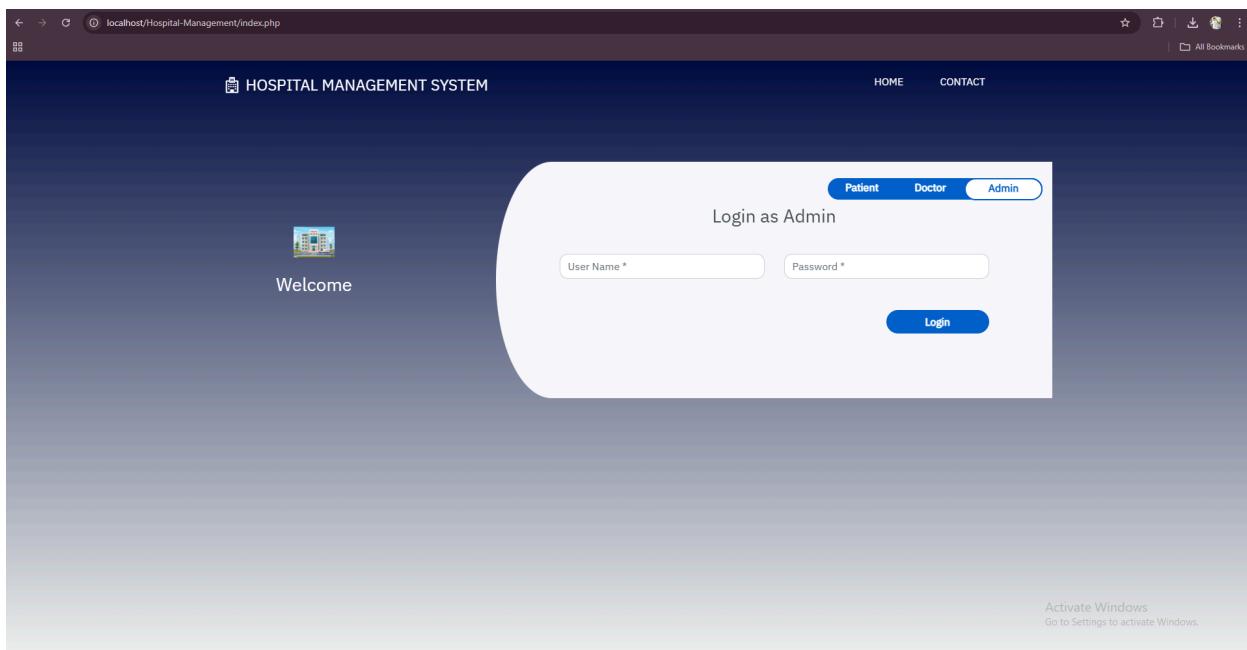
[Login](#)

Activate Windows
Go to Settings to activate Windows.

Login Page for Doctor:



Login Page for Admin:



Patient Dashboard Page:

Welcome sara khan

Dashboard

- Book Appointment
- Appointment History
- Prescriptions

Book My Appointment
[Book Appointment](#)

My Appointments
[View Appointment History](#)

Prescriptions
[View Prescription List](#)

Book Appointment for Patient:

Welcome sara khan

Dashboard

- Book Appointment**
- Appointment History
- Prescriptions

Book Appointment

Specialization:

Doctors:

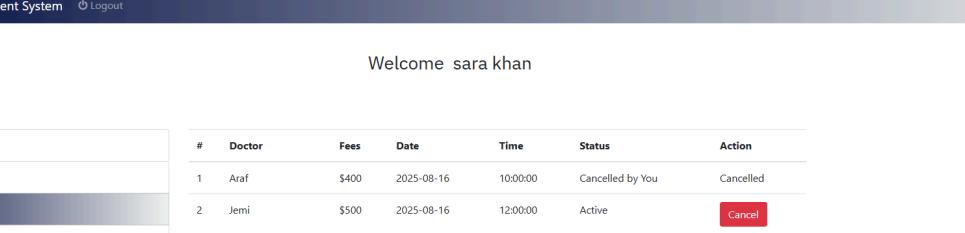
Consultancy Fees:

Appointment Date:

Appointment Time:

Create new entry

Patient's Appointment History Page:



Welcome sara khan

#	Doctor	Fees	Date	Time	Status	Action
1	Araf	\$400	2025-08-16	10:00:00	Cancelled by You	Cancelled
2	Jemi	\$500	2025-08-16	12:00:00	Active	<button>Cancel</button>
3	alsobarcafan	\$150	2025-08-27	12:00:00	Active	<button>Cancel</button>
4	Araf	\$400	2025-09-30	16:00:00	Active	<button>Cancel</button>

Dashboard

Book Appointment

Appointment History

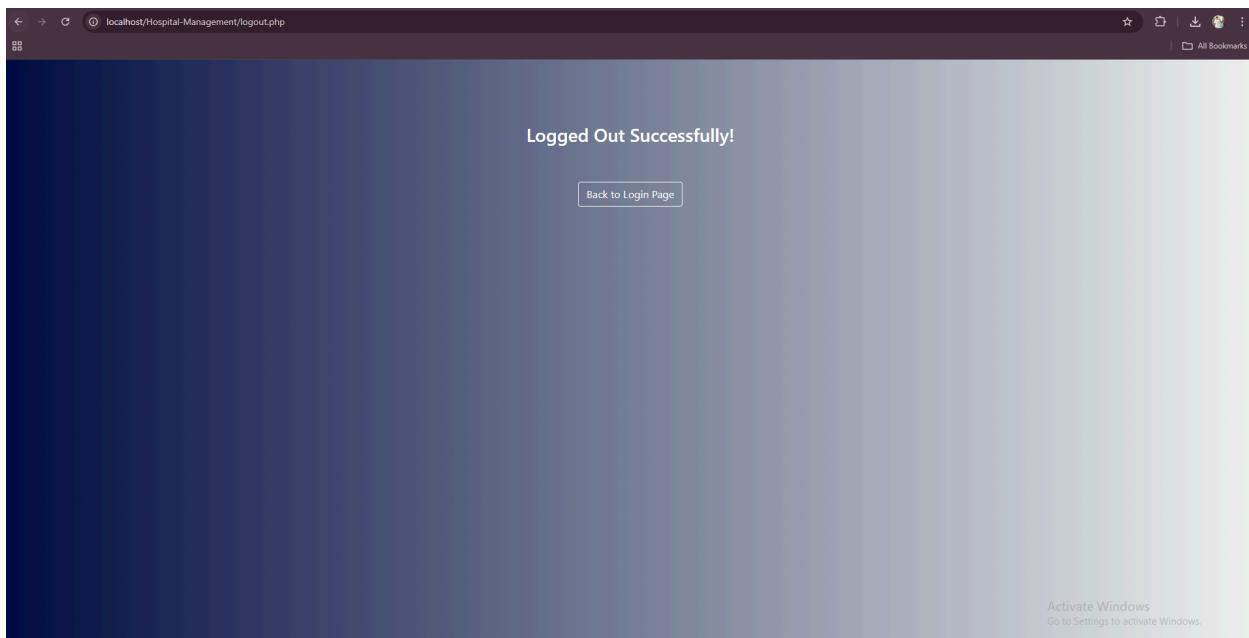
Prescriptions

Prescription Page:

