

A
PROJECT REPORT
ON
Online Doctor's Appointment System

SUBMITTED TO THE SHIVAJI UNIVERSITY, KOLHAPUR
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENT
FOR THE AWARD OF DEGREE
OF

BACHELOR OF TECHNOLOGY
in
Computer Science and Engineering

SUBMITTED BY

Mr. Sarang S. Kadam [214] [20UGCS19604]

Mr. Atharv V. Gotkhindikar [243] [20UGCS19815]

Mr. Vikas R. Kole [246] [20UGCS19910]

PROJECT GROUP NO: C21

UNDER THE GUIDANCE OF

Prof. I. I. Tamboli



Department of Computer Science and Engineering
SWVSM's

**Tatyasaheb Kore Institute of Engineering and
Technology (Autonomous), Warananagar**
Academic Year 2023-24



SWVSM's
**Tatyasaheb Kore Institute of Engineering and
Technology (Autonomous), Warananagar**

CERTIFICATE

This is to certify that the Project Report entitled,

“Online Doctor’s Appointment System”

Submitted by

Name	Exam Seat No.
Mr. Sarang S. Kadam	[20UGCS19604]
Mr. Atharv V. Gotkhindikar	[20UGCS19815]
Mr. Vikas R. Kole	[20UGCS19910]

is a bonafide work carried out and is approved for the partial fulfillment of the requirement of Shivaji University, Kolhapur for the award of Degree of Bachelor of Technology in Computer Science and Engineering. This project work is a record of students' own work, carried out by them under our supervision and guidance during academic year 2023-24.

Prof. I. I. Tamboli
Project Guide

Prof. R. B. Patil
HoD, CSE

Dr. D. N. Mane
Principal

Date: -
Place: - Warananagar

External Examiner

ACKNOWLEDGEMENT

This space is dedicated to acknowledge all those who have helped in bringing this project to fruition.

We are greatly indebted to our guide **Prof. I. I. Tamboli**, for his unstinted support and valuable suggestions. We are grateful to him not only for the guidance, but also for his unending patience and keeping our spirits high throughout. We express our sincere thanks to our beloved Head of the Department, **Prof. R. B. Patil** and Principal, **Dr. D. N. Mane** for being source of inspiration and providing us the opportunity to work on this project.

We extend heartfelt thanks to all the teaching and non-teaching staff of the department of Computer Science and Engineering for their assistance and cooperation.

Finally, we would like to thank our parents and friends for their moral support and encouragement throughout our academics.

Student Name	Exam Seat No.	Sign
Mr. Sarang S. Kadam	[20UGCS19604]	
Mr. Atharv V. Gotkhindikar	[20UGCS19815]	
Mr. Vikas R. Kole	[20UGCS19910]	

Date: -

Place: - Warananagar.

ABSTRACT

The Online Doctor's Appointment System is a comprehensive solution designed to streamline and enhance the healthcare booking process. This project aims to provide a user-friendly platform where patients can conveniently schedule appointments with healthcare professionals online. The system facilitates efficient communication between patients and doctors, reducing waiting times and enhancing overall healthcare accessibility. Features include real-time availability updates, automated reminders, and secure patient data management. By leveraging technology, this system optimizes the scheduling process, fostering a more patient centric approach to healthcare services. Ultimately, the Online Doctor's Appointment System represents a pivotal step towards improving healthcare efficiency and patient experience in the digital era.

TABLE OF CONTENTS

CHAPTER	NAME	PAGE
Certificate		
Acknowledgement I		
Abstract II		
List of Figures III		
List of Tables IV		
List of Screenshot V		
1	Introduction	1 - 3
1.1	Introduction	1
1.2	Motivation	1
1.3	Purpose	1
1.4	Problem Statement	3
1.5	Object	3
2	Literature Survey	4 - 11
2.1	Existing System	4
2.2	Proposed System with block diagram	8
2.3	Feasibility Study	10
3	Project Scope and Requirement Analysis	12 - 14
3.1	Project Scope	12
3.2	Requirement Gathering & Analysis	13
4	Project Design and Modelling Details	15 - 31
4.1	Software Requirement Specification (SRS)	15
4.2	System Modules	17
4.3	System Modelling & Design	20
4.4	Database Design	26
4.5	System Architecture	31
5	Implementation and Coding	32 - 37
5.1	Algorithm	32
5.2	Flowchart	35
5.3	Software & hardware requirement with	36

	justification	
6.	Testing	38 - 42
6.1	Fundamental of testing	38
6.2	Test plan of project	39
6.3	Test case & Test Result	40
7.	Project Plan & Schedule	43 - 48
7.1	Project planning & project resources	43
7.2	Project scheduling (ghant chart)	45
7.3	Effort estimation	46
8.	Risk Management and Analysis	49 - 50
8.1	Project risk identification	49
8.2	Project risk analysis	50
9.	Configuration Management	51 - 95
9.1	Installation & Uninstallation	51
9.2	User manual	53
10	Conclusion and Future Scope	96 - 99
10.1	Conclusion	96
10.2	Future scope	97

References

LIST OF FIGURES

Figure No	Figure Name	Page No.
Fig. 2.2.1	Block Diagram	8
Fig. 4.3.1	Use Case Diagram	20
Fig. 4.3.2	Class Diagram	22
Fig. 4.3.3	Activity Diagram	23
Fig.4.3.4	Sequence Diagram	25
Fig. 4.4.1	ER Diagram	26
Fig. 4.4.2.1	DFD's Diagram (level0)	28
Fig. 4.4.2.2	DFD's Diagram (level1)	29
Fig. 4.4.2.3	DFD's Diagram (level2)	30
Fig. 5.2.1	Flowchart	35

LIST OF TABLES

Table No.	Table Name	Page No.
Table 6.3.1	Test Case 1	40
Table 6.3.2	Test Case 2	41
Table 6.3.3	Test Case 3	41
Table 6.3.4	Test Case 4	42
Table 6.3.5	Test Case 5	42
Table 7.2.1	Ghant Chart	45
Table 8.1.1	Project Risk Management	50

LIST OF SCREENSHOTS

Screenshot no.	Screenshot Name	Page No.
Screenshot 9.2.0	Home Dashboard Page	55
Screenshot 9.2.1.1	Login Page	56
Screenshot 9.2.1.2	Administer Dashboard	57
Screenshot 9.2.1.3	Doctor's Activity (Administer Panel)	58
Screenshot 9.2.1.4	Doctor's Activity (Add New Doctor)	59
Screenshot 9.2.1.5	Doctor's Activity (Edit Doctor Details)	60
Screenshot 9.2.1.6	Doctor's Activity (View Doctor Details)	61
Screenshot 9.2.1.7	Doctor's Activity (Delete Doctor Details)	62
Screenshot 9.2.1.8	Schedule Activity (Schedule Manager)	63
Screenshot 9.2.1.9	Schedule Activity (Add New Section)	64
Screenshot 9.2.1.10	Schedule Activity (Search for Schedule)	65
Screenshot 9.2.1.11	Schedule Activity (View Doctor Schedule)	66
Screenshot 9.2.1.12	Schedule Activity (Record Delete Session)	67
Screenshot 9.2.1.13	Appointment Activity (Appointment Manager)	68
Screenshot 9.2.1.14	Appointment Activity (Delete Appointment)	69
Screenshot 9.2.1.15	Patient Activity (List of Patient)	70
Screenshot 9.2.1.16	Patient Activity (View of Patient Details)	71
Screenshot 9.2.1.1	Doctor's Panel (Login Page)	72
Screenshot 9.2.2.2	Doctor Dashboard	73
Screenshot 9.2.2.3	My Appointment Activity (Appointment Manager)	74
Screenshot 9.2.2.4	Appointment Manager (Delete Appointment)	75
Screenshot 9.2.2.5	My Session Activity (Session Manager)	76
Screenshot 9.2.2.6	My Session Activity (View Session)	77
Screenshot 9.2.2.7	My Session Activity (Delete Session)	78
Screenshot 9.2.2.8	My Patients Activity (My Patient)	79
Screenshot 9.2.2.9	My Patients Activity (View Patient Details)	80
Screenshot 9.2.2.10	Settings Activity	81

Screenshot 9.2.2.11	Settings Activity (Edit Doctor Details)	82
Screenshot 9.2.2.12	Settings Activity (View Doctor Details)	83
Screenshot 9.2.2.13	Settings Activity (Delete Account)	84
Screenshot 9.2.3.1	Patient Panel (Sign Up Page)	85
Screenshot 9.2.3.2	Create User Account	86
Screenshot 9.2.3.3	Patient Dashboard	87
Screenshot 9.2.3.4	All Doctor's Activity (List of Doctor's)	88
Screenshot 9.2.3.5	Scheduled Session Activity (All Session)	89
Screenshot 9.2.3.6	Scheduled Session Activity (Session Details)	90
Screenshot 9.2.3.7	My Booking Activity (Booking History)	91
Screenshot 9.2.3.8	My Booking Activity (Cancel Booking)	92
Screenshot 9.2.3.9	Setting Activity	93
Screenshot 9.2.3.10	Setting Activity (Account Setting)	94
Screenshot 9.2.3.11	Setting Activity (View Account Details)	95
Screenshot 9.2.3.12	Setting Activity (Delete Account Details)	96

Chapter 1

Introduction

1.1 Introduction

The introduction of the Online Doctor's Appointment System seeks to revolutionize healthcare accessibility. This project aims to create a user-friendly platform facilitating seamless scheduling between patients and doctors. By leveraging technology, it optimizes the appointment process, minimizing wait times, enhancing communication and ultimately improving the efficiency and convenience of healthcare services in the digital age.

1.2 Motivation

The motivation behind the Online Doctor's Appointment System stems from a commitment to address healthcare inefficiencies. This project is driven by the desire to streamline the appointment process, reduce wait times and enhance patient-doctor interactions. By leveraging digital solutions, it aspires to create a more accessible and patient centric healthcare experience, fostering efficiency and convenience in the medical domain

1.3 Purpose:

The purpose of an online doctor's appointment system is multifaceted, aiming to improve the efficiency, accessibility, and quality of healthcare services for both patients and healthcare providers. Below are key purposes and benefits of implementing such a system:

Convenience for Patients:

- **24/7 Access:** Patients can schedule appointments at any time, from anywhere, using a web browser or mobile app, eliminating the need for phone calls or in-person visits.
- **Ease of Use:** User-friendly interfaces make it simple for patients to search for doctors, view availability, and book appointments with just a few clicks.

Improved Access to Healthcare:

- Broader Choice: Patients can browse through a diverse range of doctors, specialists, and healthcare facilities, expanding their options beyond local providers.
- Reduced Wait Times: Online appointment systems can reduce waiting times by allowing patients to schedule appointments efficiently, minimizing overcrowding and delays at clinics.

Efficiency and Streamlined Operations:

- Automated Booking: Reduces administrative burden on healthcare staff by automating appointment scheduling, reducing manual errors, and optimizing resource allocation.
- Real-Time Updates: Enables doctors to manage their schedules more effectively and stay informed about new appointments and cancellations.

Enhanced Patient Experience:

- Appointment Reminders: Automated reminders via email or SMS help patients remember their appointments, reducing no-show rates and optimizing clinic efficiency.
- Transparent Information: Patients can access detailed information about doctors, including qualifications, specialties, and patient reviews, facilitating informed decision-making.

Optimized Resource Utilization:

- Time Management: Efficient appointment scheduling minimizes idle time for healthcare providers and ensures that resources are utilized effectively throughout the day.
- Optimal Patient Flow: Helps healthcare facilities manage patient flow, reducing overcrowding during peak hours and improving overall patient experience.

1.4 Problem Statement:

Traditional doctor appointment scheduling is inconvenient and inefficient. Patients lack information, face scheduling conflicts, and healthcare facilities struggle with paper-based records. Our system aims to design an efficient online doctor appointment system to streamline healthcare access to patients and emergency patients also.

1.5 Objective:

The project's objectives center on transforming healthcare accessibility. It aims to create an intuitive Online Doctor's Appointment System to streamline scheduling, reduce waiting times, and foster efficient patient-doctor communication. By enhancing these aspects, the project seeks to optimize healthcare services, ultimately improving patient experiences and contributing to a more accessible and responsive healthcare system.

Chapter 2

Literature Survey

2.1 Existing System

The existing system study reveals limitations in conventional healthcare appointment methods, characterized by manual booking processes and communication gaps.

- **User Roles and Authentication:**

- Identify the different user roles involved, such as patients, doctors and administrators.
- Explore how user authentication and authorization are managed to ensure data security and privacy.

- **Appointment Scheduling:**

- Examine how users can schedule appointments with doctors.
- Look into features like selecting a preferred time slot, viewing doctor availability, receiving confirmation notifications.

- **Doctor Profiles:**

- Study how doctor profiles are presented, including information like specialization, experience and availability.
- Check if there are features allowing patients to leave reviews or ratings for doctors.

- **Notification System:**

- Analyze the notification mechanism for confirming, rescheduling or cancelling appointments.
- Understand how reminders are sent to both doctors and patients.

- **Patient Records and History:**

- Investigate how patient records are managed, including medical history, prescriptions and test results.
 - Ensure that there are adequate privacy measures in place for handling sensitive medical information.
- Some other aspects like Billing and Payment, Feedback & Rating System, Security measures, Customer support.

2.1.1 Referred Journal/Conference Papers –

[1] Dependable online appointment booking system for NHIS outpatient in Nigerian teaching hospitals by Adebayo Peter Idowu, Olajide Olusegun Adeosun, Kehinde Oladipo Williams -In this fast-driven society, where the climate in the healthcare sector demands efficiency and patient's satisfaction and medical care delivery.

2.1.2 Web Reference –

[1] Apache Official Website: <https://httpd.apache.org/>

[2] MySQL official Website: <https://www.mysql.com/>

[3] Visual Studio Code Official Website: <https://code.visualstudio.com/>

[4] Microsoft's Web Development Documentation:

<https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-5.0>

2.1.3 Elaborate on Existing System Applications / Examples

An online doctor's appointment system is a digital platform designed to facilitate the booking and management of medical appointments over the internet. These systems streamline the process of scheduling appointments, allowing patients to book consultations with healthcare providers conveniently and efficiently. Below are some examples and elaborations on existing applications of online doctor's appointment systems:

- **Zocdoc:** Zocdoc is a popular online platform that allows patients to find and book appointments with doctors and healthcare providers. Users can search for specific specialties, view doctor profiles, and instantly book appointments based on availability. Zocdoc also offers features like patient reviews, insurance verification, and appointment reminders.
- **Practo:** Practo is another comprehensive healthcare platform that includes an online appointment booking system. It allows patients to search for doctors by specialty, read reviews, and book appointments online. Practo also offers telemedicine services where patients can consult with doctors remotely.

- **Healthgrades:** **Healthgrades** is a platform that provides information about healthcare providers and allows patients to schedule appointments online. Patients can search for doctors, view their credentials and patient reviews, and book appointments directly through the platform.
- **Doctolib:** **Doctolib** is a European-based online healthcare booking platform. It enables patients to find and book appointments with doctors, dentists, and other healthcare professionals. Doctolib also offers features like online video consultations and medical records management.
- **Curofy:** **Curofy** is an online doctor networking platform that also facilitates appointment scheduling. While primarily focused on connecting doctors and enabling professional networking, Curofy also offers appointment booking features for patients.
- **BookMyDoctor:** **BookMyDoctor** is an online doctor appointment scheduling system used in various regions. It allows patients to search for doctors based on location, specialty, and availability, and book appointments directly through the website or mobile app.

2.1.4 Limitations or Challenges in Existing System

While online doctor appointment systems offer numerous benefits, they also face certain limitations and challenges that can impact their effectiveness and usability. Here are some common limitations and challenges associated with existing online doctor appointment systems:

- **User Adoption and Accessibility Issues:**
 - Digital Divide: Not all patients have easy access to the internet or may struggle with using online platforms, particularly elderly or underserved populations.
 - Technological Barriers: Some users may encounter difficulties navigating or using the online appointment system, especially if they are not familiar with technology.
- **Data Privacy and Security Concerns:**
 - Patient Data Protection: Handling sensitive patient information requires robust security measures to prevent data breaches or unauthorized access.
 - Compliance: Systems must adhere to strict healthcare data privacy laws and regulations (like HIPAA in the US) to protect patient confidentiality.
- **Appointment Scheduling Complexity:**
 - Limited Availability: Some doctors may have limited availability or fluctuating schedules, making it challenging for patients to find suitable appointment times.
 - Emergency Situations: Online systems may not adequately handle urgent or emergency appointments, requiring alternative contact methods.

- **Integration with Healthcare Ecosystem:**

- Compatibility Issues: Integration with existing hospital management systems, electronic health records (EHR), or insurance networks can be complex and may not always be seamless.
- Interoperability: Lack of interoperability between different healthcare IT systems can hinder the effectiveness of online appointment systems.

- **Quality and Reliability of Information:**

- Doctor Profiles: Information provided on doctor profiles (like qualifications, experience, and patient reviews) may not always be accurate or up to date, impacting patient trust and decision-making.
- Service Consistency: Variability in service quality among healthcare providers can affect patient satisfaction and trust in the system.

- **Communication and Follow-Up:**

- Lack of Personal Interaction: Online systems may reduce direct patient-provider communication, potentially affecting the quality of care and patient outcomes.
- Follow-Up Care: Ensuring continuity of care and follow-up appointments through online platforms can be challenging.

- **Technical and Maintenance Issues:**

- System Downtime: Technical glitches, maintenance, or server outages can disrupt service availability, affecting patient access to appointments.
- Software Updates: Regular updates and maintenance are required to keep the system secure and functioning optimally.

- **Health Equity and Inclusivity:**

- Digital Divide: Online appointment systems may exacerbate existing disparities in healthcare access, particularly for marginalized or underserved populations who have limited internet access or digital literacy.

2.2 Proposed System with block diagram

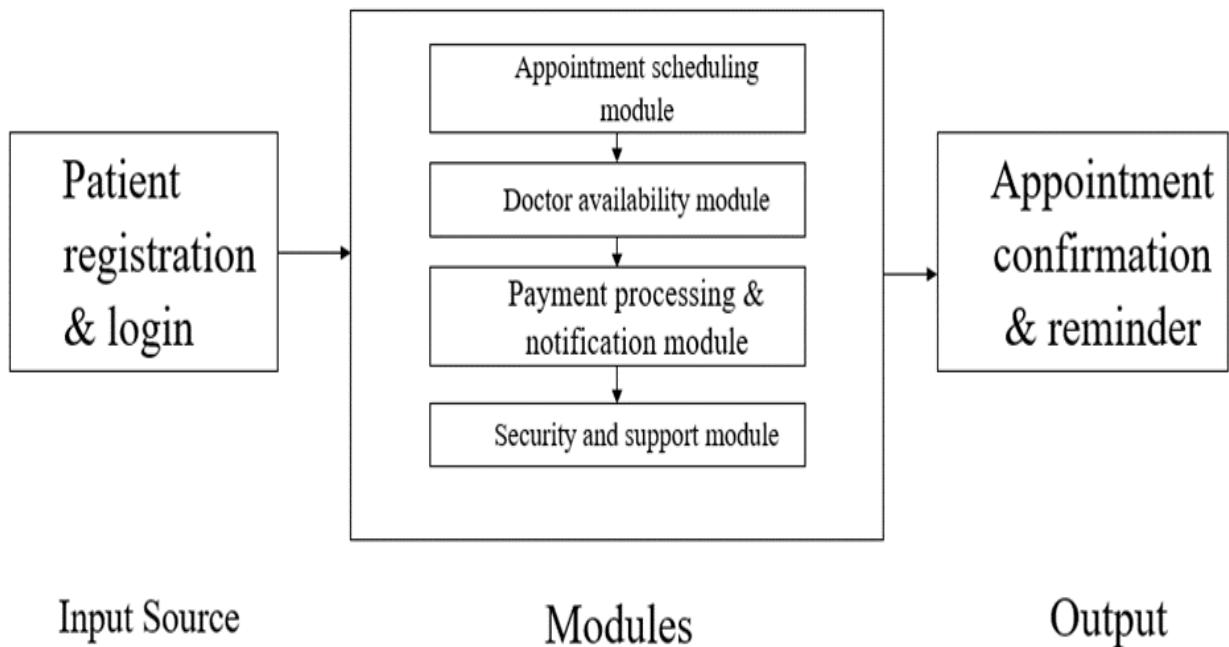


Fig:2.2.1 Block Diagram.

Components:

- User Interface:
 - Patient Interface: Allows patients to register, log in, search for doctors, view profiles, and book appointments.
 - Doctor Interface: Provides doctors with tools to manage their schedules, view appointments, and communicate with patients.
- Authentication and Authorization:
 - Manages user authentication (login) and authorization (access control) to ensure secure access to the system.
- Database:
 - Patient Database: Stores patient information including profiles, medical history, and appointment records.
 - Doctor Database: Contains details of doctors, their specialties, availability, and appointment schedules.
 - Appointment Database: Stores information about scheduled appointments, including patient-doctor relationships and time slots.

- Appointment Booking Module:
 - Handles the process of searching for available doctors based on criteria (e.g., specialty, location, availability) and booking appointments.
 - Manages real-time updates to doctor availability and appointment slots.
- Notification System:
 - Sends appointment reminders, notifications of booking confirmations, and other relevant communications to patients and doctors via email, SMS, or app notifications.
- Feedback and Review Module:
 - Allows patients to provide feedback and reviews about their experiences with doctors, helping to improve service quality and inform other patients' decisions.
- Administrative Dashboard:
 - Provides administrative tools for system management, user management, analytics, and reporting.