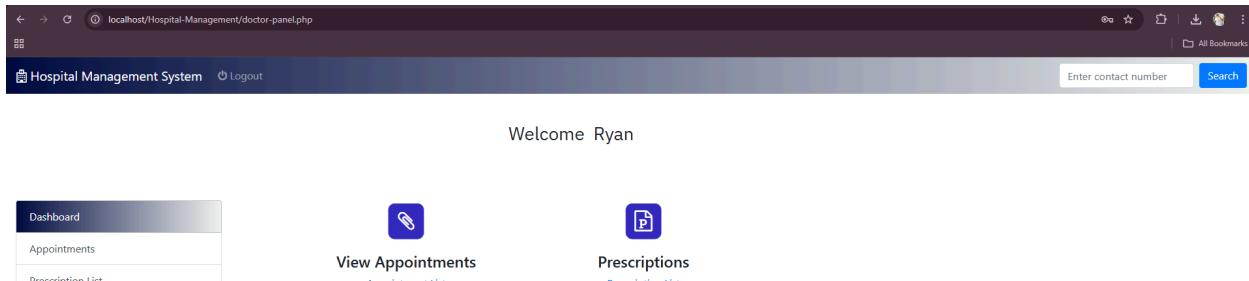
The screenshot shows a web-based hospital management system. At the top, a navigation bar includes links for 'Dashboard', 'Book Appointment', 'Appointment History', and 'Prescriptions'. The 'Prescriptions' link is highlighted with a dark blue background. The main content area displays a welcome message 'Welcome sara khan'. Below this, a table lists a patient's appointment details: Doctor (Jemi), Date (2025-06-25), Time (14:30:00), Diseases (Ovarian cysts), Allergies (00000000), Prescriptions (Isoniazid, Ethambutol (Myambutol), Linezolid), and a 'Payment' section with a 'Pay Bill' button. The entire interface has a clean, modern design with a light blue header and white background.

Doctor	Date	Appointment Time	Diseases	Allergies	Prescriptions	Payment
Jemi	2025-06-25	14:30:00	Ovarian cysts	00000000	Isoniazid, Ethambutol (Myambutol), Linezolid	Pay Bill

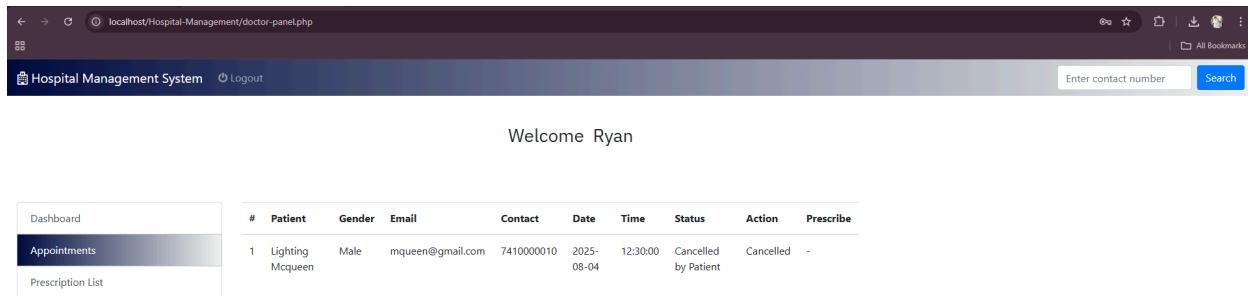
Patient Logout Page:



Doctor Dashboard Page:



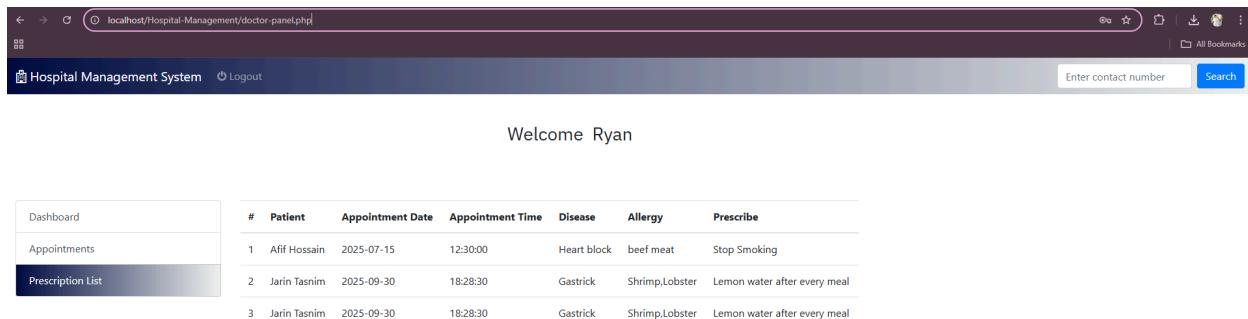
Doctor checking the appointment list:



Welcome Ryan

#	Patient	Gender	Email	Contact	Date	Time	Status	Action	Prescribe
1	Lighting Mcqueen	Male	mqueen@gmail.com	7410000010	2025-08-04	12:30:00	Cancelled by Patient	Cancelled	-

Doctor Checking the Prescription List:



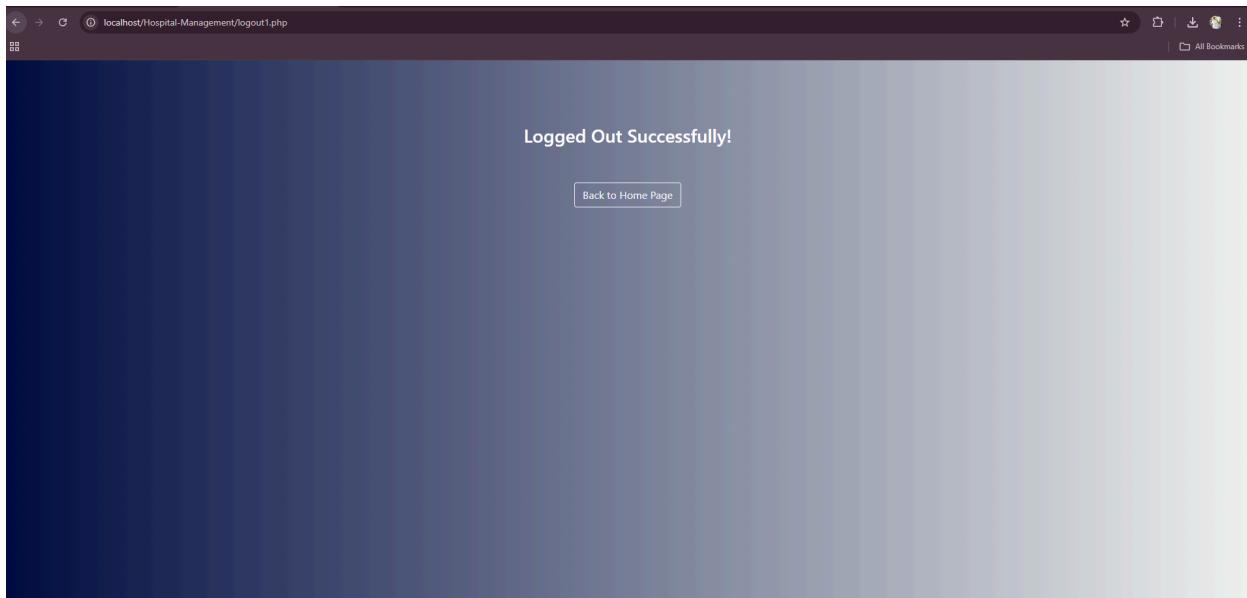
Welcome Ryan

#	Patient	Appointment Date	Appointment Time	Disease	Allergy	Prescribe
1	Afif Hossain	2025-07-15	12:30:00	Heart block	beef meat	Stop Smoking
2	Jarin Tasnim	2025-09-30	18:28:30	Gastrick	Shrimp,Lobster	Lemon water after every meal
3	Jarin Tasnim	2025-09-30	18:28:30	Gastrick	Shrimp,Lobster	Lemon water after every meal

Activate Windows
Go to Settings to activate Windows.



Logout Page for Doctor:



Admin Dashboard Page:

WELCOME ADMINISTRATOR

Dashboard

- [View Doctors](#)
- [View Patients](#)
- [View Patients \(API\)](#)
- [Appointment Details](#)
- [Prescription List](#)
- [Add Doctor](#)
- [Delete Doctor](#)
- [Queries](#)

Doctor List
[View Doctors](#)

Patient List
[View Patients](#)

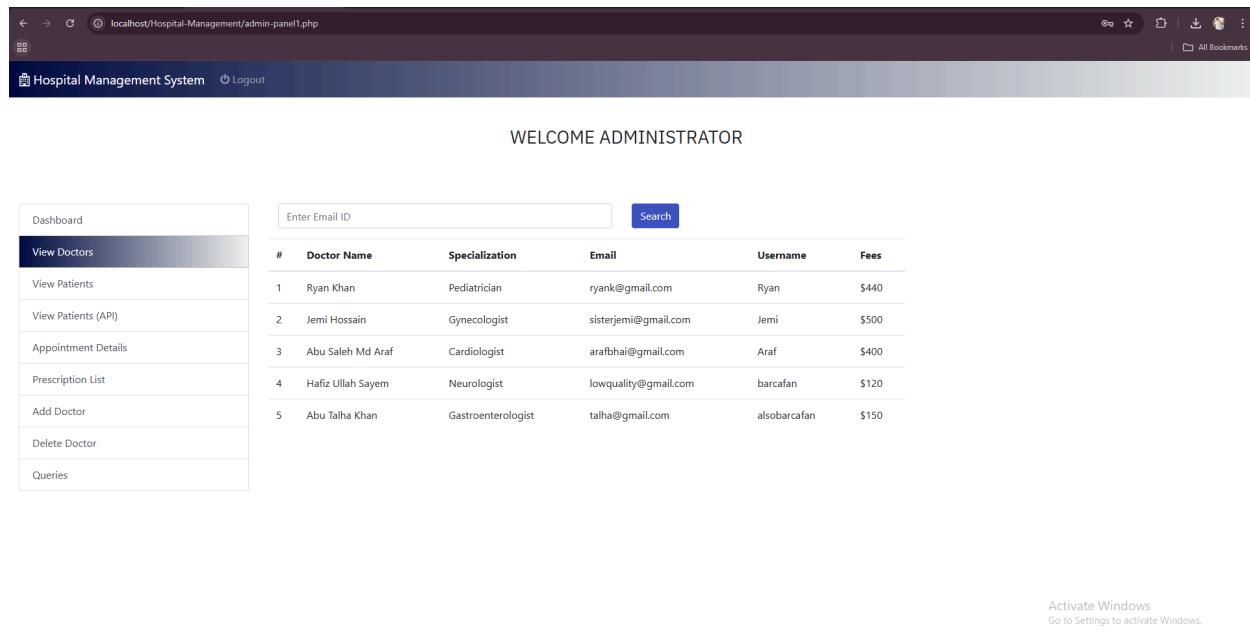
Appointment Details
[View Appointments](#)

Prescription List
[View Prescriptions](#)

Manage Doctors
[Add Doctors](#) | [Delete Doctors](#)

Activate Windows
Go to Settings to activate Windows.

Admin Viewing Doctors List:



localhost/Hospital-Management/admin-panel1.php

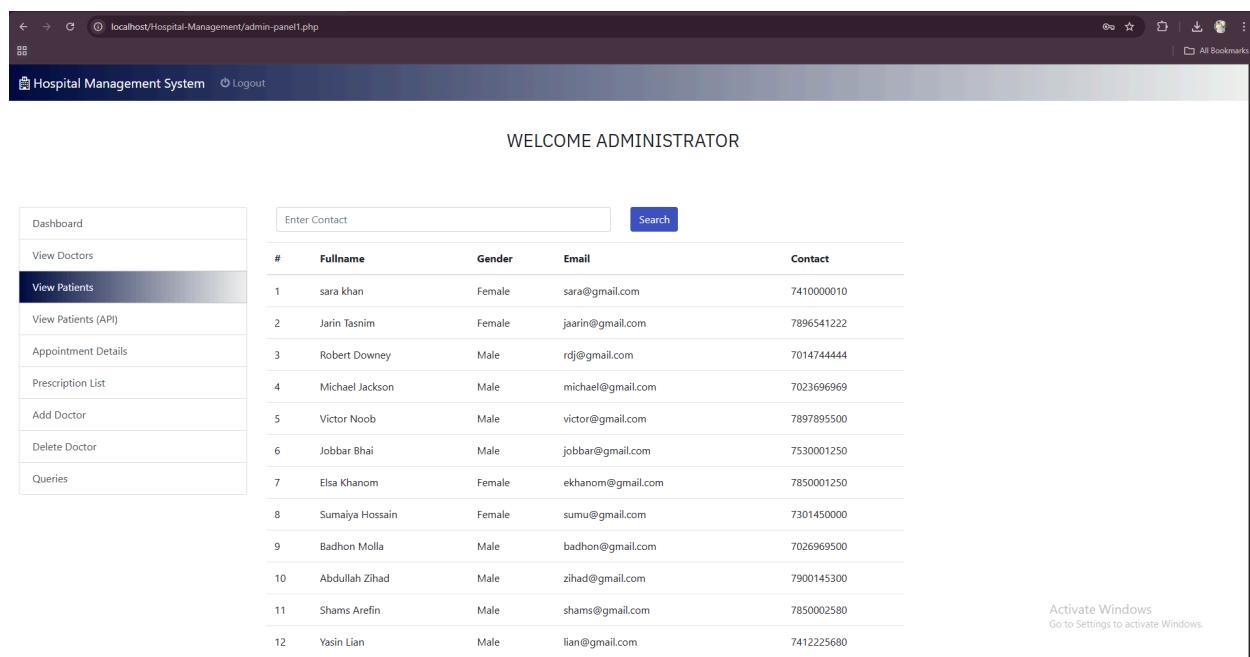
Hospital Management System Logout

WELCOME ADMINISTRATOR

#	Doctor Name	Specialization	Email	Username	Fees
1	Ryan Khan	Pediatrician	ryan@gmail.com	Ryan	\$440
2	Jemi Hossain	Gynecologist	sisterjemi@gmail.com	Jemi	\$500
3	Abu Saleh Md Araf	Cardiologist	arafbhai@gmail.com	Araf	\$400
4	Hafiz Ullah Sayem	Neurologist	lowquality@gmail.com	barcafan	\$120
5	Abu Talha Khan	Gastroenterologist	talha@gmail.com	alsobarcafan	\$150

Activate Windows
Go to Settings to activate Windows.

Admin Viewing Patients List:



localhost/Hospital-Management/admin-panel1.php

Hospital Management System Logout

WELCOME ADMINISTRATOR

#	Fullname	Gender	Email	Contact
1	sara khan	Female	sara@gmail.com	7410000010
2	Jarin Tasnim	Female	jaarin@gmail.com	7896541222
3	Robert Downey	Male	rdj@gmail.com	7014744444
4	Michael Jackson	Male	michael@gmail.com	7023696969
5	Victor Noob	Male	victor@gmail.com	7897895500
6	Jobbar Bhai	Male	jobbar@gmail.com	753001250
7	Elsa Khanom	Female	ekhanom@gmail.com	7850001250
8	Sumaiya Hossain	Female	sumu@gmail.com	7301450000
9	Badhon Molla	Male	badhon@gmail.com	7026969500
10	Abdullah Zihad	Male	zihad@gmail.com	7900145300
11	Shams Arefin	Male	shams@gmail.com	7850002580
12	Yasin Lian	Male	lian@gmail.com	7412225680

Activate Windows
Go to Settings to activate Windows.