System Workflow:

- User Registration and Login:
 - Patients and doctors register and authenticate into the system using credentials.
- Doctor Search and Appointment Booking:
 - Patients search for doctors based on specialty, location, or availability.
 - Patients select a suitable time slot and book appointments with preferred doctors.
- Appointment Confirmation and Notifications:
 - Patients receive confirmation of booked appointments via email or SMS.
 - Doctors are notified of new appointments and can view their schedules through the doctor interface.
- Feedback and Review Submission:
 - After appointments, patients provide feedback and reviews about their experiences with doctors.
- Administrative Management:
 - Administrators manage system settings, user accounts, database maintenance, and generate reports for analysis.

Advantages and Features:

- Convenience: Enables easy and efficient booking of doctor appointments from anywhere.
- Accessibility: Provides access to a wide range of doctors and specialties.
- Transparency: Offers transparent information about doctors' profiles, availability, and patient reviews.

- Efficiency: Streamlines appointment scheduling and reduces administrative workload.
- Improved Patient Experience: Enhances patient satisfaction through better access to healthcare services and personalized care.

2.3 Feasibility Study:

The feasibility study explores the practicality of implementing an online Doctor's Appointment System. Evaluating technical, operational and economic aspects it aims to ensure that the proposed system is viable and sustainable.

- **Legal Feasibility:**

Compliance:

Ensure that the proposed system complies with healthcare regulations and data protection laws. Consider the legal implications of handling sensitive patient information and implement necessary measures for compliance.

- **Technical Feasibility:**

- System Requirements:

Assess the technical requirements for developing and maintaining the system. Ensure that the necessary hardware, software, network infrastructure is available or can be acquired.

- Technology Stack:

Consider factors such as scalability, security, compatibility with existing systems.

- **Operational Feasibility:**

- User Acceptance:

Gauge the willingness of users (patients, doctors, administrators) to adopt the new system. Conduct surveys or interviews to understand user preferences and potential resistance.

- Training Needs:

Identify training requirements for users to ensure they can effectively navigate and use the system. and user-friendly design.

- **Economic Feasibility:**

- Cost-Benefit Analysis:

Conduct a thorough cost-benefit analysis to determine if the benefits of the online appointment system outweigh the associated costs. Consider development costs, maintenance expenses, and potential revenue streams.

- **Schedule Feasibility:**

- Timeline:

- Develop a realistic project timeline that includes key milestones and deadlines.

- Consider factors such as development, testing, and implementation phases.

- Dependencies:

- Identify potential dependencies that could impact the project schedule. Mitigate risks associated with dependencies to ensure timely completion.

- **Social Feasibility:**

- User Engagement:

- Assess the level of social acceptance and support for an online doctor's appointment system within the community.

- **Environmental Feasibility:**

- Resource Availability:

- Evaluate the availability of resources such as skilled personnel, infrastructure, and support services.

- Environmental Impact:

- Consider the environmental impact of the system, such as energy consumption and sustainability.

- **Risk Analysis:**

- Identify Risks:

- Identify potential risks and challenges that could impact the success of the project. This includes technical, operational, economic, and regulatory risks.

- Risk Mitigation:

- Develop strategies to mitigate and manage identified risks. Create contingency plans to address unexpected challenges.

Chapter 3

Project Scope and Requirement Analysis

3.1 Project Scope:

The project scope encompasses the development of an Online Doctor's Appointment System, aiming to create a user-friendly platform for seamless scheduling between patients and doctors. The requirement analysis will delve into essential features, ensuring the system's capability to optimize healthcare appointments, enhance communication, and provide an efficient, accessible, and secure digital environment for medical appointments.

In Scope:

- a. User Registration and Authentication:
Patients, doctors, and admin should be able to register and log in securely.
- b. Doctor Profiles:
Doctors can create profiles with their specialization, availability, and contact information.
- c. Appointment Scheduling:
Patients should be able to search for doctors by specialization, location, or availability. Patients can book appointments with available doctors. Notifications for appointment confirmation and reminders
- d. Integration with payment gateways for appointment fees.
- e. Reviews and Ratings:
Patients can rate and review doctors after appointments.
- f. Technology Stack:
Select appropriate technologies (prog. languages, frameworks, databases) for development
- g. Data Security and Compliance:
Comply with data protection and healthcare regulations.

Out Scope:

- a. User:
Out scope for online doctor appointment system project Final year
- b. Patient Billing and Payment Processing:
Handling financial transactions and insurance claims can be complex and might be excluded from the scope.
- c. Integration with Hospital Systems:
Full integration with hospital information.

3.2 Requirement Gathering & Analysis:

Stakeholders:

- Patients
- Doctors
- administrators

Functional Requirements:

- Patients, doctors, and healthcare facilities must be able to register and log in securely.
- Patients should provide personal information and medical history, while doctors and healthcare facilities should list their qualifications and specialties.
- Users should receive appointment confirmation and reminder notifications.
- Implement encryption for all user data and medical records.
- Users should securely store and manage their electronic medical records.
- Doctors should electronically prescribe medications, with patients receiving prescriptions online or at partner pharmacies.
- Users should be able to make payments securely for appointments and services.

Non-Functional Requirements:

- Ensure data security and privacy through encryption and access controls.
- System responsiveness, even during peak usage times.
- Scalable infrastructure to support system expansion.
- Accessibility features to accommodate users with disabilities.
- Reliable data backup and recovery procedures.
- Compatibility with various devices, browsers, and operating systems.
- Legal and ethical compliance in data handling and patient care.

3.3 Requirement Analysis:

In the requirement analysis phase, the collected user needs and expectations are systematically evaluated and documented for the Online Doctor's Appointment System. This involves defining functional and non-functional requirements, prioritizing features, and specifying system constraints. The goal is to create a clear and comprehensive set of specifications that will guide the development team in building a robust, user-friendly, and effective healthcare scheduling solution.

Chapter 4

Project Design and Modeling Details

4.1 Software Requirement Specification (SRS):

- **Software Requirements:**

The software requirements for the system are as follows:

- **Xampp Server:** A cross-platform web server solution that includes Apache, MySQL, PHP, and Perl components.
- **Web Browsers:** Compatible web browsers such as Google Chrome, Mozilla Firefox, or Microsoft Edge for accessing the system.
- **Apache (2.4.x):** Web server software responsible for serving web content and managing HTTP requests.
- **MySQL (8.x):** Relational database management system (RDBMS) for storing and retrieving data related to patients, doctors, and appointments.
- **Visual Studio 2019:** Integrated development environment (IDE) for developing and testing the system's codebase.
- **Internet Connection:** Required for accessing and interacting with the online doctor appointment system.

- **Hardware Requirements:**

The hardware requirements to run the online doctor appointment system are as follows:

- **PC or Laptop:** Standard computing devices capable of running web-based applications.
- **Processor:** Intel Core i5 or AMD Ryzen 5 processor (or higher) for optimal performance.
- **Memory (RAM):** 8 GB RAM or more to ensure smooth operation of the system.
- **Storage:** Minimum of 512 GB hard disk space to store system files, databases, and user

- **Functional Requirements**

- **User Registration:**

Users (patients and doctors) should be able to register with the system by providing necessary information (e.g., name, email, password).

- **Doctor Search and Appointment Booking:**

Patients should be able to search for doctors based on specialty, location, or availability.

Patients should be able to view doctor profiles, including qualifications and patient reviews.

Patients should be able to book appointments with available doctors.

- **Appointment Management:**

Doctors should have a dashboard to manage their availability, view appointment schedules, and communicate with patients.

Doctors should be able to accept, reschedule, or cancel appointments as needed.

- **Notification System:**

The system should send appointment reminders and confirmations via email or SMS to patients and doctors.

- **Non-Functional Requirements:**

Performance: The system should handle a minimum of 100 concurrent users without significant performance degradation.

Response times for user interactions (e.g., appointment booking) should be within acceptable limits.

- **Security:**

User authentication and authorization mechanisms should be implemented to ensure data privacy and system security.

Patient and doctor information should be stored securely using encryption techniques.

- **Usability:**

The system should have a user-friendly interface that is easy to navigate and intuitive to use for both patients and doctors.

- **Constraints:**

The system development should adhere to applicable healthcare regulations and data protection laws (e.g., HIPAA compliance).

- **Assumptions:**

It is assumed that users have basic computer literacy and internet access to use the online doctor appointment system.

- **Dependencies:**

The system's functionality depends on the proper configuration and integration of Xampp Server, Apache, MySQL, and Visual Studio 2019.

4.2 System Modules

An online doctor appointment system typically consists of several interconnected modules that work together to provide a seamless experience for patients and healthcare providers. Below are the key modules commonly found in such a project:

- **User Management Module:**

- This module handles user registration, login, and authentication.
- Allows patients and doctors to create and manage their profiles.
- Manages user roles and permissions.

- **Doctor Search and Appointment Booking Module:**

- Enables patients to search for doctors based on various criteria such as specialty, location, availability, and ratings.
- Provides detailed doctor profiles showcasing qualifications, experience, availability slots, and patient reviews.

- Allows patients to select a suitable appointment slot and book appointments with preferred doctors.

- **Appointment Management Module:**

- Allows doctors to manage their availability and schedule appointments.
- Provides a calendar view of appointments for doctors.
- Enables doctors to accept, reschedule, or cancel appointments as necessary.
- Sends appointment notifications and reminders to patients and doctors.

- **Patient Dashboard Module:**

- Provides a personalized dashboard for patients after login.
- Displays upcoming appointments, appointment history, and prescription information.
- Allows patients to manage their profile, update information, and view communication with doctors.

- **Doctor Dashboard Module:**

- Provides a personalized dashboard for doctors after login.
- Displays appointment schedules, patient details, and communication history.
- Enables doctors to update availability, manage patient records, and communicate with patients.

- **Notification Module:**

- Handles email and SMS notifications for appointment confirmations, reminders, and updates.
- Notifies patients and doctors about new messages, appointment changes, or other important updates.

- **Feedback and Review Module:**

- Allows patients to provide feedback and ratings for doctors after appointments.
- Displays doctor ratings and reviews to help patients make informed decisions.

- **Admin Dashboard Module:**

- Provides administrative tools to manage the system.
- Allows administrators to manage user accounts, doctor profiles, and system settings.
- Generates reports on appointment statistics, user activity, and system performance.

- **Data Management Module:**

- Handles data storage, retrieval, and manipulation using a relational database (e.g., MySQL).
- Manages patient records, doctor profiles, appointment details, and other system data securely.

- **Security Module:**

- Implements security measures such as user authentication, data encryption, and role-based access control.
- Ensures compliance with data protection regulations (e.g., HIPAA) to safeguard patient information.

4.3 System Modeling & Design

4.3.1: Use Case Diagram:

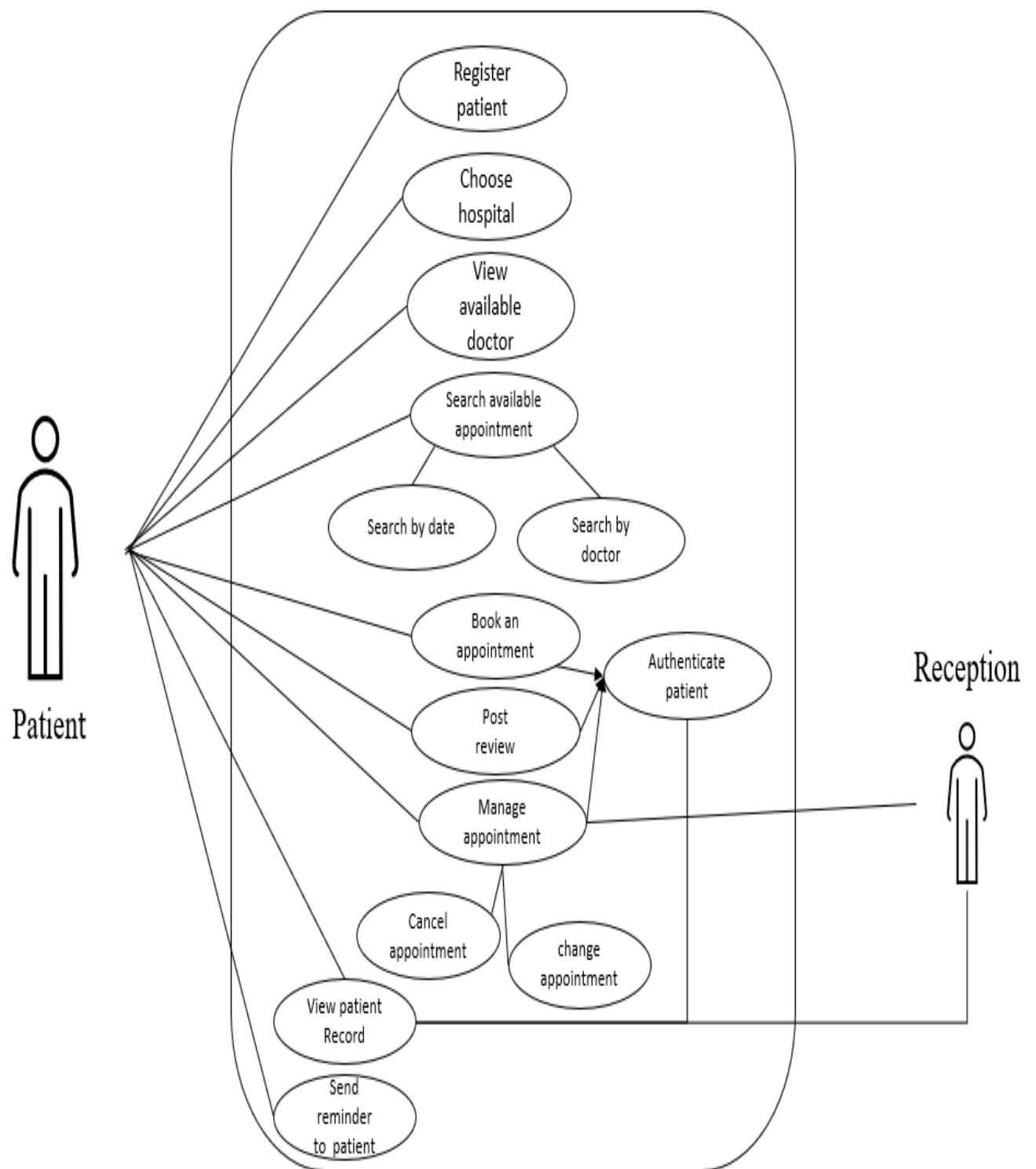


Fig:4.3.1 Use Case Diagram

Actors:

- Patient: Initiates the appointment request.
- Reception: Facilitates the appointment scheduling process.

Use Cases:

- Schedule Appointment:
 - Actor: Patient
 - Description: The patient can request an appointment with a selected doctor.

Manage Availability:

- Actor: Doctor
- Description: The doctor can set and update their availability for appointments.

Review Appointments:

- Actor: Doctor
- Description: The doctor can review and manage appointment requests.

Confirm Appointment:

- Actor: Doctor
- Description: The doctor can confirm or reject appointment requests.

Receive Appointment Confirmation:

- Actor: Patient
- Description: The patient receives confirmation or rejection of their appointment request.

Cancel Appointment:

- Actor: Patient or Doctor
- Description: Either the patient or the doctor can cancel a scheduled appointment.

Generate Prescription:

- Actor: Doctor

- Description: The doctor can generate a prescription for the patient if needed.

Associations:

- Patient schedules an appointment:
 - Between: Patient and Schedule Appointment
- Doctor manages availability:
 - Between: Doctor and Manage Availability
- Doctor reviews appointments:
 - Between: Doctor and Review Appointments
- Doctor confirms appointment:
 - Between: Doctor and Confirm Appointment
- Patient receives confirmation:
 - Between: Receive Appointment Confirmation and Patient
- Patient/Doctor cancels appointment:
 - Between: Cancel Appointment and Patient/Doctor
- Doctor generates a prescription:
 - Between: Doctor and Generate Prescription
- System Boundary:
 - The system boundary encloses all the mentioned actors and use cases, representing the scope of the online doctor's appointment system.

4.3.2: Class Diagram:

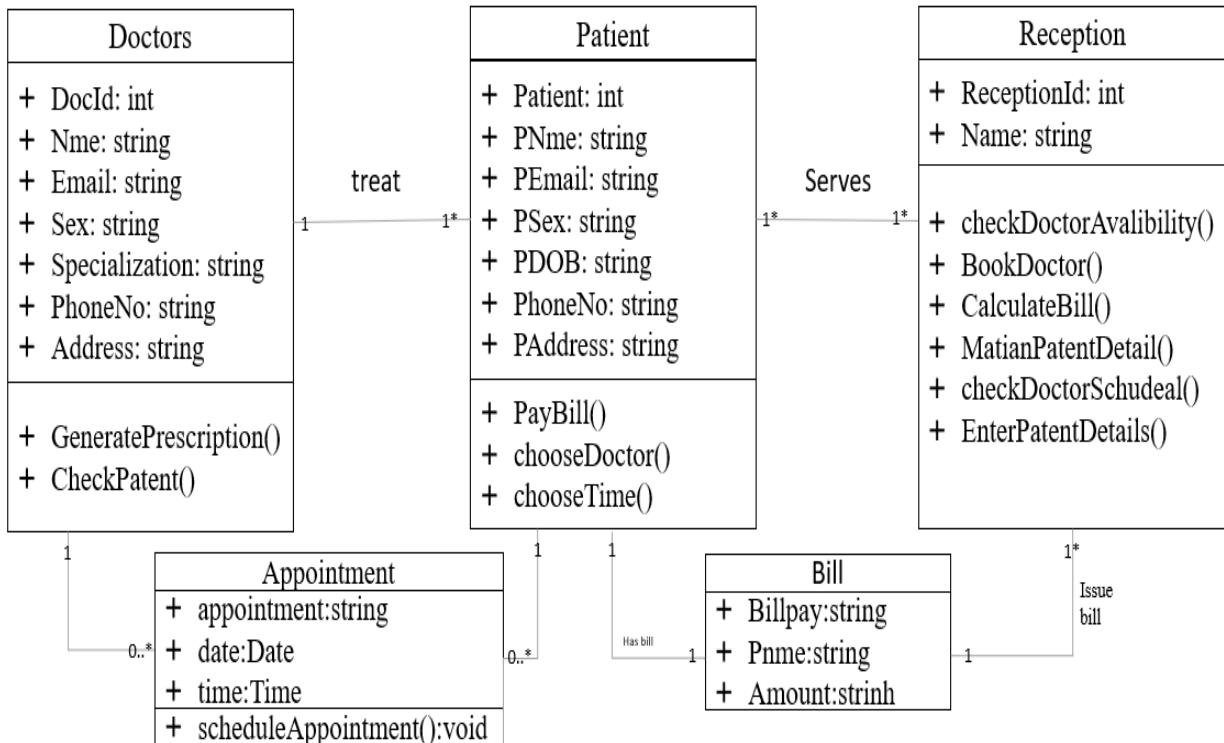


Fig:4.3.2 Class Diagram

- The Doctor class represents a doctor, including details such as the doctor ID, name, specialization, and a list of appointments. It includes methods for accessing and modifying these details, as well as adding and removing appointments.
- The Patient class represents a patient, including details such as the patient ID, name, contact number, and a list of appointments. It includes methods for accessing and modifying these details, as well as adding and removing appointments.
- The Appointment class represents an individual appointment, including details such as the appointment ID, date, status, doctor, and patient. It also provides methods for accessing and modifying these details.
- The reception class represents an individual detail such as ReceptionId, name. It also provides methods for check doctor availability, book doctor, calculate bill, maintain patent schedule, enter patent details.
- The Bill class represent an individual method like billId, billing amount, conform billing payment and etc.

4.3.3: Activity Diagram:

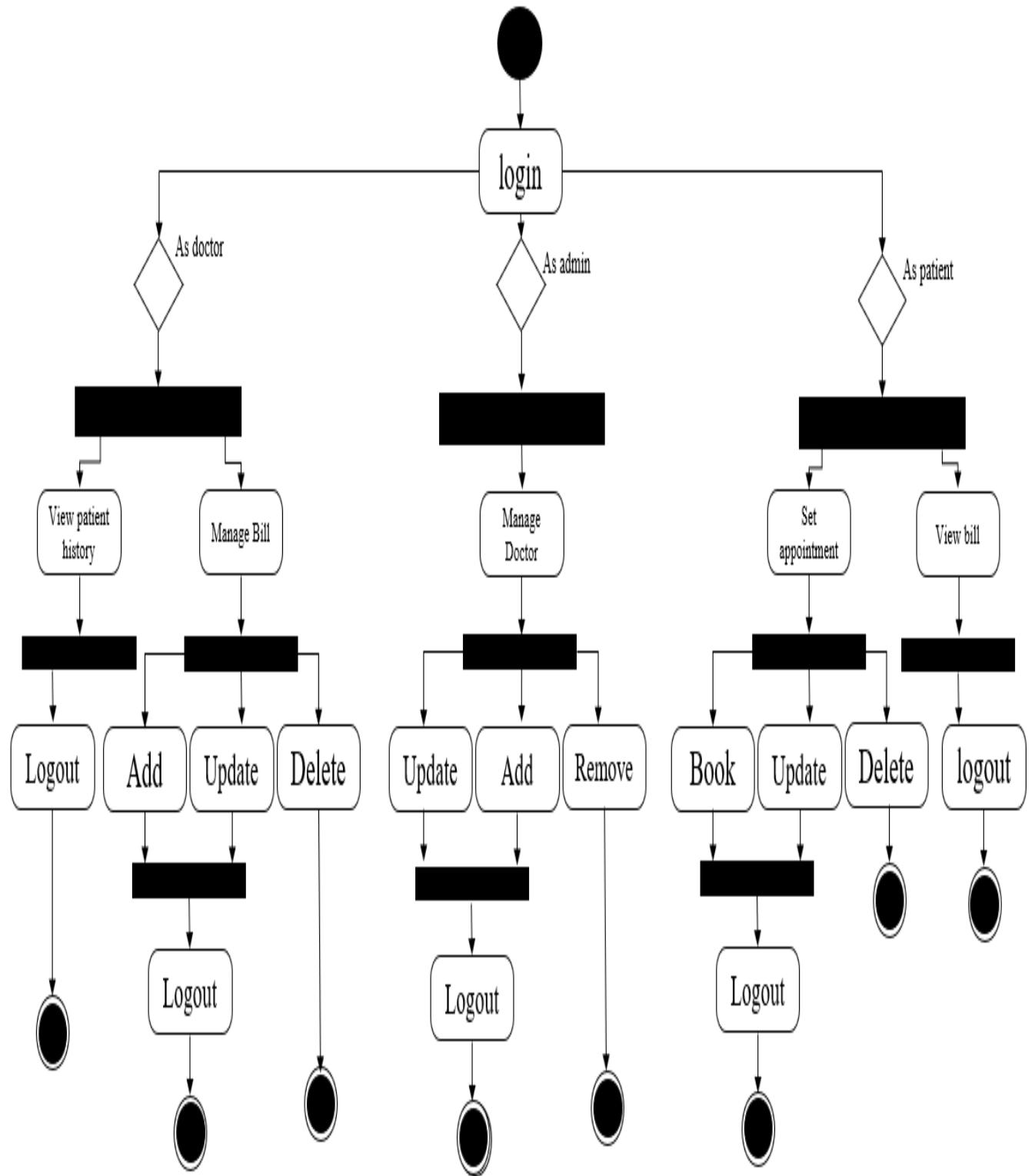


Fig:4.3.3 Activity Diagram

- Start:
 - User initiates the appointment process.
- Login:
 - User logs into the system using credentials.
- View Available Doctors:
 - System retrieves a list of available doctors based on the user's preferences and location.
- Select Doctor:
 - User selects a specific doctor from the list.
- Check Doctor's Availability:
 - System checks the selected doctor's schedule for availability.
- Choose Appointment Slot:
 - User selects a convenient time slot from the available options.
- Book Appointment:
 - System records the user's appointment request.
- Confirmation:
 - System confirms the appointment and provides details.
- Payment (Optional):
 - If required, the system may facilitate online payment for the appointment.
- Generate Receipt:
 - System generates a receipt for the appointment.
- End:
 - Appointment process is completed.