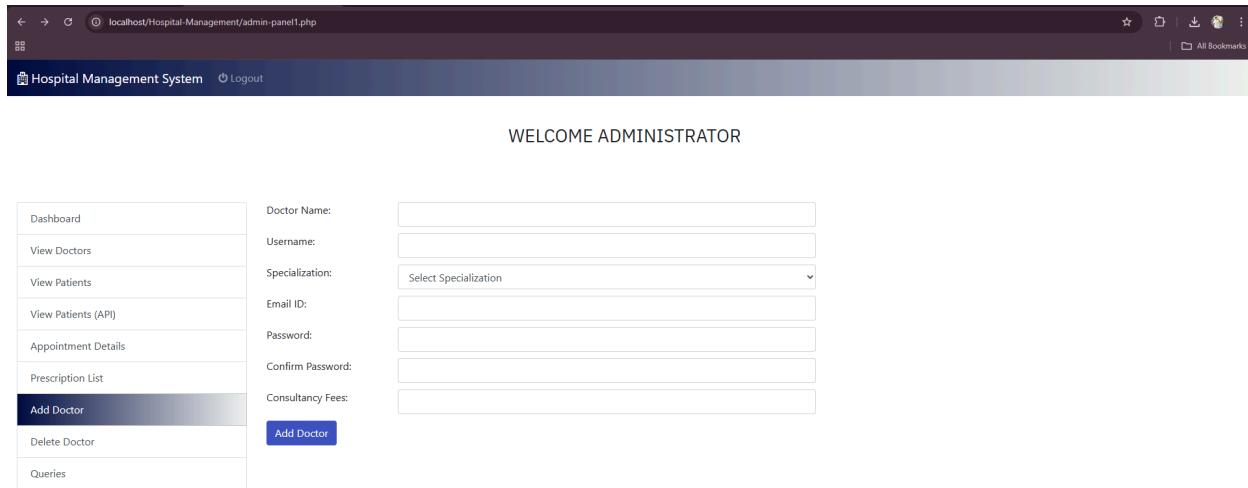
Appointment Details Page:

<div style="display: flex; justify-content: space-between;"> <div style="flex: 1;"> <input style="width: 100%; border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;" type="text" value="Enter Contact"/> </div> <div style="flex: 0.5;"> <input style="width: 100%; border: 1px solid #0056b3; background-color: #0056b3; color: white; padding: 5px;" type="button" value="Search"/> </div> </div>									
#	Fullname	Gender	Email	Contact	Doctor	Fees	Date	Time	Status
1	sami hossain	Male	sami@gmail.com	7412225680	Araf	\$400	2025-08-12	10:00:00	
2	Lighting Mcqueen	Male	mqueen@gmail.com	7410000010	ryan	\$440	2025-08-04	12:30:00	Cancelled by Patient
3	Hello World	Male	hw@gmail.com	7410000010	Jemi	\$500	2025-07-29	11:40:34	
4	Kotha Hoque	Female	kotha@gmail.com	7850002580	barcafan	\$120	2025-08-12	10:00:00	
5	Abu Saleh	Male	abu@gmail.com	7012569999	Araf	\$400	2025-08-10	11:00:00	
6	sara khan	Female	sara@gmail.com	7410000010	Araf	\$400	2025-08-16	10:00:00	Cancelled by Patient
7	sara khan	Female	sara@gmail.com	7410000010	Jemi	\$500	2025-08-16	12:00:00	Active
8	sara khan	Female	sara@gmail.com	7410000010	alsobarcafan	\$150	2025-08-27	12:00:00	Active
9	sara khan	Female	sara@gmail.com	7410000010	Araf	\$400	2025-09-30	16:00:00	Active

All Prescription List:

Patient Appointment Details							Actions	
Prescription List	#	Doctor	Fullscreen	Appointment Date	Appointment Time	Disease	Allergy	Prescription
	1	Jemi	Jariin Tasnim	2025-06-12	15:23:00	Congenital heart disease	rhinoconjunctivitis	trandolapril(Mavik)
	2	Jemi	sara khan	2025-06-25	14:30:00	Ovarian cysts	00000000	Isoniazid, Ethambutol (Myambutol), Linezolid
	3	Ryan	Afif Hossain	2025-07-15	12:30:00	Heart block	beef meat	Stop Smoking
	4	Araf	Badhon Mollah	2025-07-05	13:32:00	Cerebral Aneurysm	00000000	Nimodipine - empty stomach, at least 1 hour before breakfast
	5	Araf	Jarin Tasnim	2025-08-22	15:26:05	Chest Pain	Beef Meat	Don't Eat Beef Meat
	6	Araf	Jarin Tasnim	2025-08-22	15:26:05	Chest Pain	Beef Meat	Don't Eat Beef Meat
	7	Ryan	Jarin Tasnim	2025-09-30	18:28:30	Gastrick	Shrimp,Lobster	Lemon water after every meal
	8	Ryan	Jarin Tasnim	2025-09-30	18:28:30	Gastrick	Shrimp,Lobster	Lemon water after every meal
	9	barcatan	Shams Arefin	2025-10-09	18:28:30	Migraine pain	Nothing	Don't use mobile/laptop without putting glass
	10	barcatan	Shams Arefin	2025-10-09	18:28:30	Migraine pain	Nothing	Don't use mobile/laptop without putting glass

Doctor Adding Page:



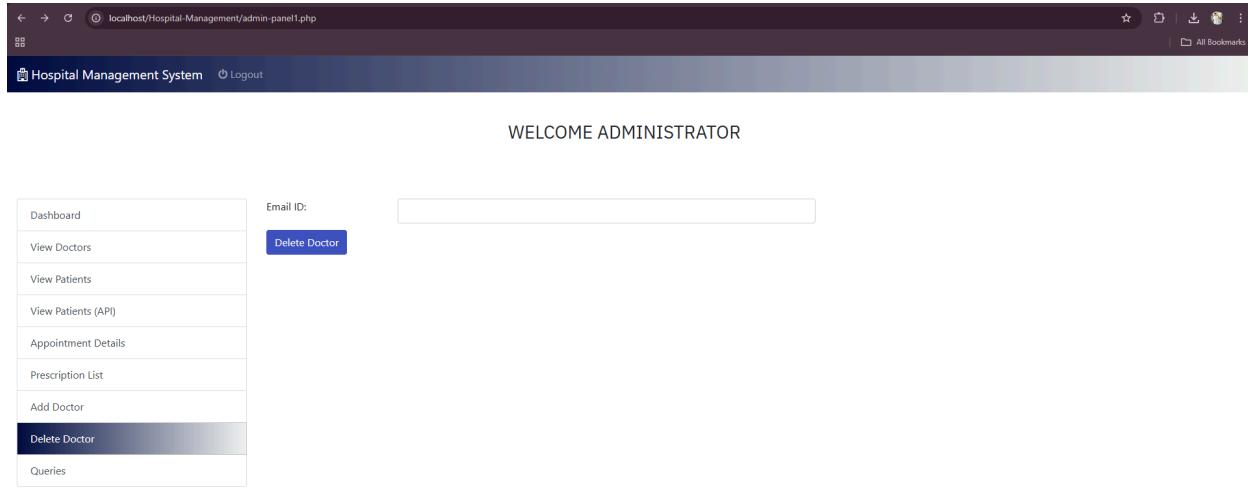
localhost/Hospital-Management/admin-panel1.php

Hospital Management System Logout

WELCOME ADMINISTRATOR

Dashboard	Doctor Name:	<input type="text"/>
View Doctors	Username:	<input type="text"/>
View Patients	Specialization:	<input type="text"/> Select Specialization
View Patients (API)	Email ID:	<input type="text"/>
Appointment Details	Password:	<input type="text"/>
Prescription List	Confirm Password:	<input type="text"/>
Add Doctor	Consultancy Fees:	<input type="text"/>
Delete Doctor	Add Doctor	
Queries		

Doctor Delete Page:



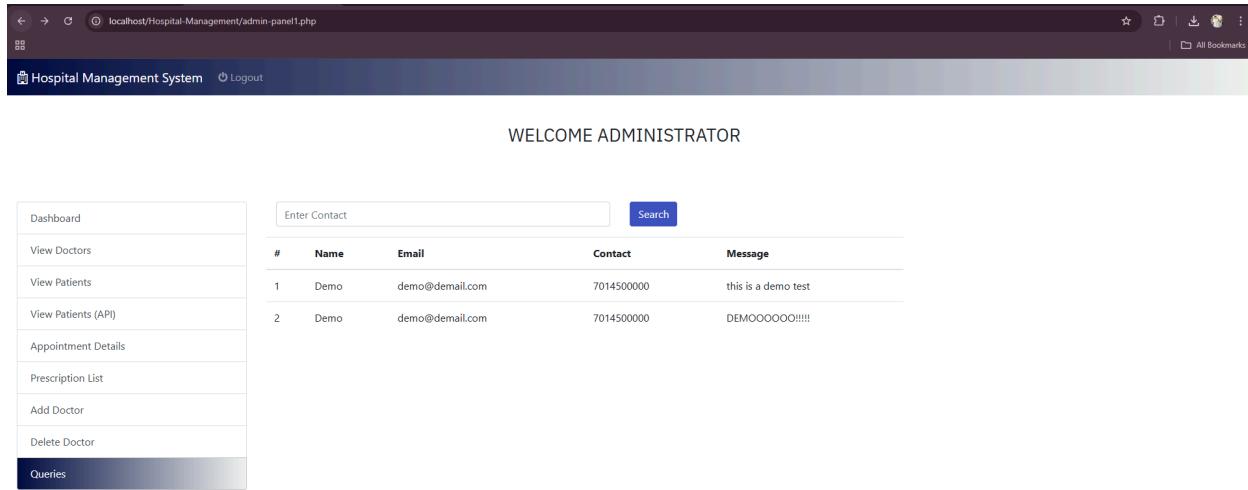
localhost/Hospital-Management/admin-panel1.php

Hospital Management System Logout

WELCOME ADMINISTRATOR

Dashboard	Email ID:	<input type="text"/>
View Doctors	Delete Doctor	
View Patients		
View Patients (API)		
Appointment Details		
Prescription List		
Add Doctor		
Delete Doctor		
Queries		

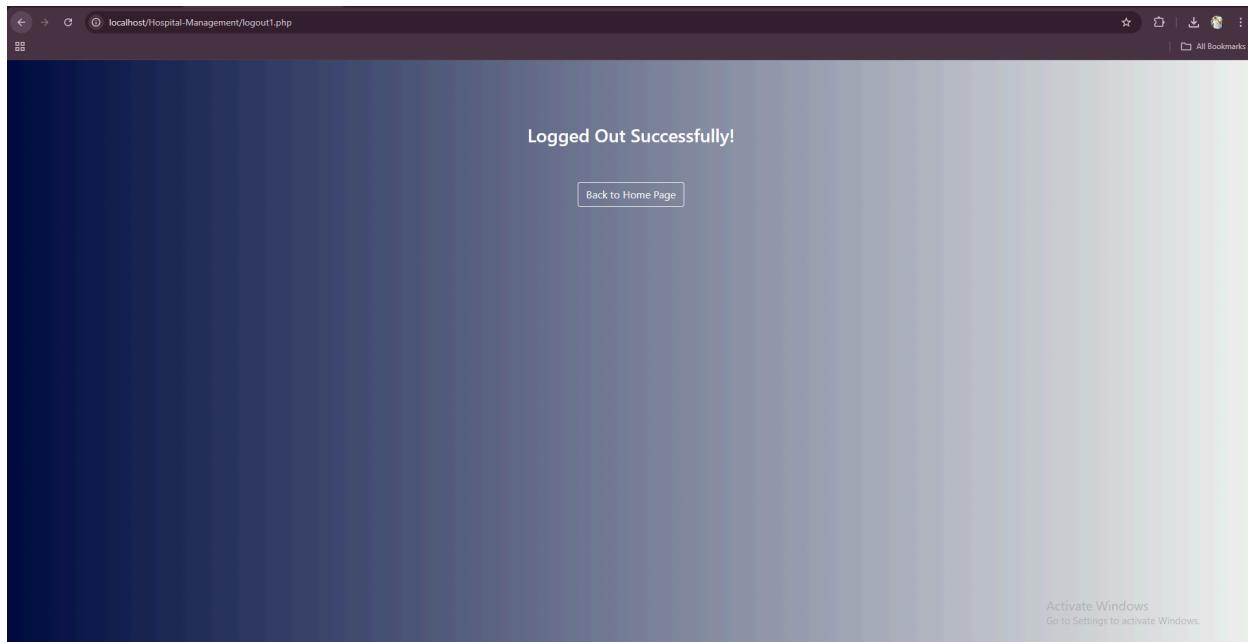
Query Page If Any Patient Has Any:



The screenshot shows a web browser window for the 'Hospital Management System' admin panel. The URL is 'localhost/Hospital-Management/admin-panel1.php'. The page is titled 'WELCOME ADMINISTRATOR'. On the left, there is a sidebar with various administrative links: Dashboard, View Doctors, View Patients, View Patients (API), Appointment Details, Prescription List, Add Doctor, Delete Doctor, and Queries. The 'Queries' link is highlighted with a dark blue background. The main content area features a search bar with 'Enter Contact' placeholder text and a 'Search' button. Below the search bar is a table with the following data:

#	Name	Email	Contact	Message
1	Demo	demo@demail.com	7014500000	this is a demo test
2	Demo	demo@demail.com	7014500000	DEMOOOOOO!!!!!!

Logout Page for Admin:



The screenshot shows a web browser window for the 'Hospital Management System' logout page. The URL is 'localhost/Hospital-Management/logout1.php'. The main content area displays the message 'Logged Out Successfully!' in a large, bold, white font. Below this message is a 'Back to Home Page' button. In the bottom right corner of the page, there is a small 'Activate Windows' watermark with the text 'Go to Settings to activate Windows.'