4.3.4: Sequence Diagram:

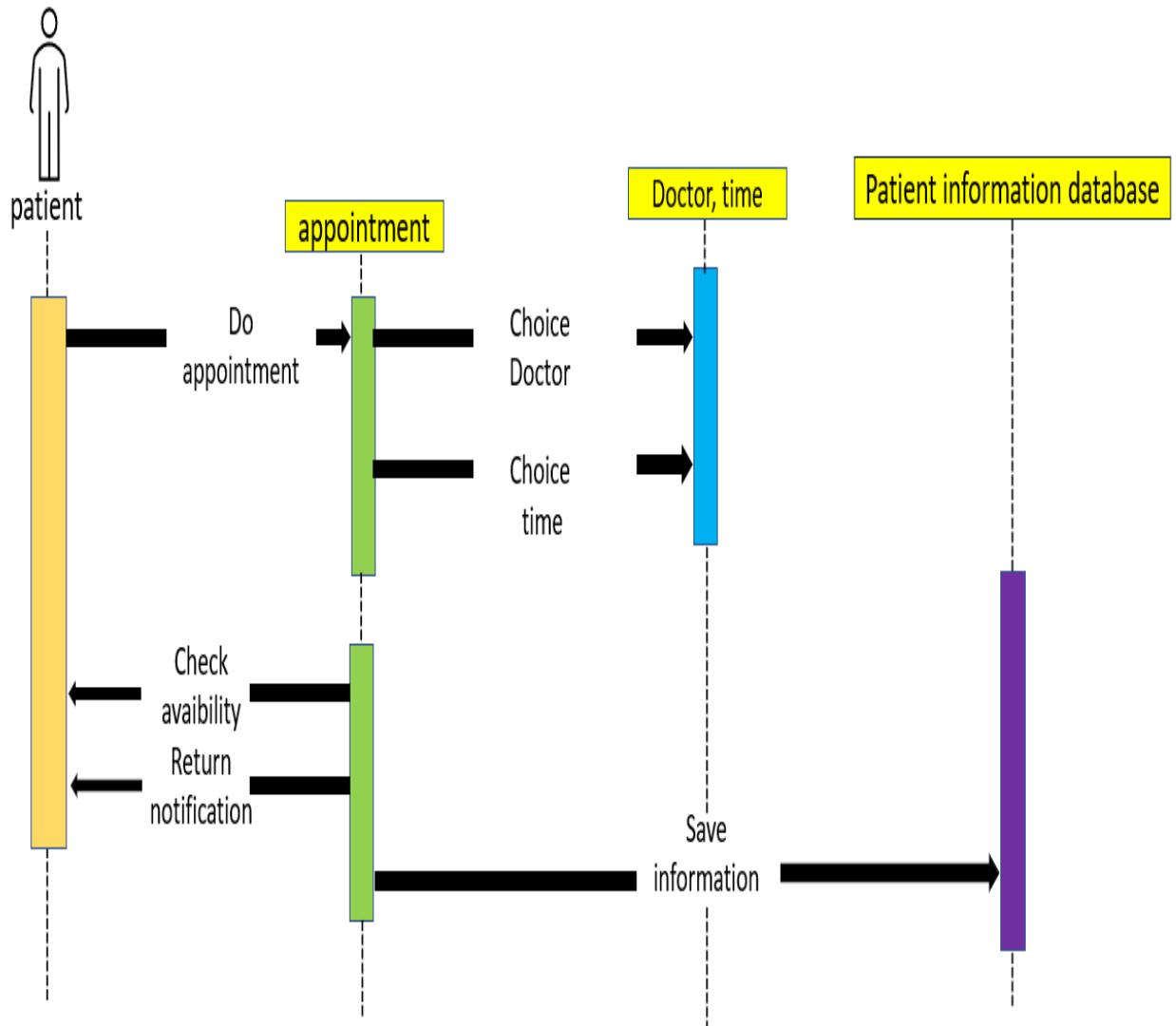


Fig:4.3. Sequence Diagram

- The user interacts with the online appointment system through a web interface.
- The user logs in, searches for a doctor, selects a doctor, chooses an appointment time, and provides necessary details.
- The online appointment system notifies the selected doctor of the appointment.
- The notification service sends a confirmation to the user.
- The online appointment system displays the confirmation to the user.

4.4 Database Design

1) ER Diagram (Doctor Module):

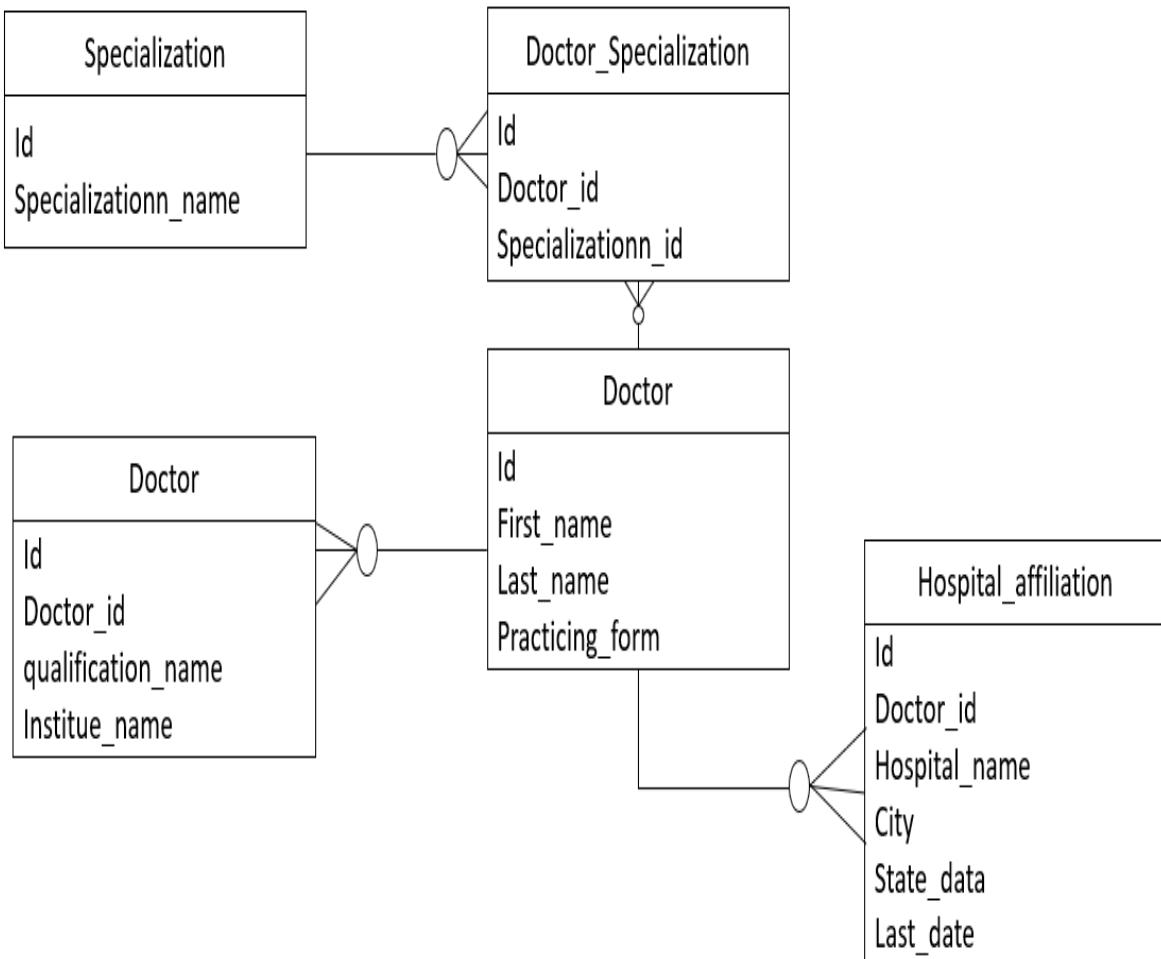


Fig. 4.4.1 ER Diagram (doctor Module)

Entities:

Doctor:

- Attributes: DoctorID (Primary Key), Name, Specialization, ContactNumber, Email, Schedule

Appointment:

- Attributes: AppointmentID (Primary Key), UserID (Foreign Key), DoctorID (Foreign Key), AppointmentDate, AppointmentTime, Status

MedicalRecord:

- Attributes: RecordID (Primary Key), UserID (Foreign Key), DoctorID (Foreign Key), Date, Diagnosis, Prescription, TestResults

Relationships:

- User-Appointment Relationship (Many-to-Many):
 - Users can have multiple appointments.
 - Appointments can have multiple users (e.g., family members sharing an account).
 - Attributes: None (may use a junction table for the many-to-many relationship)
- Doctor-Appointment Relationship (One-to-Many):
 - A doctor can have multiple appointments.
 - An appointment is associated with one doctor.
 - Attributes: None (DoctorID is a foreign key in the Appointment entity)
- User-MedicalRecord Relationship (One-to-Many):
 - A user can have multiple medical records.
 - A medical record is associated with one user.
 - Attributes: None (UserID is a foreign key in the MedicalRecord entity)
- Doctor-MedicalRecord Relationship (One-to-Many):
 - A doctor can be associated with multiple medical records.
 - A medical record is associated with one doctor.
 - Attributes: None (DoctorID is a foreign key in the MedicalRecord entity)

2) DFD's Diagram:

0 Level –

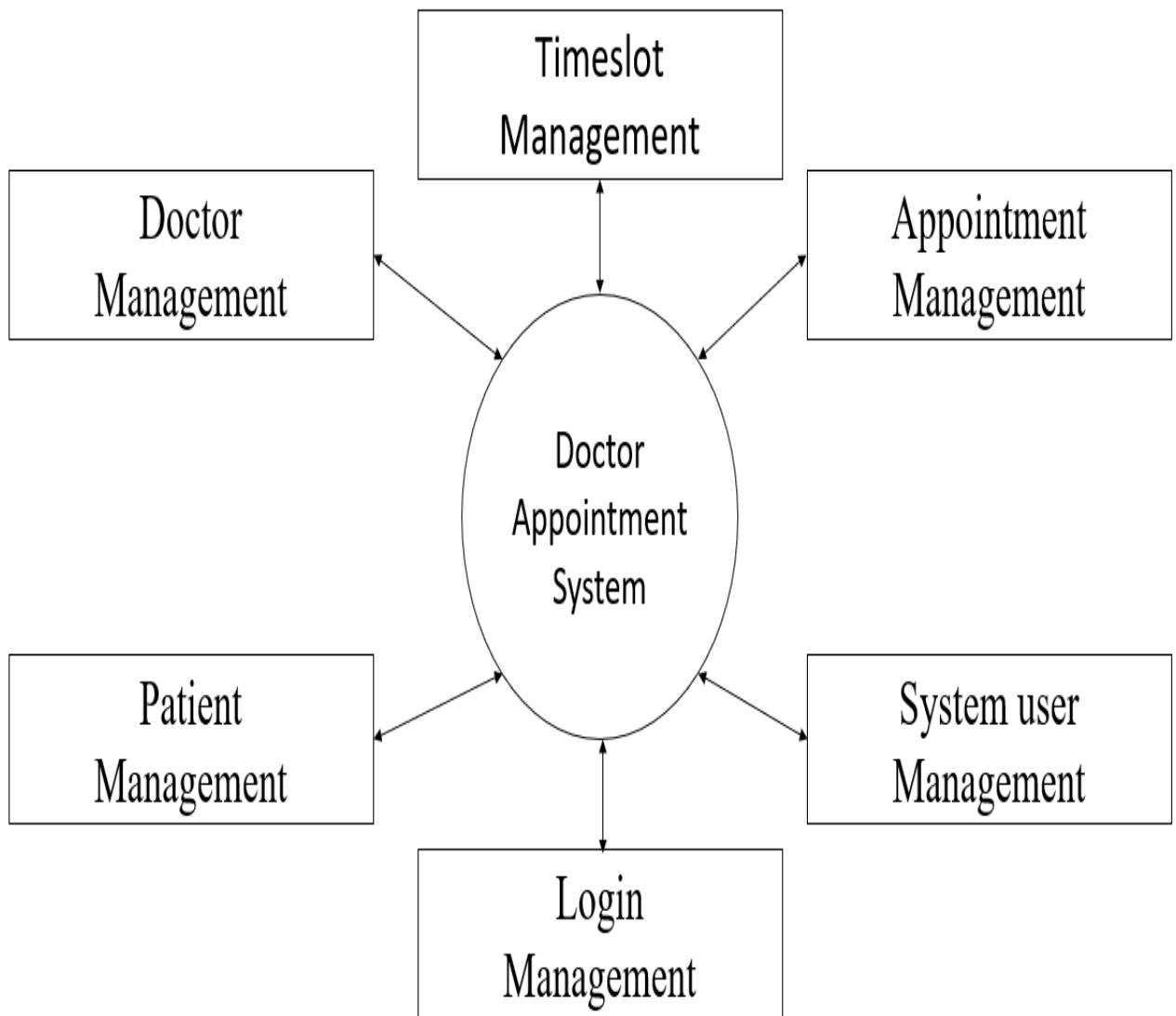
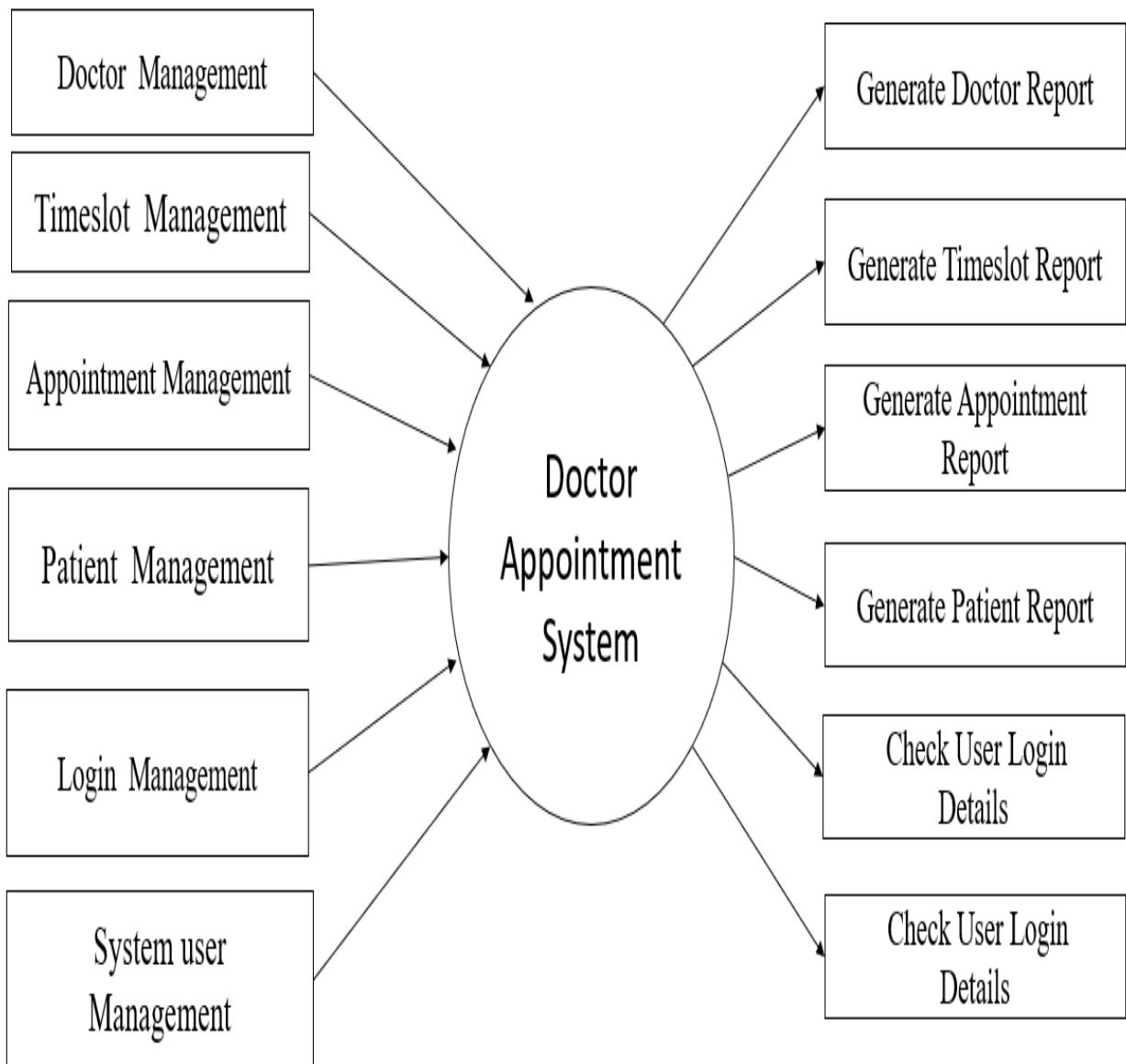
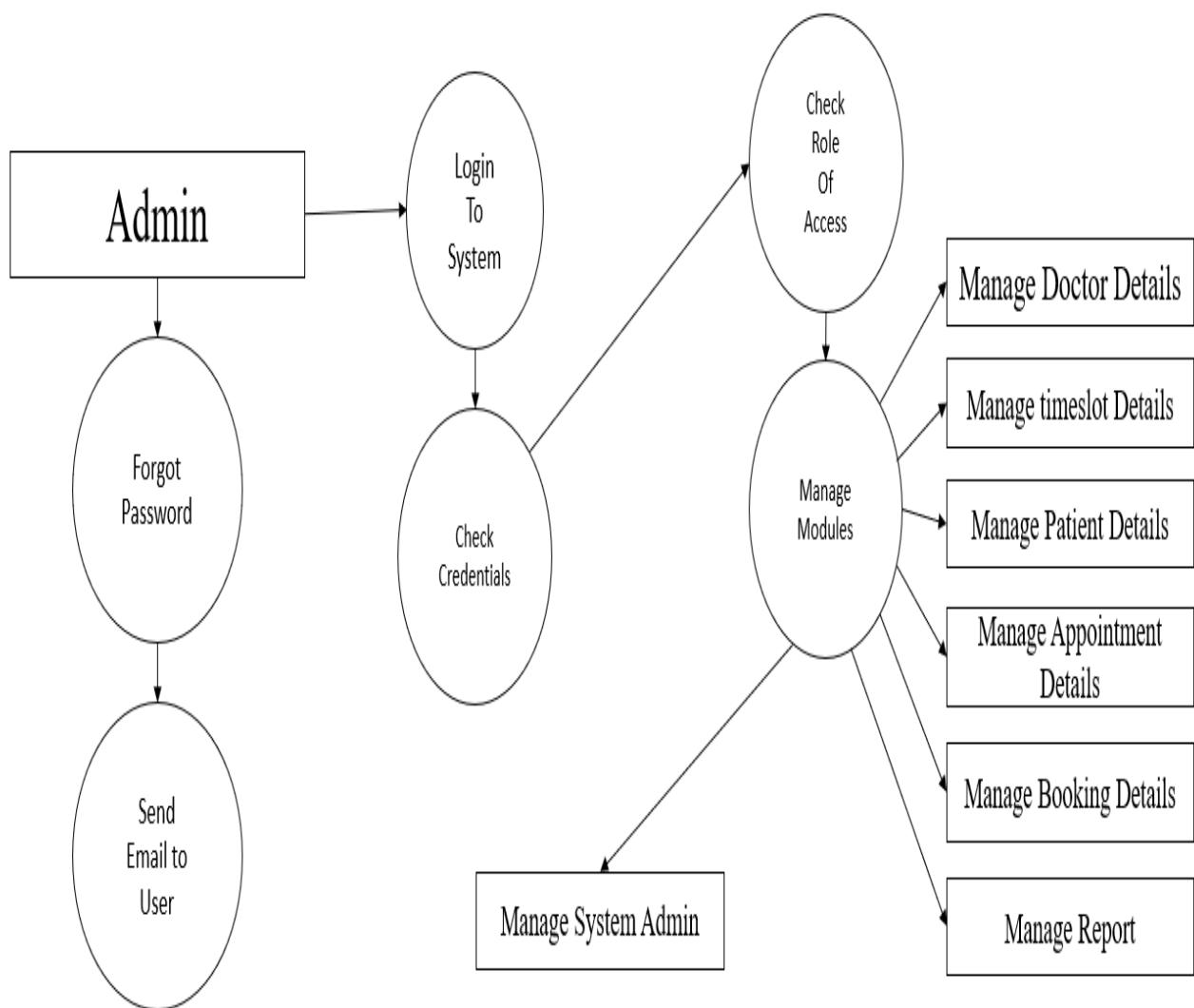


Fig.4.4.2.1 DFD's Diagram

- The "Online Appointment System" is represented as a single process at the center.
- External entities like "Patients," "Doctors," and "Administrators" interact with the system.
- Processes such as "Appointment Scheduling," "User Authentication," and "Notification" are depicted.
- Data storage includes information like "Patient Records" and "Doctor Schedules."