Code in Appendix:

1. Database connection & session bootstrap

```

<?php

// db_connect.php

declare(strict_types=1);

session_start();


$DB_HOST = "localhost";

$DB_USER = "root";

$DB_PASS = "";

$DB_NAME = "hospitalms";


mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);


try {

    $conn = new mysqli($DB_HOST, $DB_USER, $DB_PASS, $DB_NAME);

    $conn->set_charset("utf8mb4");

} catch (mysqli_sql_exception $e) {

    error_log("[DB] Connection failed: ".$e->getMessage());

    http_response_code(500);

    die("Database connection failed. Please try again later.");

}

// Small helper to require login by role

function require_role(string $role): void {

```

```

if (!isset($_SESSION['role']) || $_SESSION['role'] !== $role) {

    header("Location: error.php"); // or a 403 page

    exit();

}

}

```

```

// CSRF token bootstrap (use in forms)

if (empty($_SESSION['csrf_token'])) {

    $_SESSION['csrf_token'] = bin2hex(random_bytes(32));

}

```

This shows: session bootstrap, robust MySQL connection, UTF-8 charset, minimal auth helper, CSRF token creation.

2. Patient login authentication

```

<?php

// func.php (include 'db_connect.php' at the top of all handlers)

require_once __DIR__ . '/db_connect.php';

if (isset($_POST['login_patient'])) {

    if (!hash_equals($_SESSION['csrf_token'] ?? "", $_POST['csrf_token'] ?? "")) {

        $_SESSION['error'] = "Invalid session token.';

        header("Location: index.php");

        exit();

    }

    $email = trim($_POST['email'] ?? "");

```

```

$password = $_POST['password'] ?? "";

if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {

    $_SESSION['error'] = "Please enter a valid email address.";

    header("Location: index.php");

    exit();

}

$stmt = $conn->prepare("SELECT id, fname, lname, password_hash FROM patients WHERE email = ?");

$stmt->bind_param("s", $email);

$stmt->execute();

$stmt->bind_result($pid, $pf, $pl, $hash);

if ($stmt->fetch() && password_verify($password, $hash)) {

    $_SESSION['patient_id'] = (int)$pid;

    $_SESSION['patient_name'] = $pf . " " . $pl;

    $_SESSION['role'] = "patient";

    $_SESSION['flash'] = "Welcome back, $pf!";

    header("Location: patient-panel.php");

} else {

    error_log("[PATIENT LOGIN] Failed for $email");

    $_SESSION['error'] = "Invalid email or password.";

    header("Location: index.php");

}

$stmt->close();

}

```

This shows: email validation, password hashing verification, session creation, CSRF check, structured errors/logs.

3. Doctor login authentication

```
<?php

// func1.php

require_once __DIR__ . '/db_connect.php';

if (isset($_POST['login_doctor'])) {

    if (!hash_equals($_SESSION['csrf_token'] ?? "", $_POST['csrf_token'] ?? "")) {

        $_SESSION['error'] = "Invalid session token.';

        header("Location: index1.php");

        exit();
    }

    $username = trim($_POST['username'] ?? "");

    $password = $_POST['password'] ?? "";

    $stmt = $conn->prepare("SELECT id, name, password_hash FROM doctors WHERE username = ?");

    $stmt->bind_param("s", $username);

    $stmt->execute();

    $stmt->bind_result($did, $dname, $hash);

    if ($stmt->fetch() && password_verify($password, $hash)) {

        $_SESSION['doctor_id'] = (int)$did;

        $_SESSION['doctor_name'] = $dname;
    }
}
```

```

$_SESSION['role'] = "doctor";
$_SESSION['flash'] = "Logged in as Dr. $dname";
header("Location: doctor-panel.php");

} else {
    error_log("[DOCTOR LOGIN] Failed for $username");
    $_SESSION['error'] = "Invalid doctor credentials.";
    header("Location: index1.php");
}

$stmt->close();
}
}

```

This shows: doctor login via username, password_verify, role-based session, CSRF.

4. Admin login authentication

```

<?php
// func3.php
require_once __DIR__ . '/db_connect.php';

if (isset($_POST['login_admin'])) {
    if (!hash_equals($_SESSION['csrf_token'] ?? "", $_POST['csrf_token'] ?? "")) {
        $_SESSION['error'] = "Invalid session token.";
        header("Location: admin.php");
        exit();
    }
}

$username = trim($_POST['username'] ?? "");
$password = $_POST['password'] ?? "";

```

```

$stmt = $conn->prepare("SELECT id, username, password_hash FROM admin WHERE username =
?");

$stmt->bind_param("s", $username);

$stmt->execute();

$stmt->bind_result($aid, $auser, $hash);

if ($stmt->fetch() && password_verify($password, $hash)) {

    $_SESSION['admin_id'] = (int)$aid;

    $_SESSION['admin_user'] = $auser;

    $_SESSION['role'] = "admin";

    $_SESSION['flash'] = "Admin login successful.';

    header("Location: admin-panel.php");

} else {

    error_log("[ADMIN LOGIN] Failed for $username");

    $_SESSION['error'] = "Invalid admin credentials.';

    header("Location: admin.php");

}

$stmt->close();

}

```

This shows: admin authentication, hashed passwords, session, CSRF.

5. Patient registration

```

<?php

// func2.php

require_once __DIR__ . '/db_connect.php';

```

```

if (isset($_POST['register_patient'])) {

    if (!hash_equals($_SESSION['csrf_token'] ?? '', $_POST['csrf_token'] ?? '')) {
        $_SESSION['error'] = "Invalid session token.";
        header("Location: register.php");
        exit();
    }

    $fname = trim($_POST['fname'] ?? '');
    $lname = trim($_POST['lname'] ?? '');
    $email = trim($_POST['email'] ?? '');
    $phone = trim($_POST['phone'] ?? '');
    $pass = $_POST['password'] ?? '';
    $confirm = $_POST['confirm_password'] ?? '';

    $errors = [];

    if (!preg_match('/^([A-Za-z\']{2,})$/', $fname)) $errors[] = "First name invalid.";
    if (!preg_match('/^([A-Za-z\']{2,})$/', $lname)) $errors[] = "Last name invalid.";
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) $errors[] = "Email invalid.";
    if (!preg_match('/^+[0-9]{8,15}$/', $phone)) $errors[] = "Phone invalid.";
    if (strlen($pass) < 6) $errors[] = "Password must be ≥ 6 chars.";
    if ($pass !== $confirm) $errors[] = "Passwords do not match.";

    if ($errors) {
        $_SESSION['error'] = implode(' ', $errors);
        header("Location: register.php");
    }
}

```

```

exit();

}

// Unique email check

$stmt = $conn->prepare("SELECT 1 FROM patients WHERE email = ?");

$stmt->bind_param("s", $email);

$stmt->execute();

if ($stmt->get_result()->num_rows > 0) {

    $_SESSION['error'] = "Email already registered. ";

    header("Location: register.php");

    exit();

}

$stmt->close();

$hash = password_hash($pass, PASSWORD_DEFAULT);

$stmt = $conn->prepare("

    INSERT INTO patients (fname, lname, email, phone, password_hash, created_at)

    VALUES (?, ?, ?, ?, ?, NOW())

");

$stmt->bind_param("sssss", $fname, $lname, $email, $phone, $hash);

$stmt->execute();

$_SESSION['flash'] = "Registration successful. Please log in. ";

header("Location: index.php");

}

```

This shows: thorough validation, unique email check, password_hash, prepared INSERT, CSRF.

6. Contact / feedback submission

```
<?php

require_once __DIR__ . '/db_connect.php';

if (isset($_POST['submit_feedback'])) {

    if (!hash_equals($_SESSION['csrf_token'] ?? "", $_POST['csrf_token'] ?? "")) {

        $_SESSION['error'] = "Invalid session token./";

        header("Location: contact.html");

        exit();
    }

    $name = trim($_POST['name'] ?? "");

    $email = trim($_POST['email'] ?? "");

    $message = trim($_POST['message'] ?? "");

    if (!$name || !filter_var($email, FILTER_VALIDATE_EMAIL) || strlen($message) < 5) {

        $_SESSION['error'] = "Please provide valid name, email and a meaningful message./";

        header("Location: contact.html");

        exit();
    }

    $stmt = $conn->prepare("INSERT INTO feedback (name, email, message, created_at) VALUES (?, ?, ?, NOW())");

    $stmt->bind_param("sss", $name, $email, $message);
}
```

```

$stmt->execute();

$_SESSION['flash'] = "Thank you! Your feedback was submitted.";

header("Location: contact.html");

}

```

This shows: validation, prepared insert, session messages, CSRF.