1 Level –**Fig.4.4.2.2 DFD's Diagram level 1****Data Flows:**

- Processing Doctor record and generate of all doctors
- Processing Appointment records and generate report of all appointment
- Processing booking records and generate report of all booking
- Processing timeslot record and generate report of all timeslot
- Processing Patient record and generate report of all patient.
- Processing medicine record and generate report of all medicine

2 Level -**Fig.4.4.2.3 DFD's Diagram level 2****Data Flows:**

- Admin logs in to the system and manages all the functionalities of the doctor appointment system
- Admin can add, delete, and view the records of doctor, booking, timeslot, medicine
- Admin can manage all the details of appointment, patient
- Admin can also generate reports of doctor, appointment, booking, patient, timeslot.

4.5 System Architecture

here are some key points on system architecture for an online doctor appointment system:

- **Client Interface:** Users access the system through various client interfaces such as web browsers, mobile apps (iOS, Android), or desktop applications. These interfaces provide functionalities for users to search for doctors, view their profiles, and book appointments.
- **Server-side Logic:** This component handles the core logic of the system. It manages user authentication, appointment scheduling, doctor availability, and communication between different system components.
- **Database Management:** A database system is crucial for storing user data, doctor profiles, appointment schedules, and other relevant information. It should support efficient retrieval and storage of data to ensure smooth operation of the system.
- **Appointment Scheduler:** This module manages the scheduling of appointments between patients and doctors. It should consider factors such as doctor availability, patient preferences, and clinic hours to optimize appointment scheduling.
- **User Authentication and Authorization:** Security is paramount in any online system. User authentication verifies the identity of users, while authorization ensures that users have the necessary permissions to access certain functionalities or data within the system.
- **Notification System:** An effective notification system keeps users informed about their appointment status, reminders, and any updates or changes to their schedule. It can use email, SMS, or push notifications to reach users.
- **Integration with External Systems:** The system may need to integrate with external systems such as electronic health records (EHR) systems or payment gateways for seamless exchange of information and transactions.
- **Scalability and Performance:** The architecture should be designed to handle a large number of concurrent users and scale effectively as the user base grows. This may involve load balancing, caching, and other performance optimization techniques.
- **Data Privacy and Compliance:** Given the sensitive nature of healthcare data, the system must comply with regulations such as HIPAA (in the US) or GDPR (in the EU). Data encryption, access controls, and regular security audits are essential to maintain data privacy and compliance.
- **Backup and Disaster Recovery:** Regular backups of data and a robust disaster recovery plan are crucial to ensure data integrity and minimize downtime in case of system failures or emergencies.

Chapter 5

Implementation and Coding

5.1 Algorithm

1. User Registration:

- a. When a user accesses the registration page, they provide necessary details such as name, email, password, contact number, etc.
- b. Validate the input data to ensure it meets the required criteria (e.g., valid email format, strong password).
- c. Check if the email provided is unique and not already registered in the system.
- d. If all validations pass, store the user information securely in the database.
- e. Optionally, send a verification email to the provided email address for email confirmation.

2. User Authentication:

- a. When a user attempts to log in, they provide their registered email and password.
- b. Validate the input data and ensure the email exists in the database.
- c. Hash the provided password and compare it with the hashed password stored in the database.
- d. If the passwords match, authenticate the user and generate a session token.
- e. Store the session token in a secure HTTP-only cookie or in local storage (for web applications) or securely manage it in memory (for mobile applications).
- f. Redirect the authenticated user to their dashboard or the intended page.

3. Forgot Password:

- a. Provide a "Forgot Password" functionality where users can request a password reset.
- b. Users enter their registered email address to initiate the password reset process.
- c. Generate a unique token and store it in the database along with the user's email and a timestamp.
- d. Send a password reset link to the user's email address, including the generated token.
- e. When the user clicks on the reset link, verify the token's validity and expiration time.
- f. Allow the user to reset their password by providing a new password.

4. Account Management:

- a. Allow users to update their profile information (name, contact number, etc.) if needed.
- b. Provide an option to change passwords securely, requiring the user to enter their current password for verification.
- c. Implement features for account deletion, ensuring data privacy and compliance with regulations.

5. Doctor Availability Setup:

- a. Doctors log in to their accounts and access the availability management section.
- b. Provide an interface for doctors to set their availability preferences, including:
- c. Days of the week they are available for appointments.
- d. Time slots for each available day, specifying start and end times.
- e. Allow doctors to update their availability dynamically as needed.

6. Availability Validation:

- a. When a patient requests an appointment, verify that the selected doctor is available on the chosen date and time.
- b. Check the doctor's general availability schedule for the selected day.

7. Time Slot Management:

- a. Display these time slots to patients when they request an appointment, indicating which slots are available and which are already booked.
- b. Ensure that time slots respect the doctor's specified working hours and any exceptions.

8. Handling Overlapping Appointments:

- a. Prevent double booking by ensuring that no two appointments overlap for the same doctor.
- b. Implement logic to handle concurrent requests from multiple patients, prioritizing fairness and maintaining an orderly schedule.
- c. If a doctor updates their availability or cancels appointments, reflect these changes in real-time to prevent scheduling conflicts.
- d. Notify patients who have already booked appointments if there are any changes to the doctor's availability that affect their scheduled appointments.

9. Advanced Features:

- a. Implement features for blocking off time slots for administrative tasks, breaks, or emergencies.

10. Notification System:

- a. Notify doctors of new appointment requests and any changes to their availability on the login.

5.2 Flowchart

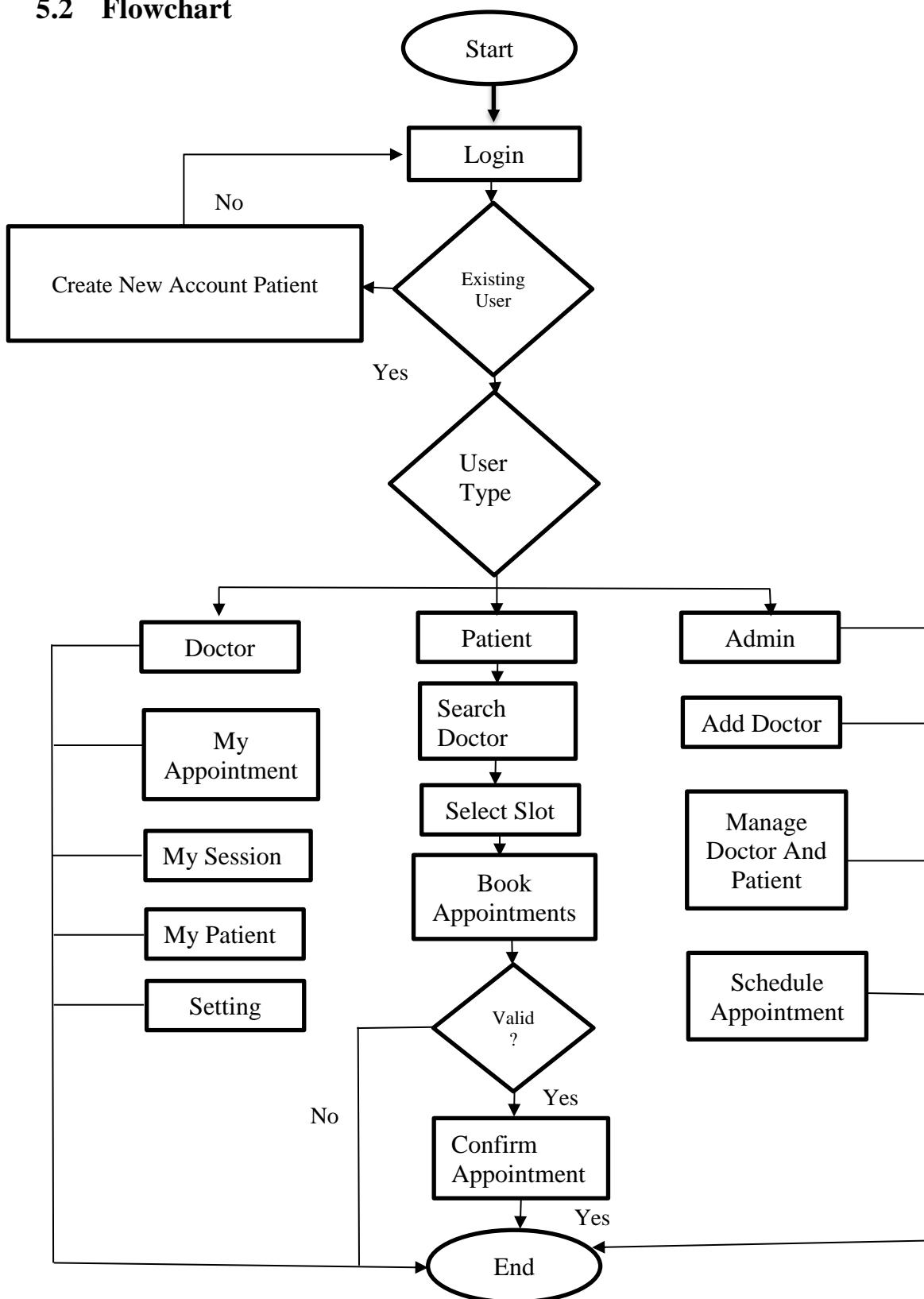


Fig.5.2.1 Flowchart

5.3 Software requirement with relevant justification

- **XAMPP Server:**

- XAMPP is a cross-platform web server solution package that includes Apache, MySQL, PHP, and Perl. It provides a convenient way to set up a local development environment for testing and debugging the online doctor appointment system before deploying it to a production server.

- **Web Browsers:**

- Web browsers are the primary interface through which users interact with the online appointment system. The system should be compatible with popular web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge to ensure broad accessibility for patients and doctors.

- **Apache (2.4.x):**

- Apache HTTP Server is one of the most widely used web servers globally. It is robust, scalable, and well-supported, making it suitable for hosting the online appointment system. Version 2.4.x offers security enhancements, performance improvements, and compatibility with modern web technologies.

- **MySQL (8.x):**

- MySQL is a popular open-source relational database management system (RDBMS) that provides fast, reliable, and scalable data storage. Version 8.x introduces several new features, improvements in performance, security enhancements, and better support for modern SQL standards, making it a suitable choice for storing user data, appointment schedules, and other relevant information.

- **Visual Studio 2019:**

- Visual Studio is a powerful integrated development environment (IDE) that supports multiple programming languages, including C#, JavaScript, HTML, and CSS, commonly used in web development. Visual Studio 2019 provides advanced features such as code debugging, version control integration, and seamless deployment options, facilitating efficient development and maintenance of the online appointment system.

- **Internet Connection:**

- An internet connection is essential for accessing online resources, downloading updates, and communicating with external services or APIs (e.g., email notifications, SMS reminders). It ensures that users can access the online appointment system from anywhere with internet connectivity, enhancing convenience and accessibility.

5.4 Hardware requirement with relevant justification

- **PC/Laptop:**

- A PC or laptop serves as the primary hardware platform for developing, testing, and deploying the online appointment system. It provides the necessary computing resources for running development tools, compiling code, and accessing web-based applications. Additionally, it serves as the interface for doctors and administrative staff to interact with the system.

- **Intel Core i5 or AMD Ryzen 5:**

- The Intel Core i5 and AMD Ryzen 5 processors offer a good balance of performance and affordability, making them suitable choices for running development environments and hosting web applications. These processors provide sufficient processing power for compiling code, running database queries, and handling concurrent user requests without significant performance bottlenecks.

- **8 GB or More RAM:**

- The amount of RAM (Random Access Memory) is crucial for ensuring smooth operation of the online appointment system, especially when handling multiple concurrent user sessions and database operations. With 8 GB or more of RAM, the system can efficiently manage memory-intensive tasks, such as running web servers, database servers, and development environments, without experiencing performance degradation or slowdowns.

- **Hard Disk 512 GB or More:**

- The hard disk space requirement ensures that sufficient storage is available for storing system files, application code, databases, and other data associated with the online appointment system. With a minimum of 512 GB of hard disk space, there is ample room for storing development environments, backups, and user-generated content, while also accommodating future system expansions and updates.

Chapter 6

Testing

6.1 Fundamental of testing

Here are some fundamental aspects of testing for an online doctor appointment system:

- **Functional Testing:** Ensure that all the functionalities of the system work as expected. This includes testing features such as user registration, appointment scheduling, doctor search, appointment cancellation, and user profile management.
- **User Interface (UI) Testing:** Verify that the user interface is intuitive, responsive, and visually appealing across different devices and screen sizes. Test for consistency in design elements, layout, and navigation.
- **Integration Testing:** Test the integration of different system components such as the client interface, server-side logic, database management, and external systems (e.g., payment gateways, EHR systems). Verify that data flows smoothly between these components and that they function together seamlessly.
- **Performance Testing:** Assess the system's performance under various load conditions to ensure it can handle concurrent user traffic effectively. Test for response times, throughput, and resource utilization to identify and address any performance bottlenecks.
- **Security Testing:** Conduct security testing to identify and mitigate potential vulnerabilities in the system. This includes testing for authentication and authorization mechanisms, data encryption, input validation, and protection against common security threats such as SQL injection and cross-site scripting (XSS).
- **Usability Testing:** Evaluate the system from the perspective of end users to ensure it is easy to use and understand. Gather feedback on user experience, accessibility, and any pain points encountered during interactions with the system.
- **Compatibility Testing:** Test the system across different web browsers, operating systems, and devices to ensure compatibility and functionality consistency. Verify that the system works correctly on popular platforms and configurations used by the target audience.
- **Regression Testing:** Perform regression testing to ensure that new updates or modifications to the system do not introduce any unintended side effects or break existing functionalities. Re-run tests on previously validated features to confirm their continued proper functioning.

- **Data Integrity Testing:** Verify the accuracy and consistency of data stored in the system's databases. Test data retrieval, storage, and manipulation operations to ensure that data integrity is maintained throughout the system.
- **Compliance Testing:** Ensure that the system complies with relevant regulations and standards, such as HIPAA (for healthcare data privacy) or GDPR (for data protection). Test for adherence to legal requirements regarding data security, privacy, and user consent.

6.2 Test plan of project

Here are some key points to include in a test plan for the project of an online doctor appointment system:

- **Introduction:** Provide an overview of the online doctor appointment system project, including its objectives, scope, and stakeholders.
- **Testing Objectives:** Clearly define the goals and objectives of the testing phase, such as verifying system functionality, ensuring performance and scalability, and validating security measures.
- **Test Scope:** Outline the scope of testing, including the features, functionalities, and components of the system that will be tested. Specify any areas or functionalities that are out of scope for the current testing phase.
- **Testing Approach:** Describe the overall testing approach to be followed, such as a combination of manual and automated testing. Outline the testing methodologies, techniques, and tools that will be used.
- **Test Environment:** Specify the hardware, software, and network infrastructure required for testing, including details such as operating systems, web browsers, and databases. Ensure that the test environment closely resembles the production environment.
- **Test Cases:** Provide a comprehensive list of test cases covering various scenarios and use cases of the system. Test cases should include steps to reproduce the scenario, expected outcomes, and any preconditions or assumptions.
- **Test Data:** Define the test data required for executing the test cases, including sample user profiles, doctor profiles, appointment schedules, and other relevant data. Ensure that test data adequately represents real-world scenarios and edge cases.
- **Test Execution:** Describe the process for executing test cases, including the sequence of testing activities, responsibilities of testing team members, and any dependencies or prerequisites for testing.

- **Defect Management:** Outline the procedures for reporting, tracking, and managing defects discovered during testing. Define the severity and priority levels for defects, as well as the workflow for defect resolution and retesting.
- **Test Schedule:** Develop a detailed test schedule, including timelines for test planning, test case preparation, test execution, defect resolution, and retesting. Allocate resources and assign responsibilities to ensure timely completion of testing activities.
- **Risks and Mitigation Strategies:** Identify potential risks and challenges that may impact the testing process or the success of the project. Develop mitigation strategies and contingency plans to address these risks and minimize their impact.
- **Acceptance Criteria:** Define the criteria that must be met for the system to be considered ready for production release. This may include passing certain predefined test cases, meeting performance benchmarks, and obtaining stakeholder approval.
- **Documentation:** Specify the documentation deliverables expected as part of the testing phase, such as test plans, test cases, test reports, and any other relevant documentation.
- **Review and Approval:** Outline the process for reviewing and approving the test plan, including the stakeholders involved and the criteria for acceptance.

6.3 Test case and test result (for every model)

Test case 1: -

Module Under Test	User end application (Homepages, sign up and sign in)
Description	A user can see the homepages and get the options for the sign in and sign up or make appointment. User have to do sign up and make appointment through sign in.
Input	Here user is required to enter basic information like name, DoB, mobile No, Email address and create password conform it to do the signup. For sign in into the application enter email and Password.
Output	User will get successfully registered. They can easily login to the application and view the task.
Remark	Test successful.

Table no. 6.3.1

Test case 2: -

Module Under Test	Admin panel
Description	After successful login to admin panel it shows admin Dashboard with various functionality
Input	<p>Admin panel dashboard menu contain doctors, Schedule, Appointment, Patient.</p> <p>Within Doctor activity admin can add new doctor or remove existing doctor</p> <p>Within Schedule activity admin can make or add a new session or also delete existing schedule</p> <p>Admin also see the existing appointment of all patient and cancel this appointment</p>
Output	All Activity can be well structured and well formatted
Remark	Test successful.

Table No. 6.3.2**Test case 3: -**

Module Under Test	Doctor panel
Description	After successful login to doctor panel it shows doctor Dashboard with various functionality
Input	<p>Doctor panel dashboard menu contain my appointment, my Session, my Patient.</p> <p>Within Doctor activity doctor can see his appointment or remove existing appointment</p> <p>Within session activity doctor can see session or delete existing session</p> <p>Doctor also see the patient with their appointment</p>
Output	All Activity can be well structured and well formatted
Remark	Test successful.

Table no. 6.3.3

Test case 4: -

Module Under Test	Patient panel
Description	After successful login to patient panel it shows patient Dashboard with various functionality
Input	Patient panel dashboard menu contain all doctor, scheduled session my booking, setting. Within patient activity patient can see all doctor list and session Within my booking it shows the booked sessions patient also see the setting for updating details and delete account
Output	All Activity can be well structured and well formatted
Remark	Test successful.

Table no. 6.3.4**Test case 5: -**

Module Under Test	Backend Application
Description	This Module will create the server program for the user end application. Here the data from the user end application will be stored into the database and data processing will be performed.
Output	The data from the user end application will be stored into the database. And application performed successfully.
Remark	Test successful.

Table no. 6.3.5

Chapter 7

Project Plan & Schedule

7.1 Project planning and project resources

- **Define Project Scope:** Clearly outline the features and functionalities of the online doctor appointment system. Identify the target users, such as patients, doctors, and administrators.
- **Set Objectives and Goals:** Specify the goals you want to achieve with the project, such as improving access to healthcare, reducing waiting times, etc.
- **Requirements Gathering:** Collect and document detailed requirements from potential users, stakeholders, and any regulatory or legal considerations.
- **System Architecture:** Design the overall system architecture, including the database structure, front-end, back-end, and any third-party integrations.
- **Technology Stack:** Choose the appropriate technologies for development, considering factors like scalability, security, and ease of use.
- **Project Timeline:** Break down the project into milestones and create a timeline with deadlines for each phase.
- **Risk Assessment:** Identify potential risks and develop strategies to mitigate them. This could include technical challenges, regulatory compliance issues, or unexpected changes in requirements.
- **Project Budget:** Estimate the resources required, including hardware, software, and human resources. Create a budget for the project. Online Doctors Appointment System Dept. of CSE TKIET Warananagar 22
- **Team Structure:** If you're working in a team, define roles and responsibilities for each team member.
- **Development Methodology:** Choose a development methodology (e.g., Agile, Scrum) and plan sprints or iterations.

Project Resources:

1. Human Resources:

- Developers (front-end, back-end, full-stack)
- UI/UX designers.
- Database administrators
- Project manager.
- Quality assurance/testing team

2. Technological resources:

- Server space (consider cloud hosting)
- Development tools and software
- Database management systems.
- Version control systems (e.g., Git)
- Integrated Development Environment (IDE)

3. Data Resources:

- Medical databases (for information on doctors, specialties, etc.)
- User data (if required, ensuring compliance with data protection laws)

4. Financial Resources:

- Budget for software and hardware
- Licensing fees for any third-party tools or services

5. Time Resources:

- Project management software
- Communication tools for team collaboration

6. Training Resources:

- Provide training resources for end-users (doctors, patients, administrators) on how to use the system. Online Doctors Appointment System Dept. of CSE TKIET Warananagar 23

7. Documentation Resources:

- Create user manuals, technical documentation, and any necessary compliance documentation.