7. Appointment booking with slot-clash check (patient booking handler)

```

<?php

require_once __DIR__ . '/db_connect.php';

require_role('patient'); // Only patients can book


if (isset($_POST['book_appointment'])) {

    if (!hash_equals($_SESSION['csrf_token'] ?? "", $_POST['csrf_token'] ?? "")) {
        $_SESSION['error'] = "Invalid session token./";

        header("Location: patient-panel.php");
        exit();
    }
}

$patient_id = (int)$_SESSION['patient_id'];

$doctor_id = (int)($_POST['doctor_id'] ?? 0);

$date      = $_POST['date'] ?? "";
$time      = $_POST['time'] ?? "";

// Basic constraints: future time, 15-min slots

```

```

$dtString = "$date $time:00";

$slot = DateTime::createFromFormat('Y-m-d H:i:s', $dtString);

$now = new DateTime('now');

if (!$slot || $slot <= $now) {

    $_SESSION['error'] = "Please choose a future time.';

    header("Location: patient-panel.php");

    exit();

}

// Clash check: same doctor, same timestamp

$stmt = $conn->prepare("

    SELECT id FROM appointments

    WHERE doctor_id = ? AND appt_datetime = ?

    AND status IN ('Pending','Approved')

    LIMIT 1

");

$stmt->bind_param("is", $doctor_id, $sqlDateTime);

$stmt->execute();

if ($stmt->get_result()->num_rows > 0) {

    $_SESSION['error'] = "Sorry, this slot is already booked for that doctor.';

    header("Location: patient-panel.php");

    exit();
}

```

```

}

$stmt->close();

$status = "Pending";
$payment_status = "Unpaid";

$stmt = $conn->prepare(
    "INSERT INTO appointments (patient_id, doctor_id, appt_datetime, status, payment_status,
    created_at)
    VALUES (?, ?, ?, ?, ?, NOW())
");

$stmt->bind_param("iisss", $patient_id, $doctor_id, $sqlDateTime, $status, $payment_status);
$stmt->execute();

$_SESSION['flash'] = "Appointment request submitted.";
header("Location: patient-panel.php");
}

```

This shows: future-time check, slot-clash detection, safe insert, patient-only access.

8. Appointment payment status update

```

<?php

require_once __DIR__ . '/db_connect.php';
require_role('admin');

if (isset($_POST['mark_paid'])) {
    if (!hash_equals($_SESSION['csrf_token'] ?? "", $_POST['csrf_token'] ?? "")) {

```

```

$_SESSION['error'] = "Invalid session token.";

header("Location: admin-panel.php");

exit();

}

$appt_id = (int)($_POST['appointment_id'] ?? 0);

$stmt = $conn->prepare("UPDATE appointments SET payment_status = 'Paid', payment_at = NOW()
WHERE id = ?");

$stmt->bind_param("i", $appt_id);

$stmt->execute();

$_SESSION['flash'] = "Payment marked as Paid. ";

header("Location: admin-panel.php");

}

```

Shows: role-restricted update, uses NOW() timestamp, CSRF.

9. Doctor prescription entry

```

<?php

require_once __DIR__ . '/db_connect.php';

require_role('doctor');

if (isset($_POST['save_prescription'])) {

    if (!hash_equals($_SESSION['csrf_token'] ?? "", $_POST['csrf_token'] ?? "")) {

        $_SESSION['error'] = "Invalid session token. ";

        header("Location: doctor-panel.php");

        exit();
    }
}

```

```

}

$appointment_id = (int)($_POST['appointment_id'] ?? 0);

$doctor_id = (int)$_SESSION['doctor_id'];

$notes = trim($_POST['prescription'] ?? "");

if (strlen($notes) < 5) {

    $_SESSION['error'] = "Please write a meaningful prescription.';

    header("Location: doctor-panel.php");

    exit();

}

// Ensure the appointment belongs to this doctor

$check = $conn->prepare("SELECT id FROM appointments WHERE id = ? AND doctor_id = ?");

$check->bind_param("ii", $appointment_id, $doctor_id);

$check->execute();

if ($check->get_result()->num_rows === 0) {

    $_SESSION['error'] = "Unauthorized appointment access.';

    header("Location: doctor-panel.php");

    exit();

}

$stmt = $conn->prepare("UPDATE appointments SET prescription = ?, status = 'Approved' WHERE id = ?");

$stmt->bind_param("si", $notes, $appointment_id);

$stmt->execute();

```

```

$_SESSION['flash'] = "Prescription saved.";
header("Location: doctor-panel.php");
}

```

This shows: doctor ownership check, validation, status update to Approved.

10. Doctor-specific appointment search

```

<?php
require_once __DIR__ . '/db_connect.php';
require_role('doctor');

$q = trim($_GET['q'] ?? '');
$doctor_id = (int)$_SESSION['doctor_id'];

$sql = "
SELECT a.id, p.fname, p.lname, a.appt_datetime, a.status, a.payment_status
FROM appointments a
JOIN patients p ON p.id = a.patient_id
WHERE a.doctor_id = ? AND (
    p.fname LIKE CONCAT('%', ?, '%') OR
    p.lname LIKE CONCAT('%', ?, '%') OR
    DATE(a.appt_datetime) = ?
)
ORDER BY a.appt_datetime DESC
";

```

```

$stmt = $conn->prepare($sql);

$dateTry = preg_match('/^\d{4}-\d{2}-\d{2}$/', $q) ? $q : '0000-00-00';

$stmt->bind_param("isss", $doctor_id, $q, $q, $dateTry);

$stmt->execute();

$result = $stmt->get_result();

// Render simple list/UI (or return JSON)

while ($row = $result->fetch_assoc()) {

    echo "<div class='appt-row'>#{$row['id']} &middot; {$row['fname']} {$row['lname']} &middot;
{$row['appt_datetime']} &middot; {$row['status']} &middot; {$row['payment_status']}</div>";

}

```

This shows: role isolation, flexible search (name/date), safe params, descending order.

11. Doctor lookup by email

```

<?php

require_once __DIR__ . '/db_connect.php';

require_role('admin');

$email = trim($_POST['email'] ?? "");

if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {

    $_SESSION['error'] = "Enter a valid email.";

    header("Location: admin-panel.php");

    exit();
}

$stmt = $conn->prepare("SELECT id, name, email, specialization FROM doctors WHERE email = ?");


```

```

$stmt->bind_param("s", $email);

$stmt->execute();

$res = $stmt->get_result();

if ($row = $res->fetch_assoc()) {

    $_SESSION['doctor_lookup'] = $row; // store for UI to render
    $_SESSION['flash'] = "Doctor found: ".$row['name'];

} else {

    $_SESSION['error'] = "No doctor found with that email./";

} header("Location: admin-panel.php");

```

This shows: email validation, admin-only, prepared SELECT, passing result to UI via session.