8. Testing Resources:

- Testing environments
- Test data
- - Automated testing to

7.2 Project scheduling (ghant chart)

Task	Duration
Project definition	5 Days
Literature review	7 Days
Data collection	15 Days
Pre-processing	5 Days
Model selection	5 Days
Model training	15 Days
Validation and tuning	2 Days
Evaluation	7 Days
Documentation	7 Days
Gantt chart preparation	7 Days
Implementation	28 Days
Testing and validation	5 Days
Final documentation and reporting	5 Days
Deployment	2 Days

Table No.7.2.1 Ghant chart

7.3 Effort estimation

An online doctor appointment system facilitates the scheduling of appointments between patients and healthcare providers through digital platforms such as websites or mobile applications. This system streamlines the appointment booking process, enhances accessibility for patients, and optimizes the scheduling workflow for medical professionals. Estimating the effort required for developing such a system involves breaking down the development process into various tasks and allocating time and resources accordingly.

Overview:

Effort estimation for the online doctor appointment system involves several stages, including requirement analysis, system design, frontend and backend development, testing, deployment, documentation, and ongoing maintenance and support. Each stage requires careful planning and execution to ensure the successful implementation of the system.

Breakdown of Development Stages:

- Requirement Analysis:
 - Gather and analyze requirements from stakeholders, including patients, doctors, and administrators.
 - Define functional and non-functional requirements.
 - Conduct interviews, surveys, and workshops to gather user feedback.
 - Document requirements and create use cases or user stories.
- System Design (1-2 weeks):
 - Design the system architecture, including frontend and backend components.
 - Define database schema and data models.
 - Create wireframes or mockups to visualize user interfaces.
 - Evaluate technology stack and third-party integrations.
 - Review and refine design based on feedback from stakeholders.
- Frontend Development (4-8 weeks):
 - Develop user interfaces for patients, doctors, and administrators.
 - Implement responsive design for seamless access on various devices.
 - Integrate design elements with backend functionality.
 - Ensure accessibility and usability standards are met.
 - Conduct user testing to gather feedback for iterative improvements.

- Backend Development (6-12 weeks):
 - Implement server-side logic for user authentication, appointment scheduling, and data management.
 - Integrate with external APIs for features such as payment processing or notifications.
 - Optimize performance and scalability of backend services.
 - Implement security measures to protect sensitive data.
 - Write clean, maintainable code with appropriate documentation.
 - Page 3: Testing, Deployment, and Documentation
- Database Design and Development (2-4 weeks):
 - Design and implement the database schema based on system requirements.
 - Optimize database performance and ensure data integrity.
 - Implement data access logic and query optimization.
 - Conduct testing to validate database functionality and scalability.
 - Backup and recovery planning for data security.
- Integration Testing (2-4 weeks):
 - Test the integration between frontend and backend components.
 - Verify data flow and communication between different system modules.
 - Identify and resolve integration issues and bottlenecks.
 - Perform regression testing to ensure new features do not break existing functionality.
 - Automated testing where applicable to improve efficiency.
- User Acceptance Testing (UAT) (1-2 weeks):
 - Allow end-users to test the system in a simulated environment.
 - Gather feedback and address any usability or functionality concerns.
 - Ensure the system meets stakeholders' expectations and requirements.
 - Obtain sign-off from stakeholders before deployment.
- Deployment (1-2 weeks):
 - Deploy the system to production servers or cloud platforms.
 - Configure server infrastructure and environment settings.
 - Perform final testing in the production environment.
 - Monitor performance and address any issues that arise during deployment.
 - Implement disaster recovery and failover mechanisms for high availability.

Documentation:

- Write comprehensive documentation for users, administrators, and developers.
- Include user manuals, API documentation, and technical guides.

- Update documentation with any changes or enhancements to the system.
 - Ensure documentation is accessible and easy to understand for various stakeholders.
8. Maintenance and Support:
- Provide ongoing support to address user inquiries, issues, and bug fixes.
 - Monitor system performance and security, applying patches and updates as needed.
 - Implement feature enhancements and improvements based on user feedback.
 - Plan for scalability and future expansion of the system.
 - Establish service level agreements (SLAs) for response times and resolution of support requests.

Chapter 8

Risk Management and Analysis

8.1 Project risk identification

Risk	Identification	Mitigation
Technical Risks	<ul style="list-style-type: none"> - Inadequate technology expertise in the team - Compatibility issues with different browsers or devices 	<ul style="list-style-type: none"> - Ensure the team has the necessary skills or arrange for training. - Conduct thorough compatibility testing during development.
Scope Creep	<ul style="list-style-type: none"> - Changes in requirements not communicated or documented. 	<ul style="list-style-type: none"> - Clearly define and document project requirements.
Data Security and Privacy:	<ul style="list-style-type: none"> - Inadequate measures to protect patient data. 	<ul style="list-style-type: none"> - Implement encryption and secure authentication. - Comply with relevant data protection laws.
User Acceptance:	<ul style="list-style-type: none"> - Users may not find the system user-friendly. 	<ul style="list-style-type: none"> - Involve users in the design and testing phases. - Conduct usability testing and gather feedback.
Team Coordination	<ul style="list-style-type: none"> - Communication breakdowns or conflicts within the team. 	<ul style="list-style-type: none"> - Regular team meetings and status updates. - Encourage open communication and address conflicts promptly.
Testing Issues	<ul style="list-style-type: none"> - Inadequate testing leads to undetected bugs. 	<ul style="list-style-type: none"> - Implement thorough testing plans. - Use automated testing tools where applicable
Internet Connectivity Issues	<ul style="list-style-type: none"> - Dependency on internet connectivity for system access. 	<ul style="list-style-type: none"> - Implement offline capabilities where feasible.

Table No. 8.1.1 Project Risk Management

8.2 Project risk analysis

1. Technical Risks:

- Inadequate server capacity
- Integration challenges with third-party systems
- Security vulnerabilities

2. Operational Risks:

- Insufficient training for users
- Inadequate documentation
- Lack of user acceptance

3. External Risks:

- Changes in regulatory requirements
- Unforeseen changes in healthcare policies
- Vendor-related risks (if using third-party services)

4. Project Management Risks:

- Scope creep
- Resource constraints
- Unrealistic timelines

Chapter 9

Configuration Management

9.1 Installation and uninstallation

- **Installation Step for XAMPP Server:**

1. **Download XAMPP:** Visit the official XAMPP website (<https://www.apachefriends.org/index.html>) and download the version suitable for your operating system (Windows, macOS, or Linux).

2. **Run the Installer:**

- Windows: Double-click the downloaded installer file (e.g., xampp-windows-x64-7.4.27-0-VC15-installer.exe) and follow the installation wizard.
- macOS: Open the downloaded DMG file, drag the XAMPP folder to your Applications folder.
- Linux: Open a terminal, navigate to the directory where the installer is downloaded, and run `sudo ./xampp-linux-x64-7.4.27-0-installer.run`.

3. **Choose Components:** During installation, you'll be prompted to choose which components of XAMPP you want to install. By default, Apache, MySQL, PHP, and phpMyAdmin are selected. You can customize this according to your needs.

4. **Choose Installation Directory:** Choose the directory where you want to install XAMPP. The default is usually C:\xampp on Windows, /Applications/XAMPP on macOS, and /opt/lampp on Linux.'

5. **Complete Installation:** Follow the on-screen instructions to complete the installation process. Once done, XAMPP is installed on your system.

6. **Start XAMPP Control Panel:** After installation, you'll find the XAMPP Control Panel. Start it, and you can start Apache, MySQL, and other components from here.

7. **Test Installation:** Open a web browser and navigate to `http://localhost` or `http://127.0.0.1`. If you see the XAMPP dashboard, the installation was successful.

- **Uninstallation Step for XAMPP Server:**

1. **Stop XAMPP Services:** Before uninstalling, make sure to stop all XAMPP services. You can do this from the XAMPP Control Panel by clicking "Stop" next to each service.

2. **Uninstall XAMPP:**

- Windows: Navigate to Control Panel > Programs > Programs and Features, find XAMPP in the list, and click "Uninstall".

- macOS: Simply drag the XAMPP folder from your Applications folder to the Trash.
- Linux: Run the uninstaller script located in the XAMPP directory. Open a terminal, navigate to the XAMPP directory (e.g., /opt/lampp) and run .sudo/uninstall.

3. Remove XAMPP Directory:

- Windows: If the XAMPP directory (C:\xampp by default) still exists, delete it manually.
- macOS: Empty the Trash to delete the XAMPP folder permanently.
- Linux: If the XAMPP directory (/opt/lampp by default) still exists, delete it manually.

4. Remove XAMPP Configuration Files:

- Windows: Navigate to C:\Users\<YourUsername>\AppData\Roaming and delete the XAMPP folder if it exists.
- macOS: Navigate to ~/.config and delete the XAMPP folder if it exists.
- Linux: Navigate to ~/.config and delete the XAMPP folder if it exists.

5. Remove Environment Variables (Optional):

- Windows: Remove the XAMPP directory from the PATH environment variable.
- macOS / Linux: If you added XAMPP to your PATH manually, remove it from your shell configuration file (e.g., bashrc, .bash_profile, zshrc).

- **Installation Step for VS Code:**

- **Download VS Code:** Visit the official Visual Studio Code website (<https://code.visualstudio.com/>) and download the installer suitable for your operating system (Windows, macOS, or Linux).
- **Run the Installer:**
 - Windows: Double-click the downloaded installer file (e.g., VSCodeSetup.exe) and follow the installation wizard.
 - macOS: Open the downloaded DMG file, drag the Visual Studio Code icon to the Applications folder.
 - Linux: Open a terminal, navigate to the directory where the installer is downloaded, and follow the distribution-specific installation instructions.
- **Launch VS Code:** After installation, you can launch Visual Studio Code from the Start menu (Windows) or from the Applications folder (macOS). On Linux, you can usually launch it from the application menu or by typing code in the terminal.
- **Extensions:** VS Code has a rich ecosystem of extensions. You can install extensions from the VS Code Marketplace to enhance functionality.

- **Uninstallation Step for VS Code:**

➤ **Close VS Code:** Before uninstalling, make sure Visual Studio Code is closed.

➤ **Uninstall VS Code:**

- Windows: Navigate to Control Panel > Programs > Programs and Features, find Visual Studio Code in the list, and click "Uninstall".
- macOS: Simply drag the Visual Studio Code icon from the Applications folder to the Trash.
- Linux: Depending on how you installed VS Code (e.g., via package manager or manually), the method of uninstallation may vary. If you installed it via a package manager, you should uninstall it using the same package manager. If you installed it manually, you can usually delete the installation directory or follow the uninstallation instructions provided with the package.

➤ **Remove Configuration Files:**

- Windows: Navigate to C:\Users\<YourUsername>\AppData\Roaming and delete the Code or VSCode folder if it exists.
- macOS: Navigate to ~/Library/Application Support and delete the Code or VSCode folder if it exists.
- Linux: Navigate to ~/.config and delete the Code or VSCode folder if it exists.

➤ **Remove Extensions (Optional):**

If you installed any extensions, you might want to remove them manually. Extension files are typically stored in the vscode/extensions directory in your user's home directory.

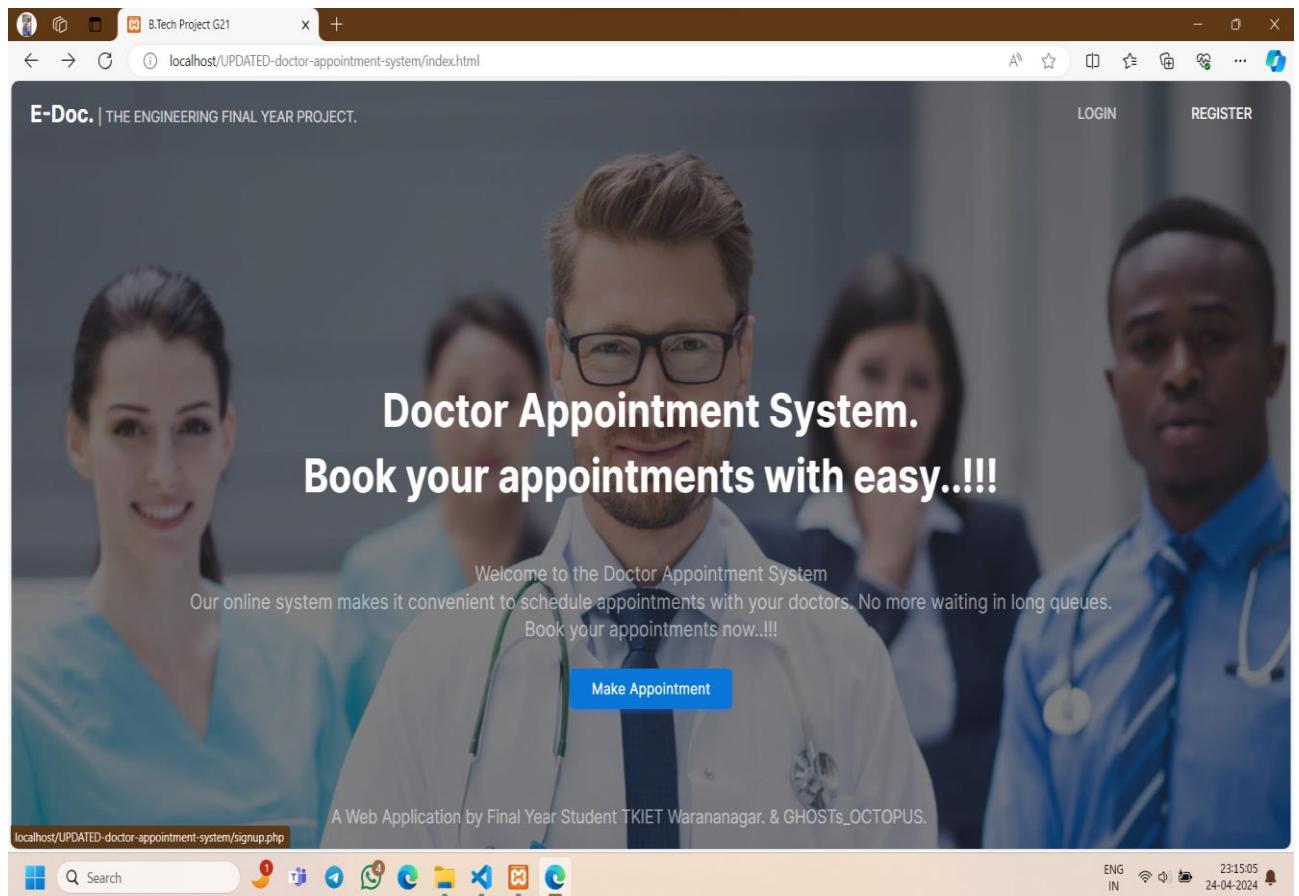
9.2 User manual

GET STARTED:

1. Open your XAMPP Control Panel and start Apache and MySQL.
2. Extract the downloaded source code zip file.
3. Copy the extracted source code folder and paste it into the XAMPP's "htdocs" directory.
4. Browse the PHPMyAdmin in a browser. i.e. <http://localhost/phpmyadmin>
5. Create a new database naming `edoc`.
6. Import the provided SQL file. The file is known as DATABASE edoc.sql located inside the source code root folder.
7. Browse the Doctor's Appointment System in a browser. i.e. <http://localhost/UPDATED-doctor-appointment-system/index.html>

9.2.0 Output Screenshot

➤ Home Dashboard Page:

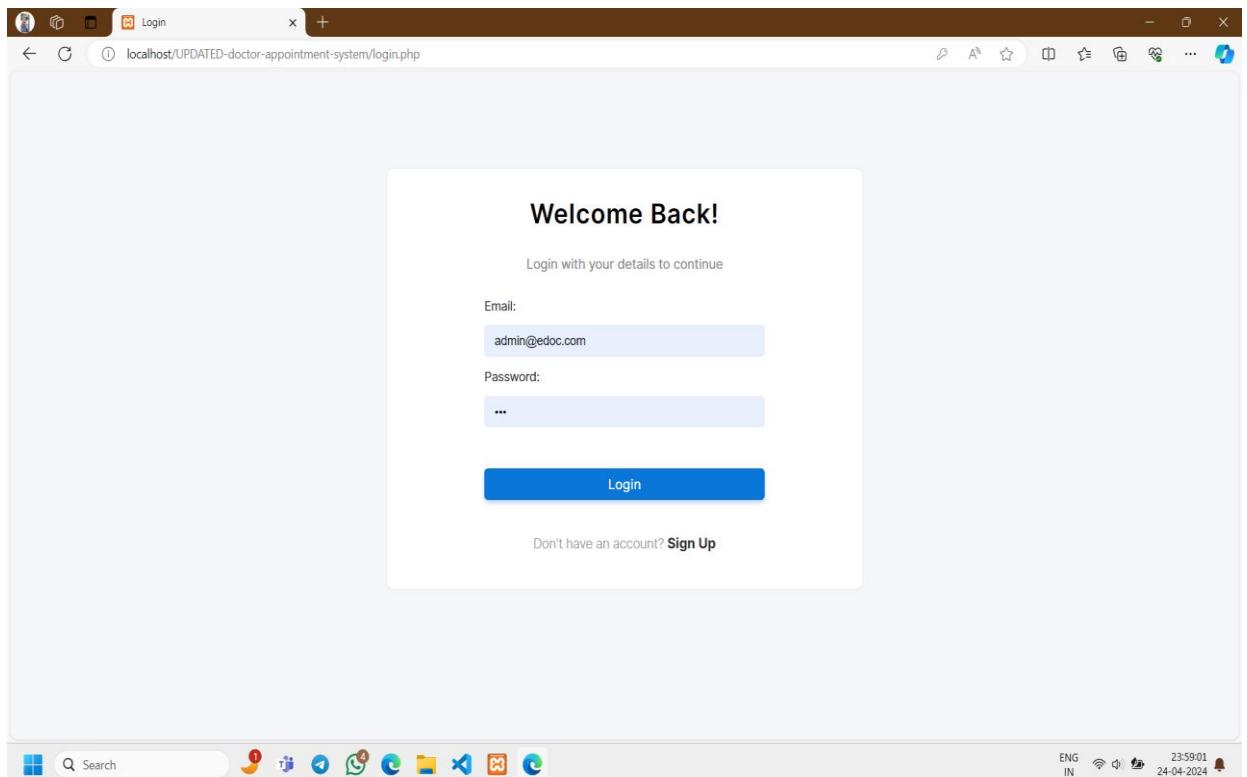


Screenshot 9.2.0 Home Dashboard Page

- A user can see the homepages of online doctor appointment system with Head, Tital and Body which contain different container and button for the sign in and sign up or make appointment. User have to do sign up and make appointment through sign in. It also contains footer.
- This dashboard is common to all doctors, all Patient, and Administer for their login dashboard.

9.2.1 Administer Panel

➤ Login Page



Screenshot 9.2.1.1 Login Page

- After click on Login button of home page it jumps to login page
- On that page there is textbox for email and password. One button for login to panel and key for sign Up for new registration.
- If enter valid email id and password then enter to administer panel

➤ Administer Dashboard

The screenshot shows the Administer Dashboard interface. At the top right, it displays "Today's Date" as 2024-04-25. On the left, there's a sidebar with a user profile for "Administrator" (admin@edoc.com) and links for Dashboard, Doctors, Schedule, Appointment, and Patients. The main area has a search bar at the top. Below it, a section titled "Status" shows counts for Doctors (10), Patients (4), NewBooking (0), and Today Sessions (0). Two main sections follow: "Upcoming Appointments until Next Thursday" and "Upcoming Sessions until Next Thursday". Each section includes a brief description, a table header, and a sample row. At the bottom, there are "Show all Appointments" and "Show all Sessions" buttons.