12. JSON API — add patient

```

<?php

require_once __DIR__ . '/../db_connect.php';

header('Content-Type: application/json');

if ($_SERVER['REQUEST_METHOD'] !== 'POST') {

    http_response_code(405);

    echo json_encode(["success" => false, "message" => "Use POST"]);

    exit();

}

$data = json_decode(file_get_contents('php://input'), true);

```

```

$fname = trim($data['fname'] ?? "");
$lname = trim($data['lname'] ?? "");
$email = trim($data['email'] ?? "");
$phone = trim($data['phone'] ?? "");
$password = $data['password'] ?? "";

if (!$fname || !$lname || !filter_var($email, FILTER_VALIDATE_EMAIL) || strlen($password) < 6) {
    http_response_code(422);
    echo json_encode(["success" => false, "message" => "Validation failed"]);
    exit();
}

$stmt = $conn->prepare("SELECT 1 FROM patients WHERE email = ?");

$stmt->bind_param("s", $email);
$stmt->execute();

if ($stmt->get_result()->num_rows > 0) {
    http_response_code(409);
    echo json_encode(["success" => false, "message" => "Email already exists"]);
    exit();
}

$stmt->close();

$hash = password_hash($password, PASSWORD_DEFAULT);

$stmt = $conn->prepare("

    INSERT INTO patients (fname, lname, email, phone, password_hash, created_at)

```

```

VALUES (?, ?, ?, ?, ?, NOW())
");

$stmt->bind_param("sssss", $fname, $lname, $email, $phone, $hash);

$stmt->execute();

echo json_encode(["success" => true, "id" => $conn->insert_id]);

```

This shows: RESTful validation/verbs, conflict check, hashed password, JSON responses.

13. JSON API — get patients

```

<?php

require_once __DIR__ . '/../db_connect.php';

header('Content-Type: application/json');

// Optional: admin-only

if (( $_SESSION['role'] ?? '' ) !== 'admin') {

    http_response_code(403);

    echo json_encode(["success" => false, "message" => "Forbidden"]);

    exit();
}

$res = $conn->query("SELECT id, fname, lname, email, phone, created_at FROM patients ORDER BY id DESC");

$rows = [];

while ($r = $res->fetch_assoc()) {

    $rows[] = $r;
}

```

```
echo json_encode($rows);
```

This shows: role guard, clean JSON, limited columns.

14. JSON API — delete patient

```
<?php

require_once __DIR__ . '/../db_connect.php';

header('Content-Type: application/json');

if (( $_SESSION['role'] ?? "" ) !== 'admin') {

    http_response_code(403);

    echo json_encode(["success" => false, "message" => "Forbidden"]);

    exit();

}

if ( $_SERVER['REQUEST_METHOD'] !== 'DELETE' ) {

    http_response_code(405);

    echo json_encode(["success" => false, "message" => "Use DELETE"]);

    exit();

}

$data = json_decode(file_get_contents('php://input'), true);

$id = (int)($data['id'] ?? 0);

if ($id <= 0) {

    http_response_code(422);

    echo json_encode(["success" => false, "message" => "Invalid id"]);

    exit();

}
```

```

}

$stmt = $conn->prepare("DELETE FROM patients WHERE id = ?");

$stmt->bind_param("i", $id);

$stmt->execute();

```

```
echo json_encode(["success" => true, "deleted" => $stmt->affected_rows > 0]);
```

This shows: DELETE verb, admin guard, validation, structured JSON reply.

15. Session termination

```

<?php

require_once __DIR__ . '/db_connect.php';

// Optionally clear everything except a flash message

$_SESSION = [];

if (ini_get("session.use_cookies")) {

    $params = session_get_cookie_params();

    setcookie(session_name(), "", time() - 42000,

        $params["path"], $params["domain"],

        $params["secure"], $params["httponly"]

    );

}

session_destroy();

header("Location: index.php?logged_out=1");

exit();

```

This shows: full session destruction, cookie invalidation, redirect with flag.

16. UI helper — doctor list filter by specialization

```
<!-- admin-panel.php -->

<input id="specFilter" type="text" placeholder="Filter by specialization..." oninput="filterDoctors()" />

<table id="doctorTable">

  <thead>
    <tr><th>Name</th><th>Email</th><th>Specialization</th></tr>
  </thead>

  <tbody>
    <!-- rows rendered from PHP -->
  </tbody>

</table>

<script>

function filterDoctors() {

  const term = (document.getElementById('specFilter').value || "").toLowerCase().trim();

  const rows = document.querySelectorAll('#doctorTable tbody tr');

  rows.forEach(row => {

    const spec = (row.cells[2]?.innerText || "").toLowerCase();

    row.style.display = spec.includes(term) ? '' : 'none';
  });
}

// Bonus: preserve filter on reload

document.addEventListener('DOMContentLoaded', () => {
  const last = sessionStorage.getItem('specFilter') || '';

```

```

document.getElementById('specFilter').value = last;
filterDoctors();
});

document.getElementById('specFilter').addEventListener('change', (e) => {
  sessionStorage.setItem('specFilter', e.target.value);
});

</script>

```

This shows: front-end table filtering, instant UI feedback, small UX nicety (persist filter).

Contribution Table:

Name	ID	Contribution (%)	Contribution Category
Sami Hossain	223014146	30%	Code Implementation, Debugging, Report Writing, Class diagram,ER Diagram,Activity Diagram,Data Flow Diagram.
Abu Saleh MD Araf	231014047	25%	Code Implementation, Debugging, Data Collection, Use Case Diagram, Report Writing,
Md Hafiz Ullah	223014199	25%	Code Implementation, Data Collection, Analysis, Report Writing,
Fahim Shihab Touhid	223014025	20%	Code Implementation, Data Collection, Sequence Diagram, Report Writing.



Fahim Shihab Touhid is pursuing a B.Sc. in Computer Science and Engineering at the University of Liberal Arts Bangladesh (ULAB), where he is building a strong academic foundation in computing and emerging technologies. His academic and research interests span cybersecurity, artificial intelligence and machine learning, DevOps, data science, and animation. Alongside his studies, he has developed professional experience as a telesales executive in international telemarketing agencies, which enhanced his adaptability, communication, and problem-solving abilities. Beyond academics and career, he has actively engaged in leadership and community development through volunteering at the Bangladesh Youth Leadership Center (BYLC) and serving as a scout in BNCC. With a passion for research and innovation, he aspires to pursue higher studies at the M.Sc. level and contribute to advancing knowledge in his chosen fields.



Sami Hossain is pursuing a B.Sc. in Computer Science and Engineering at the University of Liberal Arts Bangladesh (ULAB), with a strong academic background in data-driven applications and automation systems. He has recently developed a project using Arduino, also some web based projects. Sami's work demonstrates his skills in web development, database management, and project execution. Recognized for his critical thinking and public speaking, he placed 3rd in the GSCDC Nationals 2020 Debate Contest. His professional interests include business intelligence, data analytics, and web technologies.



Abu Saleh Md. Araf is pursuing a B.Sc. in Computer science and engineering at University of Liberal Arts Bangladesh (ULAB). His academic interests include web development, data science, and artificial intelligence. He has developed several projects, including a Student Exam Management System and an Online Job Portal. He is passionate about using technology to solve real-world problems and create innovative solutions.



Hafiz Ullah Sayem, a CSE undergrad at ULAB driven by an unshakable passion for AI, machine learning, and automation. I thrive on pushing boundaries—whether it's coding bold experiments, exploring futuristic space-tech ideas, or tackling challenges that demand creativity and critical thinking. Adaptability and fast learning are at the core of who I am, allowing me to turn curiosity into action and ideas into solutions. Beyond academics, I immerse myself in the digital world, where teamwork and problem-solving fuel my growth.