Appointment number	Patient name	Doctor	Session

Session Title	Doctor	Scheduled Date & Time
Accident and emergency medicin	Test Doctor	2024-04-30 10:00

Screenshot 9.2.1.2 Administer Dashboard

- After login to admin panel admin dashboard open
- Admin dashboard contain menu bar, search bar, today's date, status of doctor, patient, booking, today's session and upcoming appointment

➤ Doctor's Activity (Administer panel)

The screenshot shows the 'Doctors' section of the Administer panel. At the top, there is a search bar with the placeholder 'Search Doctor name or Email' and a 'Search' button. To the right of the search bar is a date field labeled 'Today's Date' with the value '2024-04-25'. On the far left, a sidebar menu includes 'Dashboard', 'Doctors' (which is currently selected), 'Schedule', 'Appointment', and 'Patients'. The main content area is titled 'Add New Doctor' and shows a table titled 'All Doctors (10)'. The table has columns for 'Doctor Name', 'Email', 'Specialties', and 'Events'. Each row contains a doctor's information and three buttons: 'Edit', 'View', and 'Remove'. The data in the table is as follows:

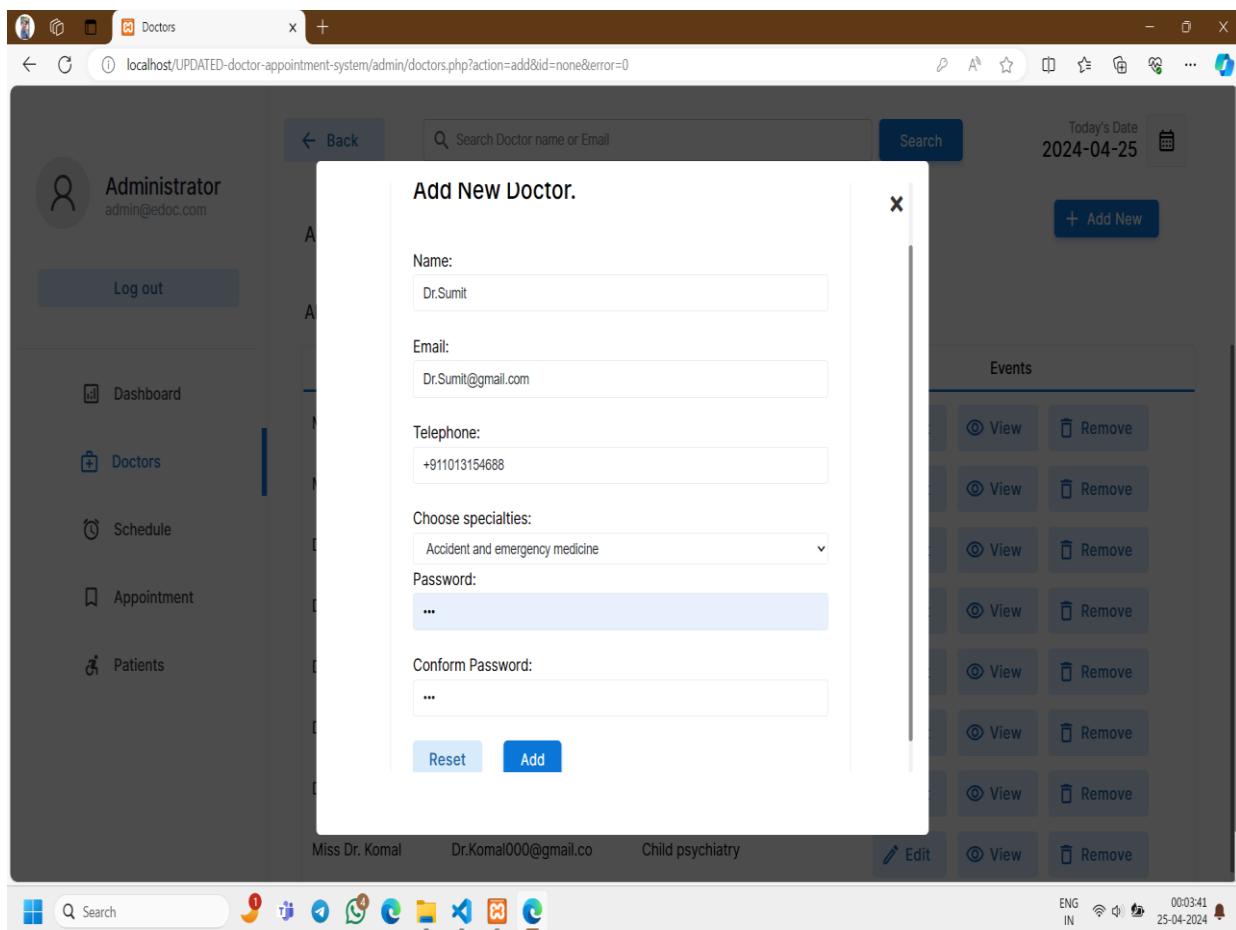
Doctor Name	Email	Specialties	Events
Miss Dr. Vedhika	Dr.Vedhika@gmail.com	Dental, oral and max	<button>Edit</button> <button>View</button> <button>Remove</button>
Miss Dr. Anupa	Dr.Anupa@gmail.com	Cardiology	<button>Edit</button> <button>View</button> <button>Remove</button>
Dr. Sunny	Dr.Sunny@gmail.com	Public health and Pr	<button>Edit</button> <button>View</button> <button>Remove</button>
Dr. Keran	Dr.Keran@gmail.com	Cardiology	<button>Edit</button> <button>View</button> <button>Remove</button>
Dr. Adarsh	Dr.Adarsh@gmail.com	Accident and emergen	<button>Edit</button> <button>View</button> <button>Remove</button>
Dr. Raj	Dr.Raj@edoc.com	General surgery	<button>Edit</button> <button>View</button> <button>Remove</button>
Dr. Kunal	Dr.kunal000@gmail.co	General surgery	<button>Edit</button> <button>View</button> <button>Remove</button>
Miss Dr. Komal	Dr.Komal000@gmail.co	Child psychiatry	<button>Edit</button> <button>View</button> <button>Remove</button>
Miss Dr. Arpita	Dr.kolearpita000@gma	Dental, oral and max	<button>Edit</button> <button>View</button> <button>Remove</button>

At the bottom of the screen, there is a taskbar with various icons and system status indicators.

Screenshot 9.2.1.3 Doctor's Activity (Administer panel)

- Doctor activity contain search bar at top for specific doctor search
- A button for add new doctor to database
- And list of all doctors with their details name, email, specification and event like edit, view and remove.

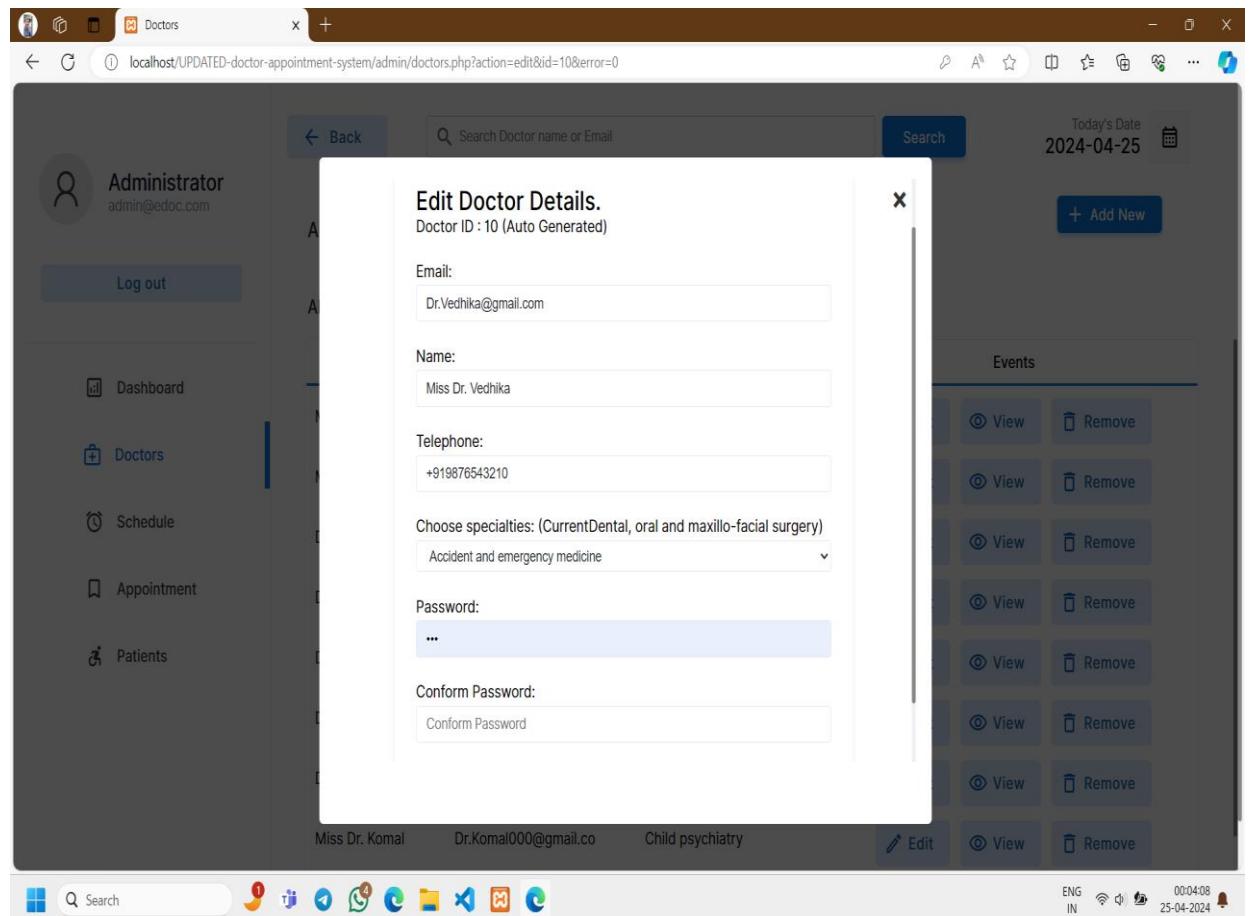
➤ Doctor's Activity (Add New Doctor)



Screenshot 9.2.1.4 Doctor's Activity (Add New Doctor)

- After click on add new button its open pop up tab for add new doctor
- In that tab there are some textbox for name, email, telephone no, and password and conform password and two buttons for reset and save.

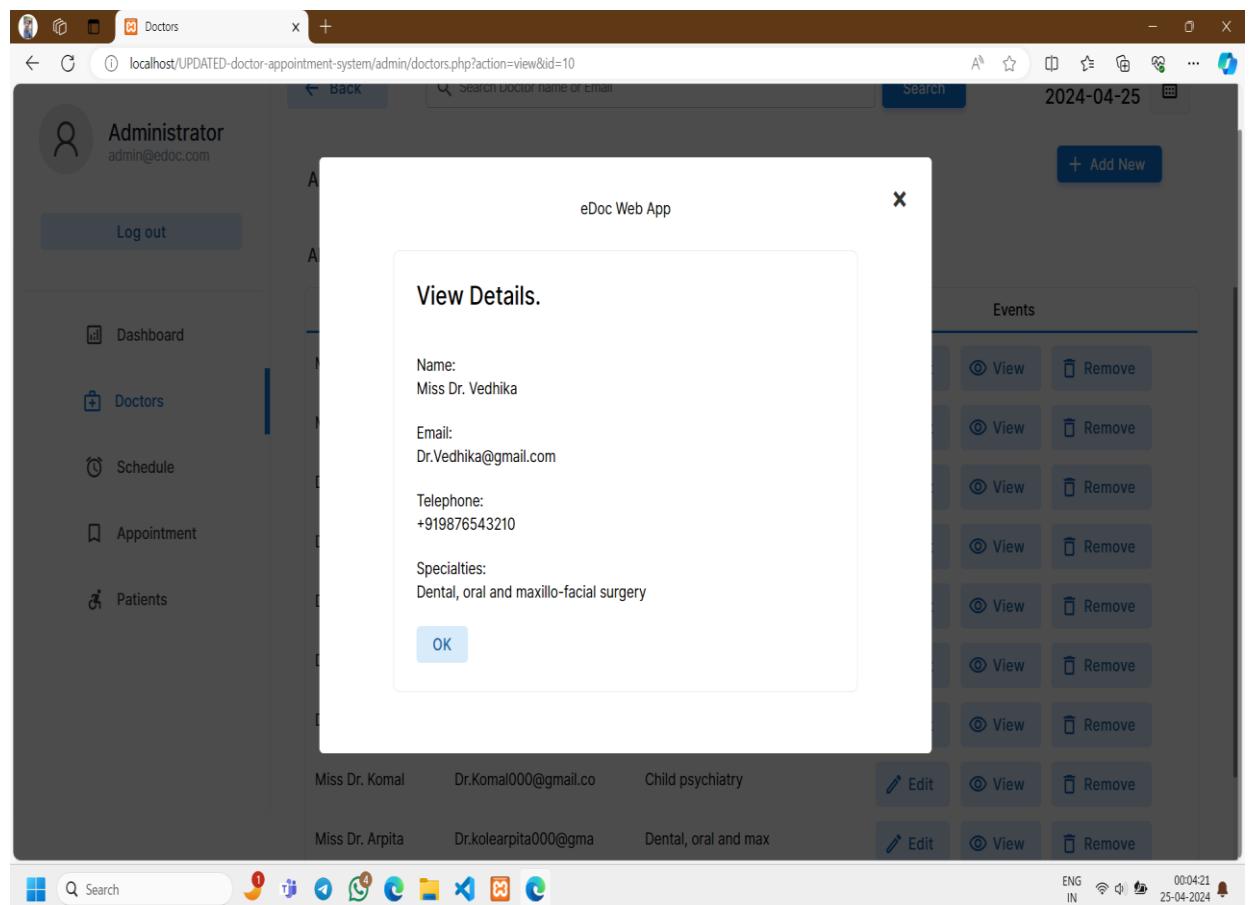
➤ Doctor's Activity (Edit Doctor Details)



Screenshot 9.2.1.5 Doctor's Activity (Edit Doctor Details)

- After click on the edit button in the event its open the popup tab for edit doctor details
- In that tab doctors can update their details like email, name, telephone, specialties and password

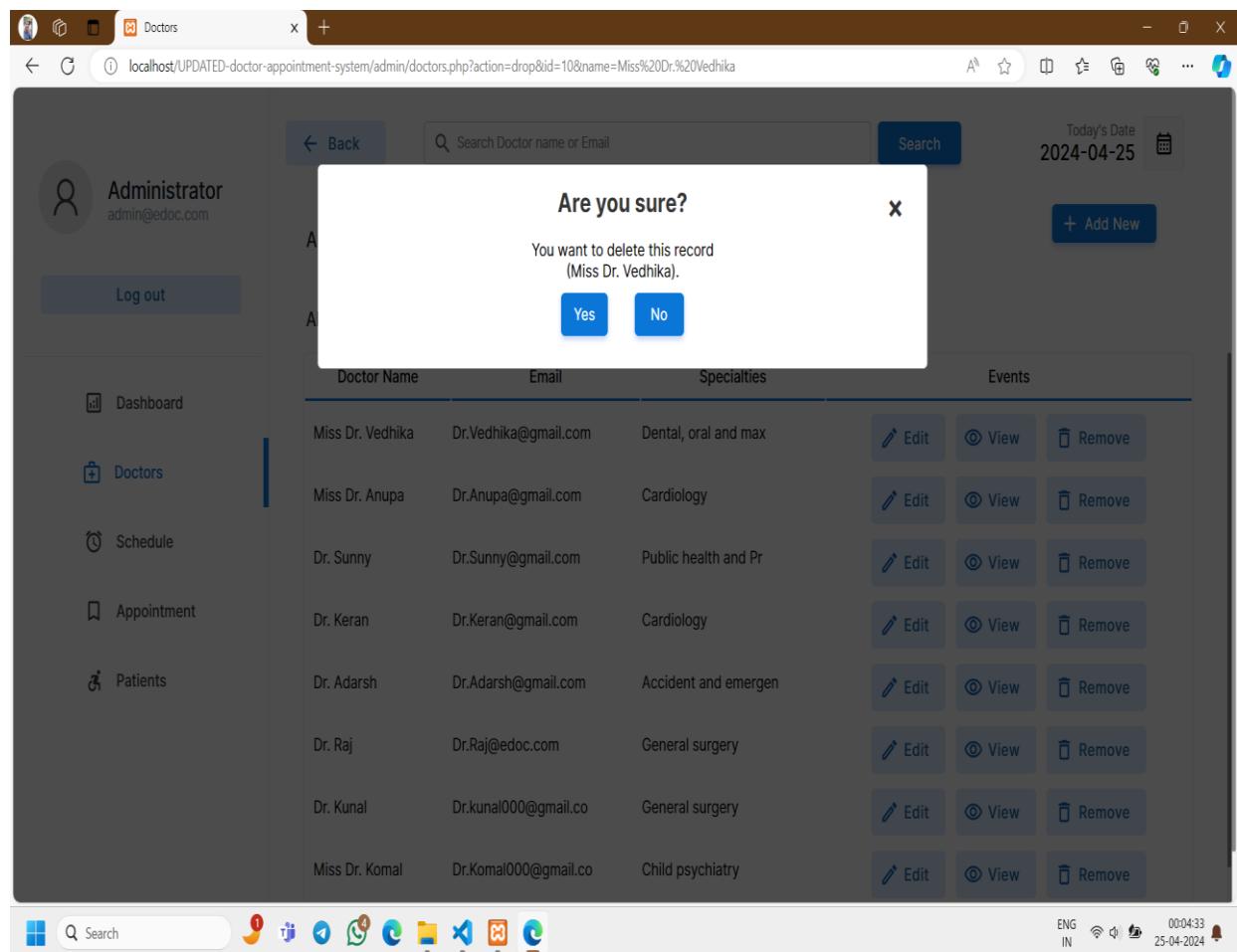
➤ Doctor's Activity (View Doctor Details)



Screenshot 9.2.1.6 Doctor's Activity (View Doctor Details)

- After click on the view button in the event its open the popup tab for view doctor details
- In that tab there are doctor details like email, name, telephone, Specialties

➤ Doctor's Activity (Delete Details)



Screenshot 9.2.1.7 Doctor's Activity (Delete Doctor Details)

- After click on the remove button in the event its open the popup tab for delete doctor details
- In the popup tab show the conformation notification for delete record of doctors
- It gives two option as “yes” or “no” to user

➤ Schedule Activity (schedule Manager)

The screenshot shows the 'Schedule Manager' page of an online doctors appointment system. At the top right, it displays 'Today's Date' as 2024-04-25. On the left, there is a sidebar with navigation links: Dashboard, Doctors, Schedule (which is currently selected), Appointment, and Patients. The main content area has a header 'Schedule Manager' with a 'Back' button and a 'Schedule a Session' button. Below this is a section titled 'All Sessions (6)' with a search/filter bar. A table lists six sessions:

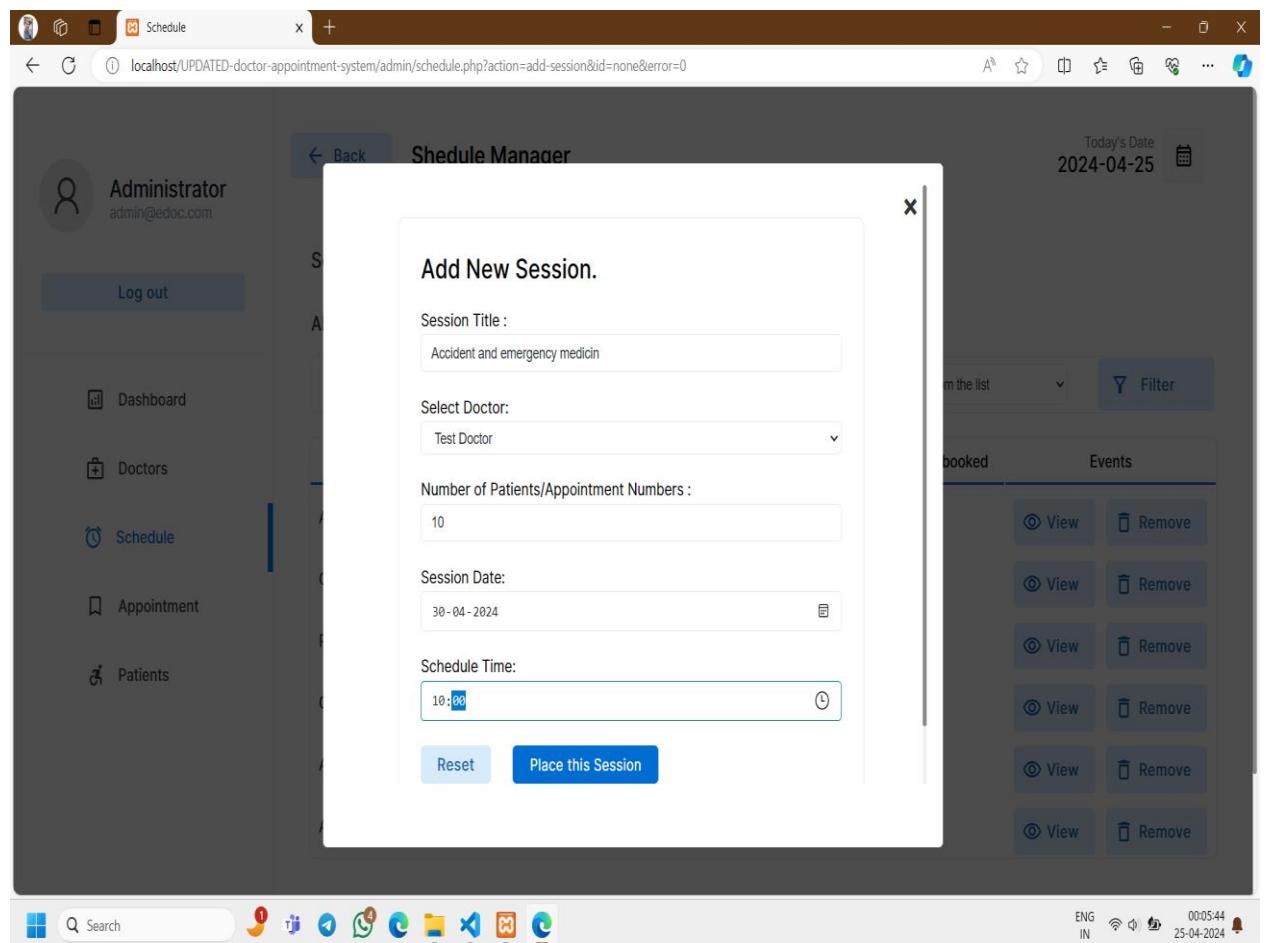
Session Title	Doctor	Scheduled Date & Time	Max num that can be booked	Events
Accident and emergency medicin	Test Doctor	2024-04-30 10:00	5	<button>View</button> <button>Remove</button>
General surgery	Dr. Kunal	2024-04-18 12:15	3	<button>View</button> <button>Remove</button>
Public health and Preventive M	Dr. Sunny	2024-04-15 10:15	4	<button>View</button> <button>Remove</button>
General surgery	Dr. Kunal	2024-04-15 14:00	2	<button>View</button> <button>Remove</button>
Accident and emergency medicin	Dr. Adarsh	2024-04-15 09:00	15	<button>View</button> <button>Remove</button>
Accident and emergency medicin	Test Doctor	2024-03-05 09:00	5	<button>View</button> <button>Remove</button>

At the bottom, there is a toolbar with various icons and system status indicators like ENG IN, 00:45s, and 25-04-2024.

Screenshot 9.2.1.8 Schedule Activity (Schedule Manager)

- When we move forward to menu as schedule its open schedule manager page
- In that page it shows the time and date at corner of right side
- Below next line it gives button for add new session of schedule appointment for register doctors
- Below that there is list of all doctors whose already booked their appointment session
- In that session it shows session title, name of doctor, time & date of schedule, number of booking, event like view and remove

➤ Schedule Activity (add new session)



Screenshot 9.2.1.9 Schedule Activity (Add new section)

- After click on add session button its open popup tab for book new session
- In that there is section title, select doctor, select total number of patients, session date, schedule time tabs to fill.

➤ Schedule Activity (search through date or doctor name)

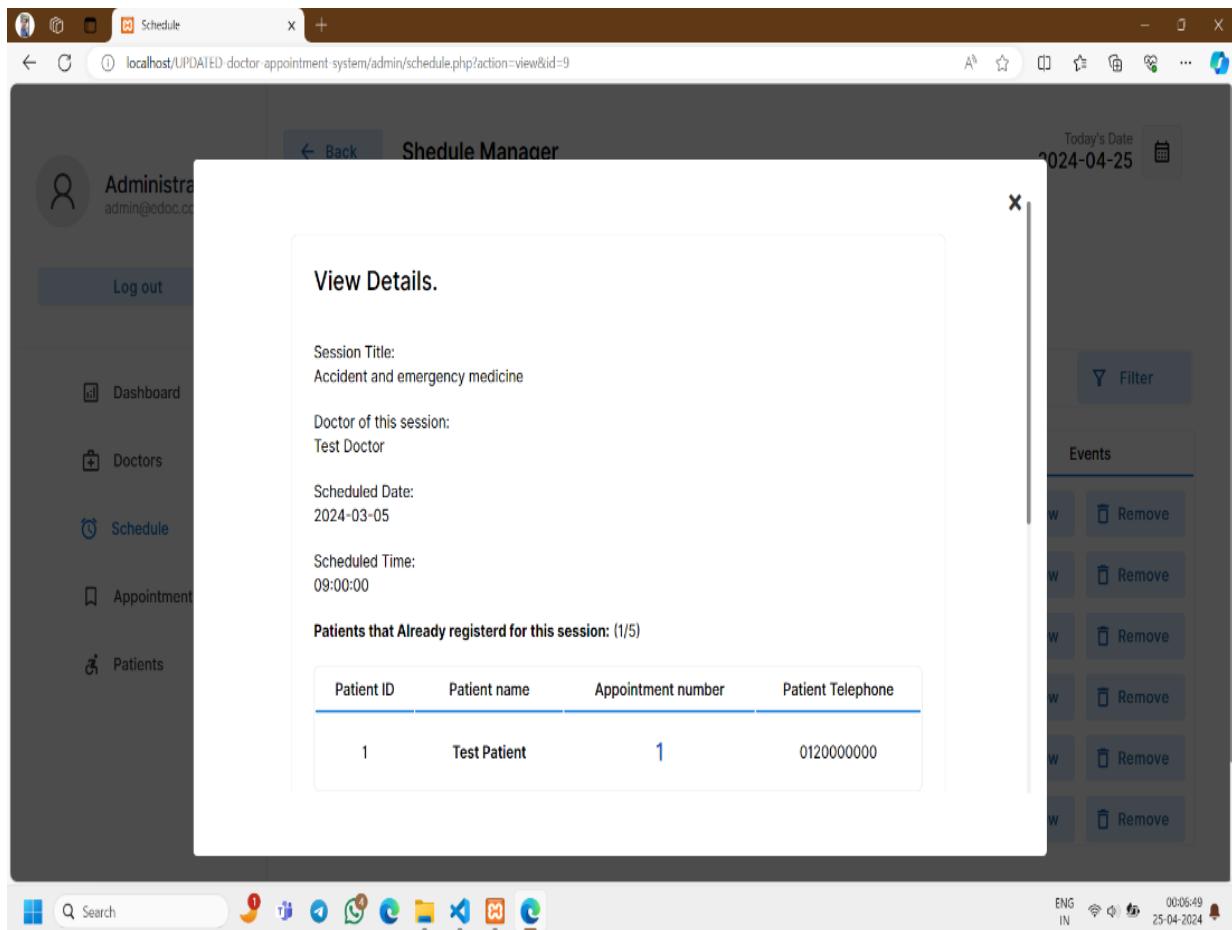
The screenshot shows the 'Schedule Manager' page of an online doctors appointment system. On the left, there's a sidebar with navigation links: Dashboard, Doctors, Schedule (which is selected and highlighted in blue), Appointment, and Patients. The main content area has a title 'Schedule Manager' with a 'Back' button. Below it is a section titled 'Schedule a Session' with a 'Add a Session' button. A table lists 'All Sessions (6)'. The table columns are 'Session Title', 'Doctor', 'Scheduled Date & Time', and 'Events'. The 'Doctor' column includes a dropdown menu titled 'Choose Doctor Name from the list' which is currently open, showing a list of names: Dr. Adarsh, Dr. Keran, Dr. Kunal, Dr. Raj, Dr. Sunny, Miss Dr. Anupa, Miss Dr. Apita, Miss Dr. Komal, Miss Dr. Vedhika, and Test Doctor. To the right of the table, there are 'View' and 'Remove' buttons for each session. At the bottom of the page, there's a toolbar with various icons and a status bar showing 'ENG IN', '00:06:04', '25-04-2024', and a battery icon.

Session Title	Doctor	Scheduled Date & Time	Events
Accident and emergency medicin	Test Doctor	2024-04-30 10:00	View Remove
General surgery	Dr. Kunal	2024-04-18 12:15	View Remove
Public health and Preventive M	Dr. Sunny	2024-04-15 10:15	View Remove
General surgery	Dr. Kunal	2024-04-15 14:00	View Remove
Accident and emergency medicin	Dr. Adarsh	2024-04-15 09:00	View Remove
Accident and emergency medicin	Test Doctor	2024-03-05 09:00	View Remove

Screenshot 9.2.1.10 Schedule Activity (Search or Schedule)

- In that section administrator can find the schedule section through date or doctor name

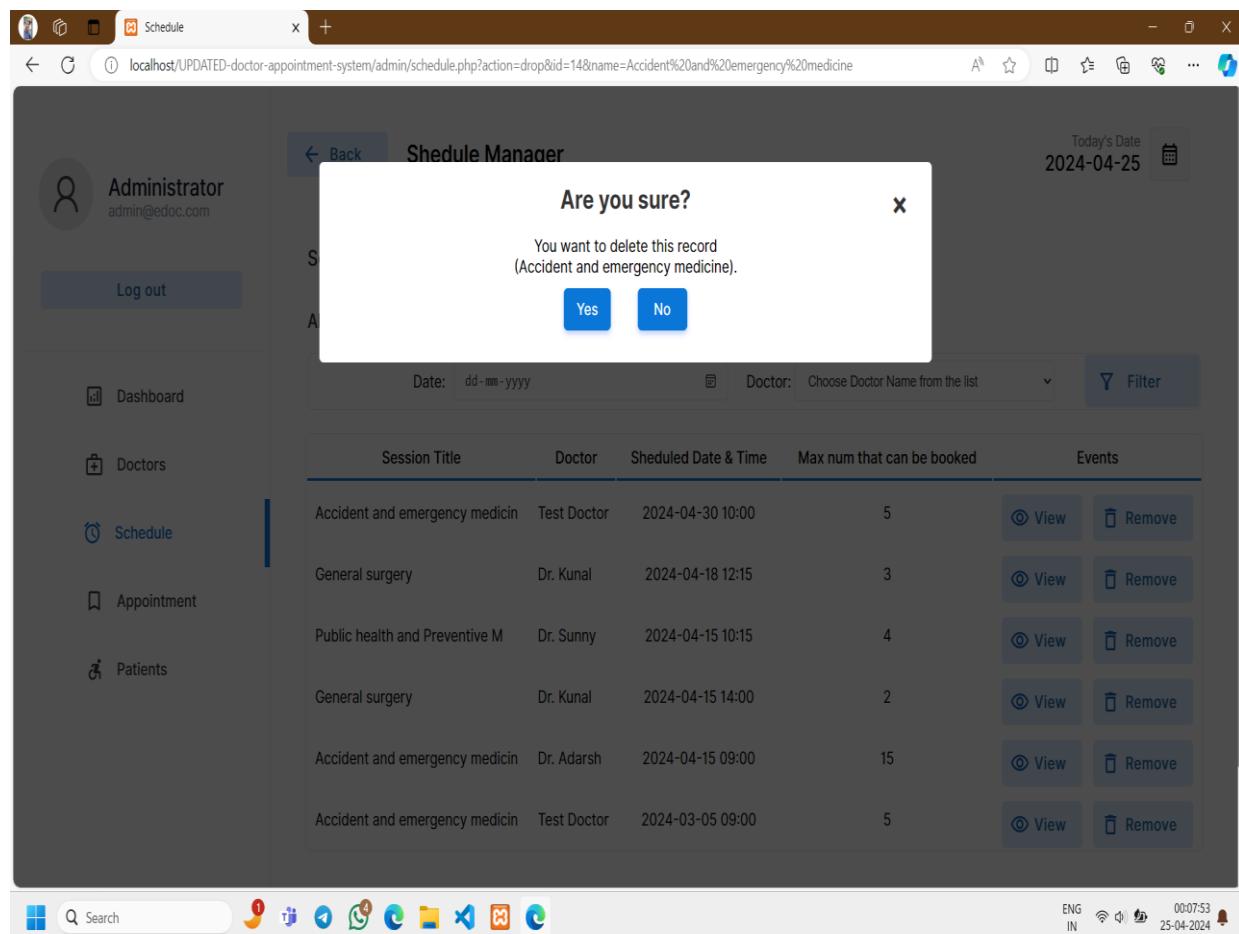
➤ Schedule Activity (view doctor schedule)



Screenshot 9.2.1.11 Schedule Activity (view doctor schedule)

- After click on view button of schedule event it gives popup tab for view session detail for that particular doctor
- In that view details there are session title, name of doctor, scheduled date and time of already register patients

➤ Schedule Activity (Delete Session)



Screenshot 9.2.1.12 Schedule Activity (record delete session)

- After click on remove button of schedule event it gives popup tab for remove session detail for that particular doctor
- On that popup tab it gives confirmation notification for delete record of that session and also provide buttons for "yes" and "no" for confirmation

➤ Appointment Activity (Appointment Manager)

The screenshot shows the 'Appointment Manager' page of an online doctors appointment system. The top navigation bar includes icons for user profile, appointment, and a search bar labeled 'Appointments'. The title 'Appointment Manager' is centered above a table. On the left, a sidebar menu lists 'Administrator' (admin@edoc.com), 'Log out', 'Dashboard', 'Doctors', 'Schedule', 'Appointment' (selected), and 'Patients'. The main content area displays 'All Appointments (8)' with a search/filter bar. The table lists the following data:

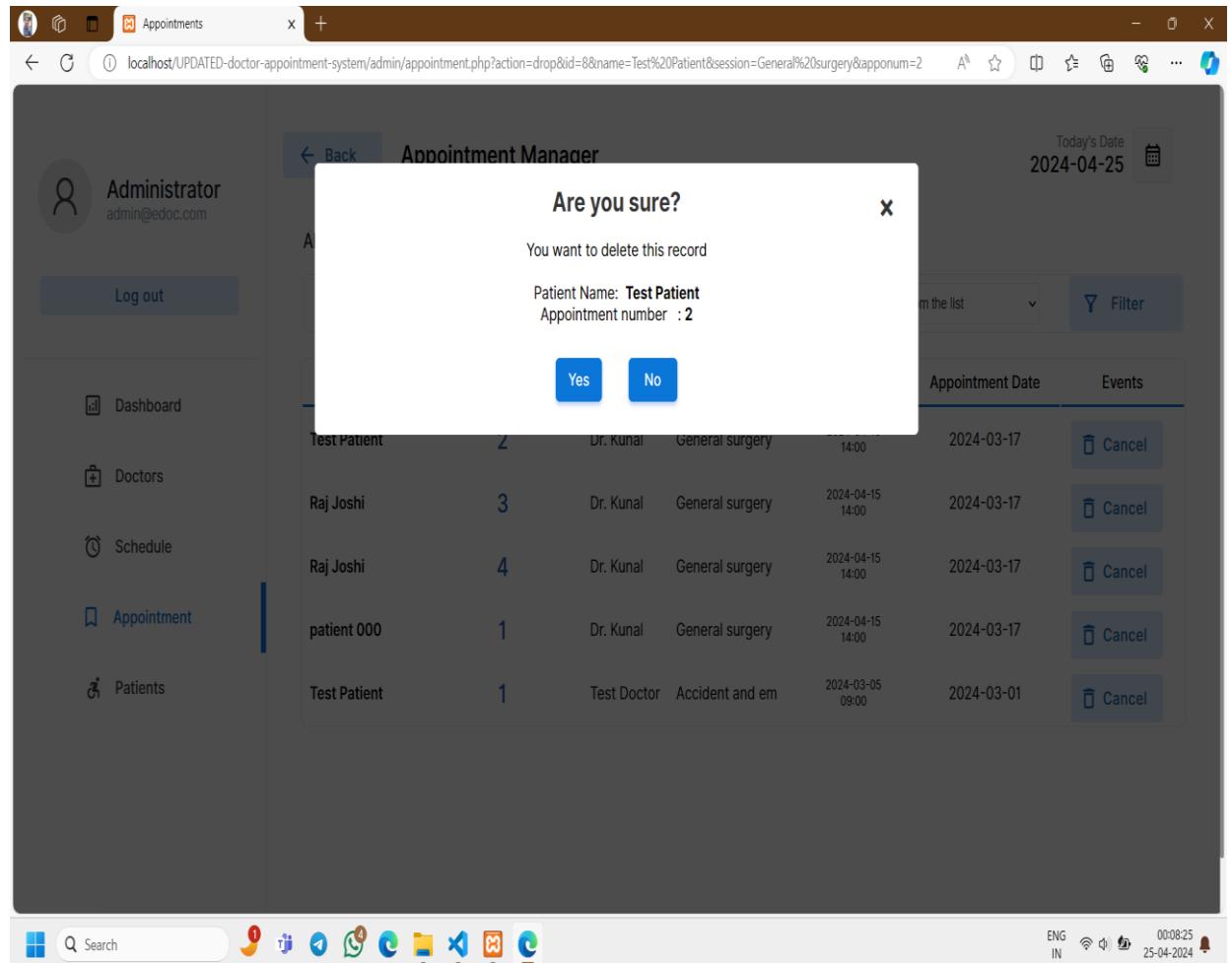
Patient name	Appointment number	Doctor	Session Title	Session Date & Time	Appointment Date	Events
Test Patient	2	Dr. Kunal	General surgery	2024-04-15 14:00	2024-03-17	<button>Cancel</button>
Raj Joshi	3	Dr. Kunal	General surgery	2024-04-15 14:00	2024-03-17	<button>Cancel</button>
Raj Joshi	4	Dr. Kunal	General surgery	2024-04-15 14:00	2024-03-17	<button>Cancel</button>
patient 000	1	Dr. Kunal	General surgery	2024-04-15 14:00	2024-03-17	<button>Cancel</button>
Test Patient	1	Test Doctor	Accident and em	2024-03-05 09:00	2024-03-01	<button>Cancel</button>

The bottom status bar shows system information: ENG IN, battery level, time (00:08:08), date (25-04-2024), and a notification bell.

Screenshot 9.2.1.13 Appointment Activity (Appointment Manager)

- When we move forward to menu as appointment its open appointment manager page
- In that page it shows the time and date at corner of right side
- Below next line it shows search tab for search for who doctor can have booked appointment with filter
- Below that all list of booked appointment for that particular doctor in that there is patient name, appointment number, doctor name, session title, data & time and event block
- In event block a button for cancel appointment for that particular doctor and patient.

➤ **Appointment Activity (Delete Appointment)**



Screenshot 9.2.1.14 Appointment Activity (Delete Appointment)

- After click on the cancel button of event block it gives the popup tab for conformation
- In that tab there are details of that patient name and appointment number and button for “yes” or “no” for conformation

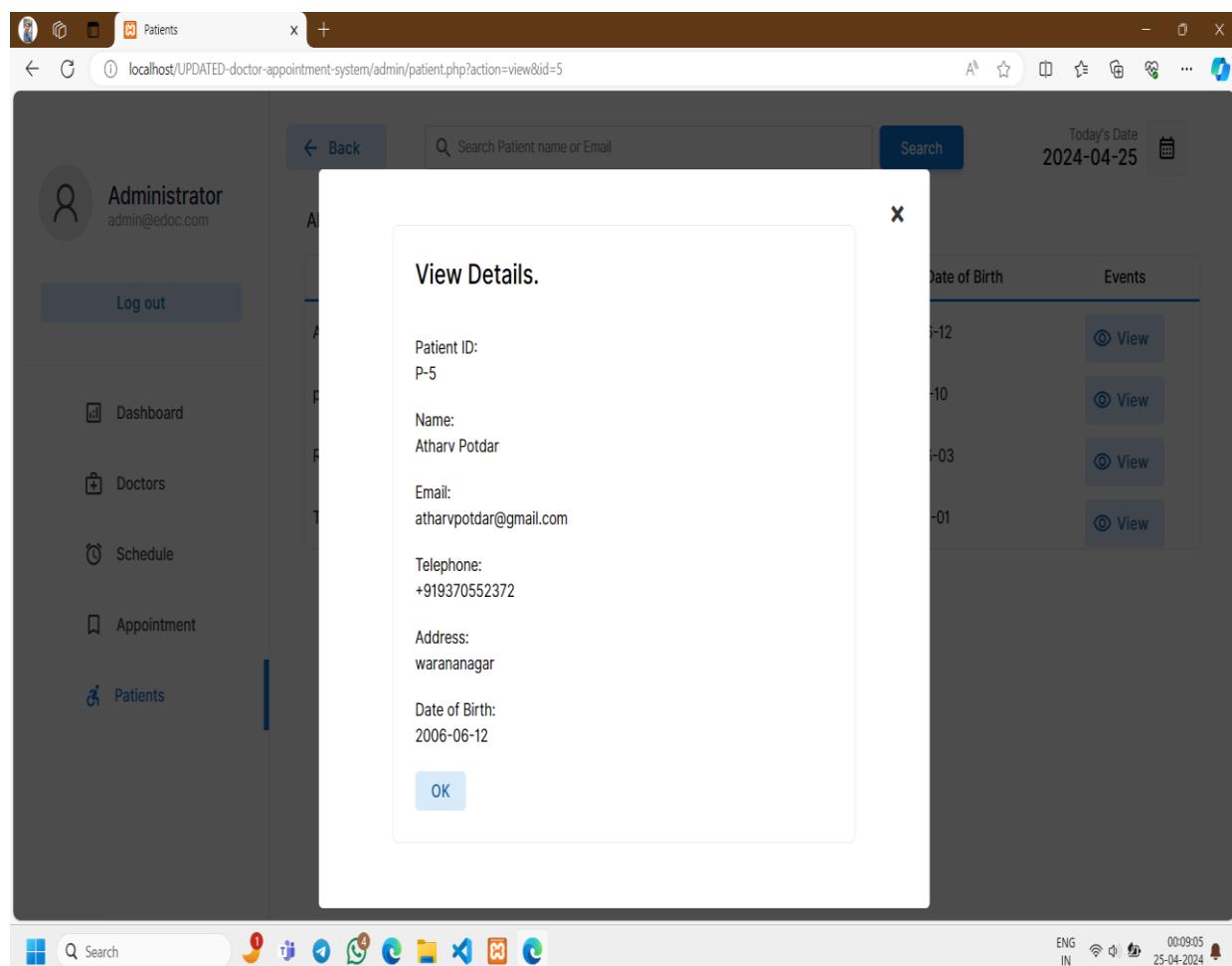
➤ **Patient Activity (List of Patient)**

Name	Telephone	Email	Date of Birth	Events
Atharv Potdar	+919370552	atharvpotdar@gmail.c	2006-06-12	View
patient 000	+919370552	patient000@edoc.com	2001-10-10	View
Raj Joshi	+911001001	RajJoshi@edoc.com	2022-06-03	View
Test Patient	0120000000	patient@edoc.com	2000-01-01	View

Screenshot 9.2.1.15 Patient Activity (List of Patient)

- When we move forward to menu as patient its open patient page
- In that page it shows the time and date at corner of right side
- Below that all list of registered patients with their details like name, telephone, email id, date of birth and view event etc.
- In event block a button for view all details of that particular patient.

➤ **Patient Activity (View of Patient Details)**

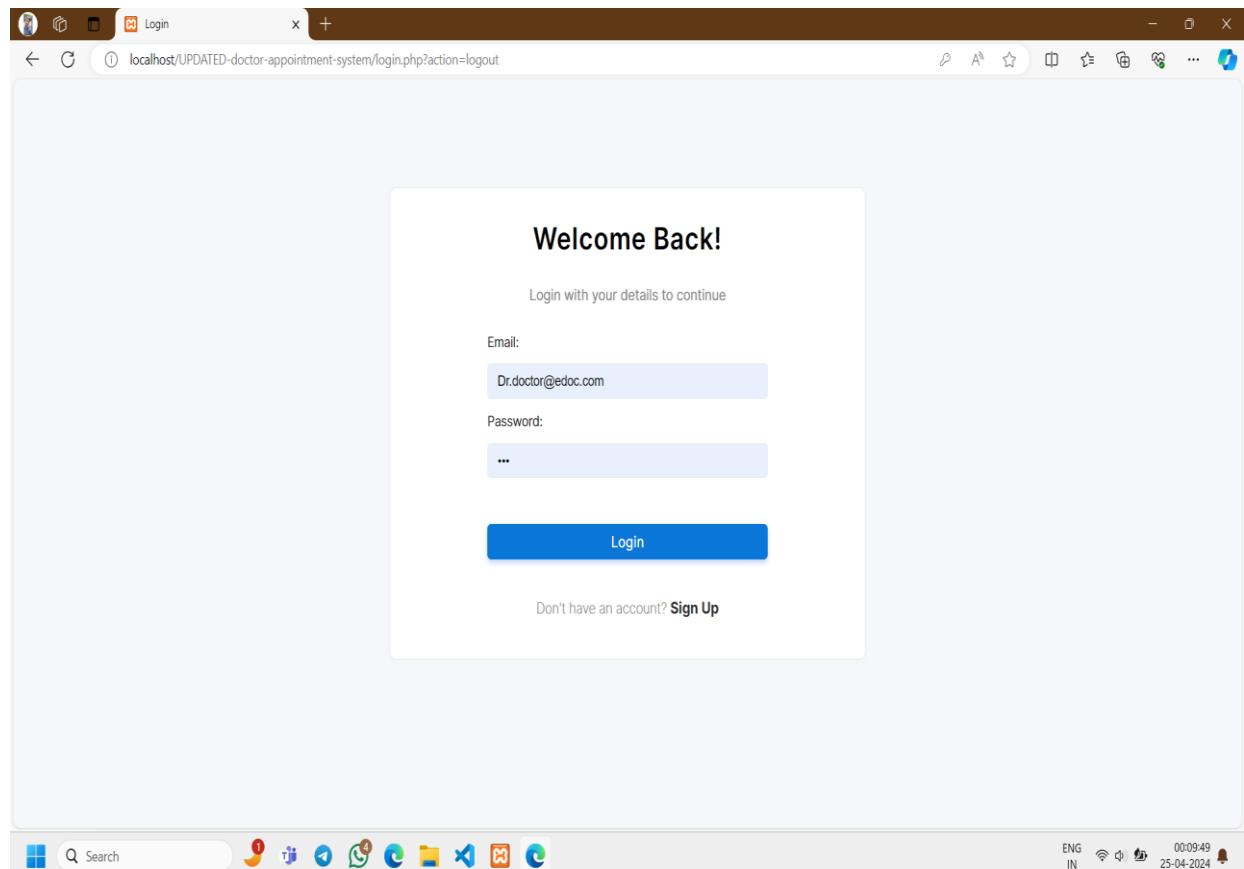


Screenshot 9.2.1.16 Patient Activity (View of Patient Details)

- After click on view button of event section of patient activity it displays popup tab for view details of patient
- Within that tab there are patient id, patient name, email id, telephone number, address and date of birth.

9.2.2 Doctor's Panel

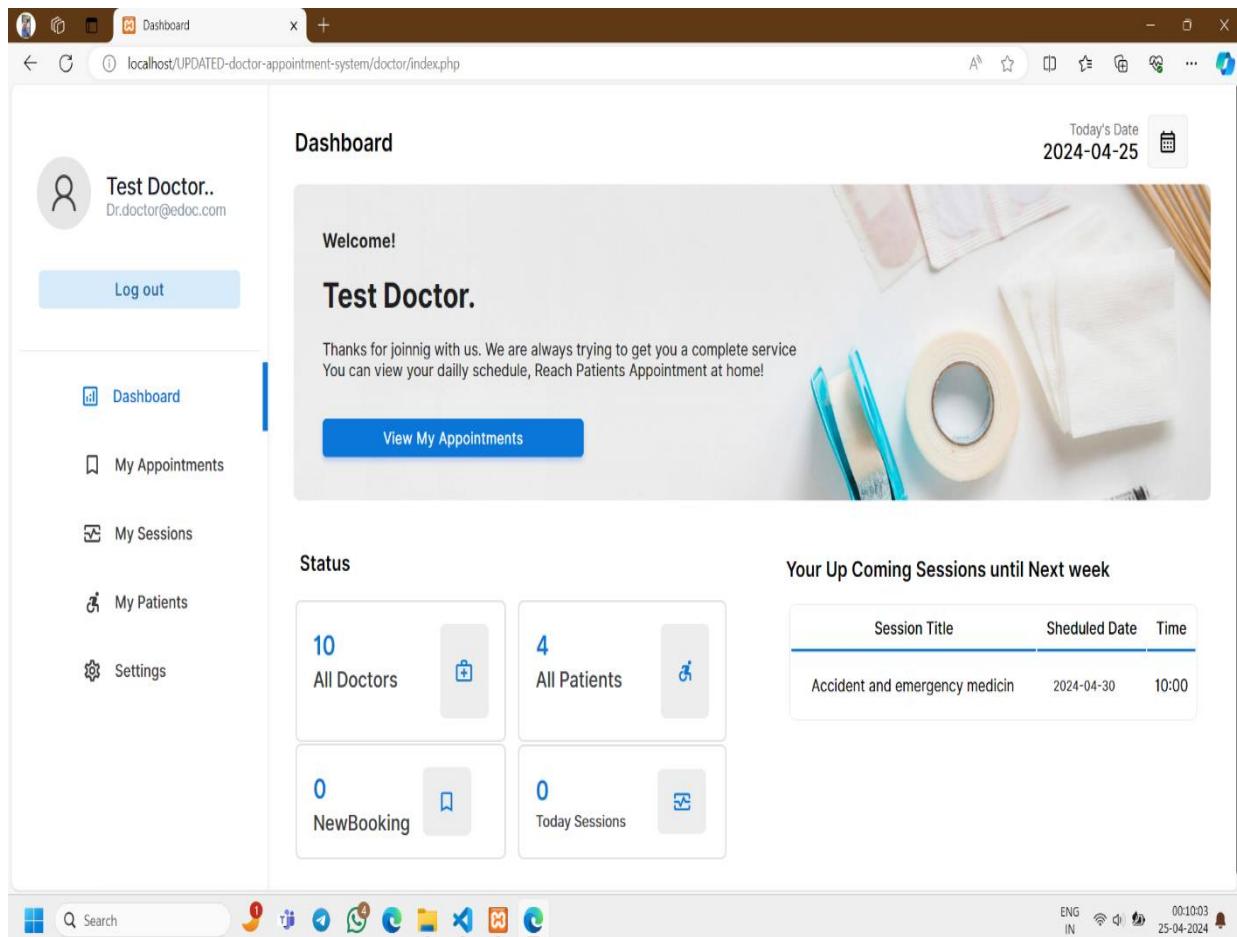
➤ Login Page



Screenshot 9.2.2.1 Login Page

- After click on Login button of home page (Dashboard Page) it jumps to login page
- On that page there is textbox for email and password. One button for login to panel and key for sign Up for new registration.
- If enter valid email id and password which is provide by administrator then enter to doctor's dashboard

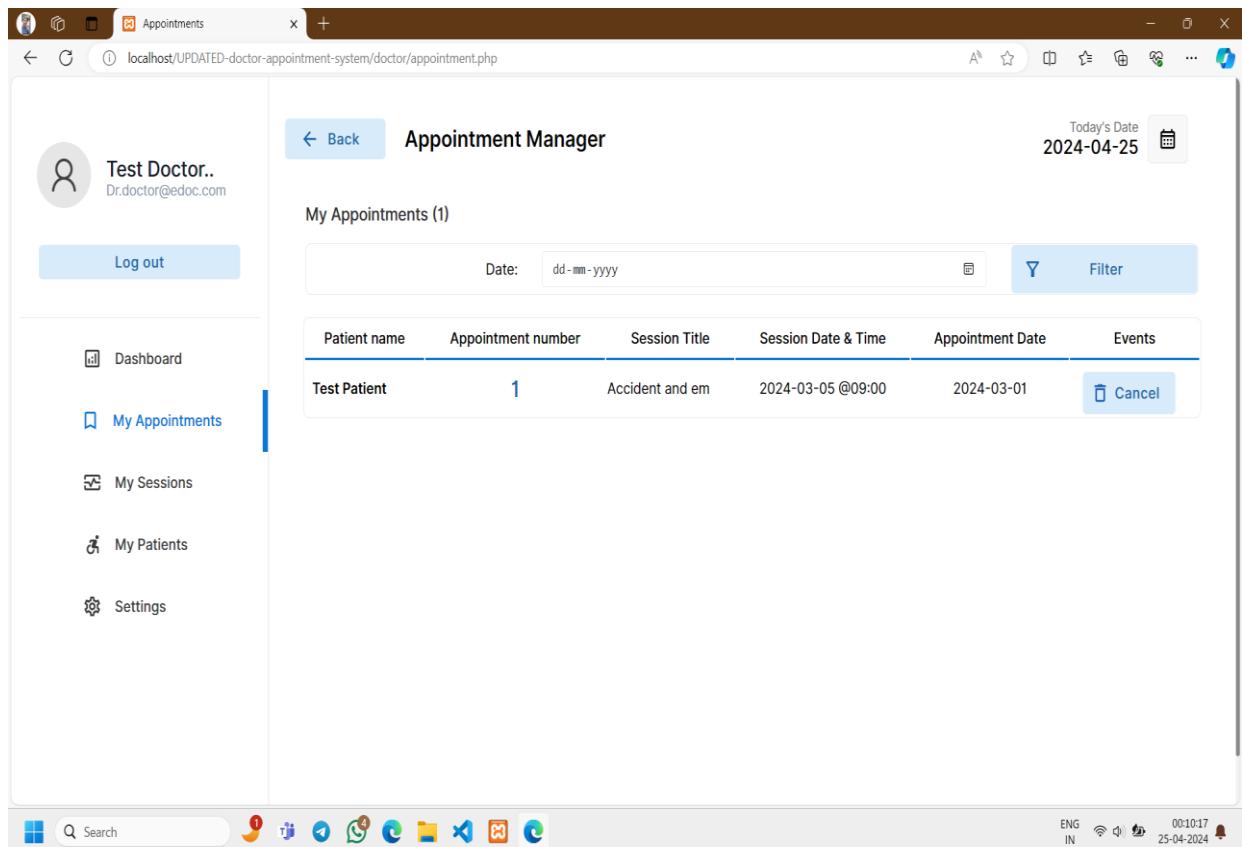
➤ Doctor Dashboard



Screenshot 9.2.2.2 Doctor Dashboard

- After login to doctor panel doctor dashboard open
- Doctor dashboard contain menu bar, today's date, status of doctor, upcoming appointment session
- Menu bar contain my appointment, my session, my patient and setting activity
- Status contain overall record in short total doctors, total patient, total booking and total sessions
- In upcoming session show session up to next week

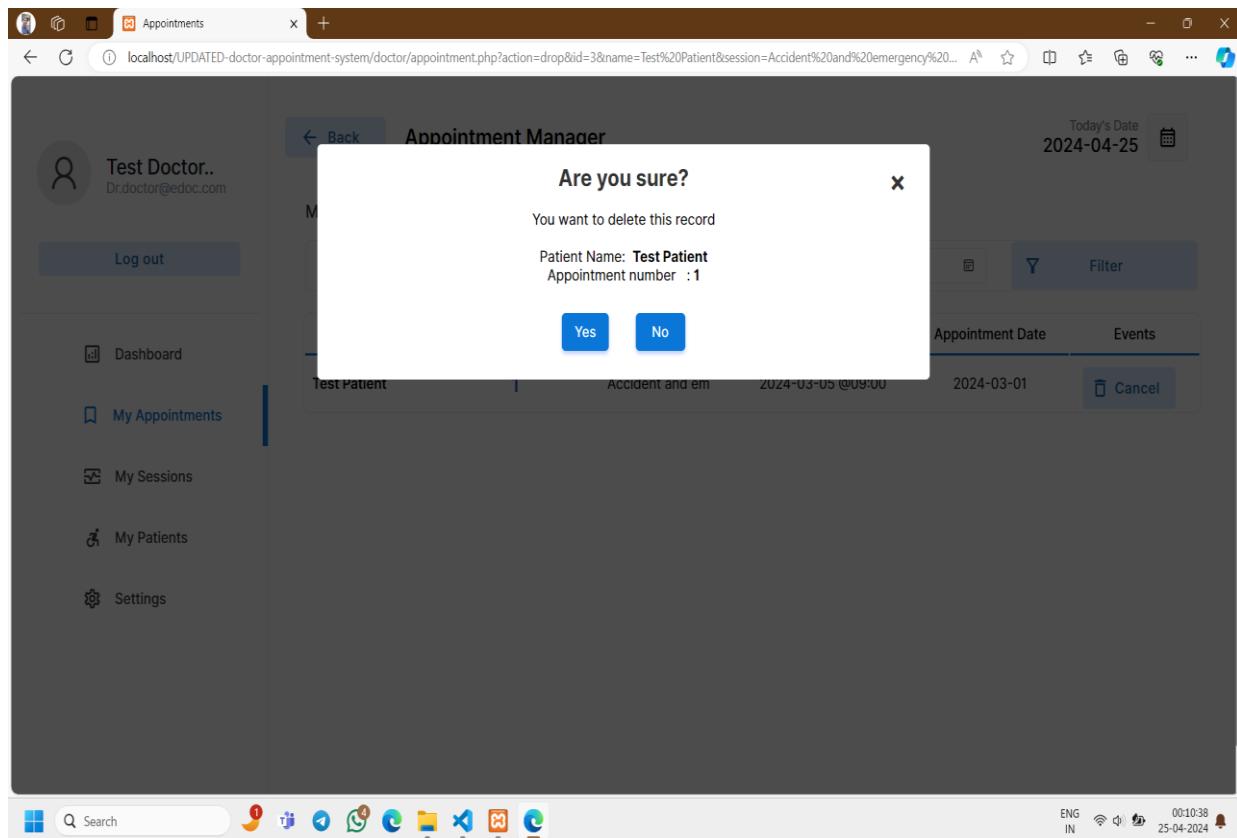
➤ My Appointment Activity (Appointment Manager)



Screenshot 9.2.2.3 My Appointment Activity (Appointment Manager)

- After click on my appointment its open appointment management page
- On the appointment management page there is date on right upper corner
- Below that there is search bar for the search date of appointment
- Below that there is block for all booked appointment for that particular doctor
- In this block there is patient name, appointment number, session title, session date & time, appointment date

➤ Appointment Manager Activity (delete Appointment)



Screenshot 9.2.2.4 Appointment Manager (Delete Appointment)

- After click on cancel button its open popup tab for conformation of cancel the patient appointment booking
- Its gives two option for the user, user can select “yes” or “no” for the conform it

➤ My Session Activity (Session Manager)

The screenshot shows the 'My Sessions' section of the doctor's dashboard. The left sidebar includes links for Dashboard, My Appointments, My Sessions (which is selected), My Patients, and Settings. The main area displays 'My Sessions (2)' with the following details:

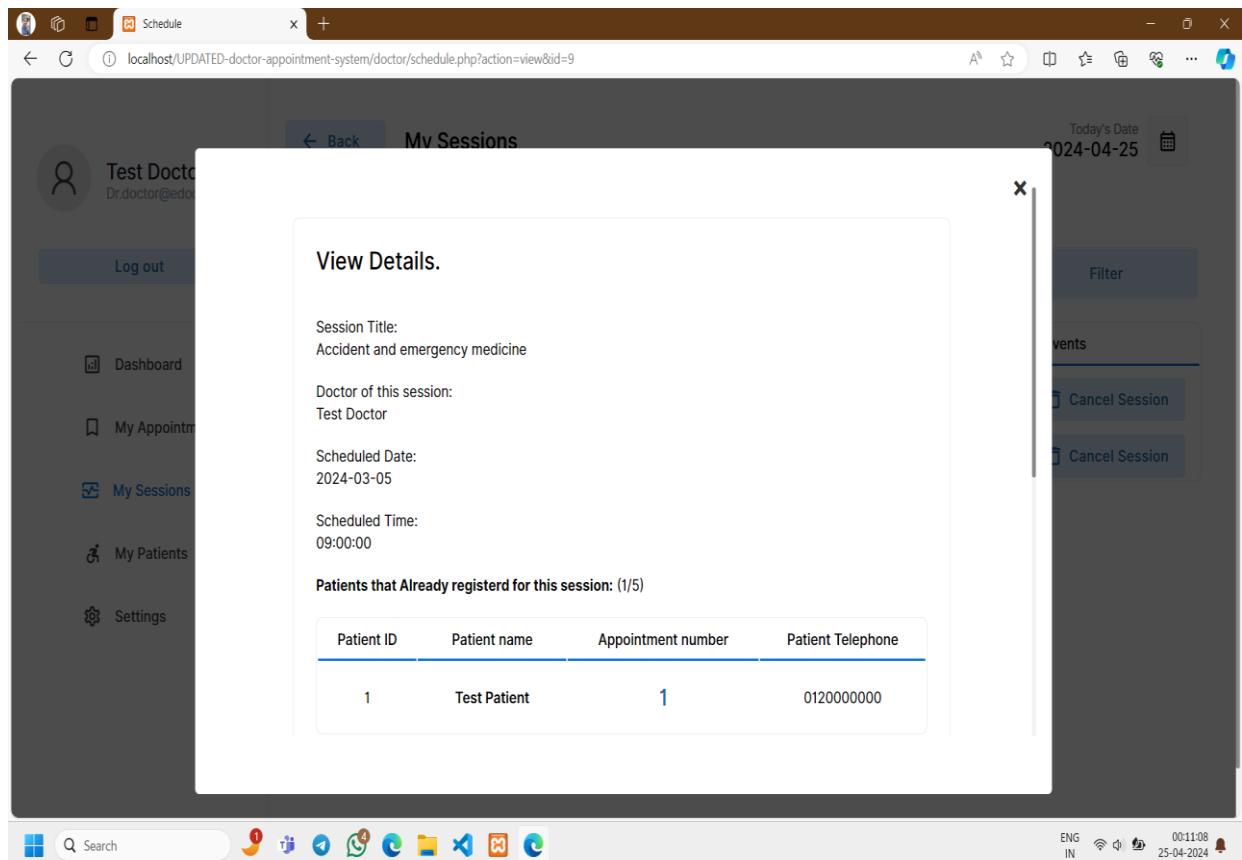
Session Title	Scheduled Date & Time	Max num that can be booked	Events
Accident and emergency medicin	2024-03-05 09:00	5	View Cancel Session
Accident and emergency medicin	2024-04-30 10:00	5	View Cancel Session

At the top right, it shows 'Today's Date' as 2024-04-25. The bottom status bar shows system information like ENG IN, battery level, and the date 25-04-2024.

Screenshot 9.2.2.5 My Session Activity (Session Manager)

- When we move to forward in menu bar of doctor dashboard there is my session activity
- In this session activity there is date at right top corner, below that there is search bar for date to search directly for the session
- Below that there is section of details of session title, scheduled date & time, and event like view and cancel session

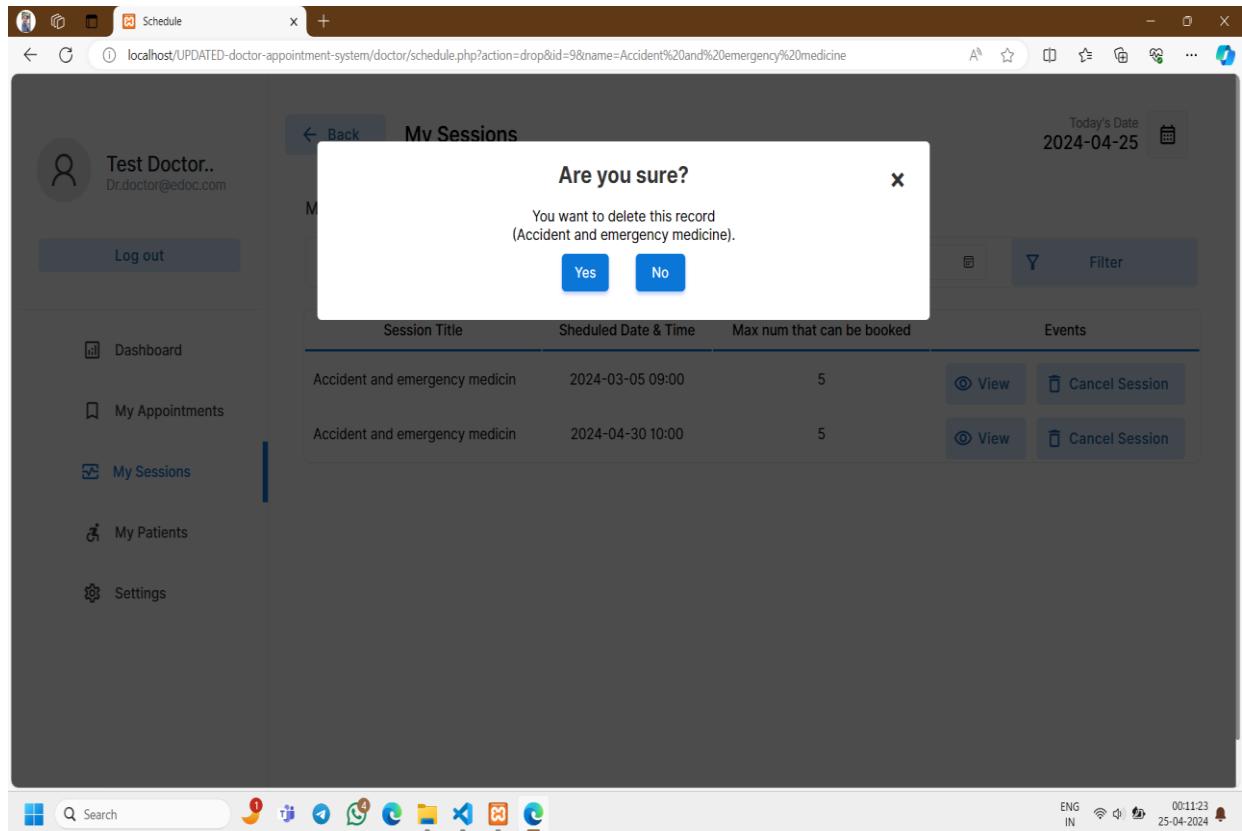
➤ **My Session Activity (View Details)**



Screenshot 9.2.2.6 My Session Activity (View Session)

- After click on the view button of event in my session its open the popup tab for view details of that particular session
- Within that tab there are the session title, doctor name for that section, scheduled date and list of patients of who booked that session

➤ My Session Activity (Delete Session)



Screenshot 9.2.2.7 My Session Activity (Delete Session)

- After click on cancel session button of event in my session activity its open the popup tab for the delete the record of that session
- On that tab there is “yes” or “no” option for the conformation

➤ My Patients Activity (My Patients)

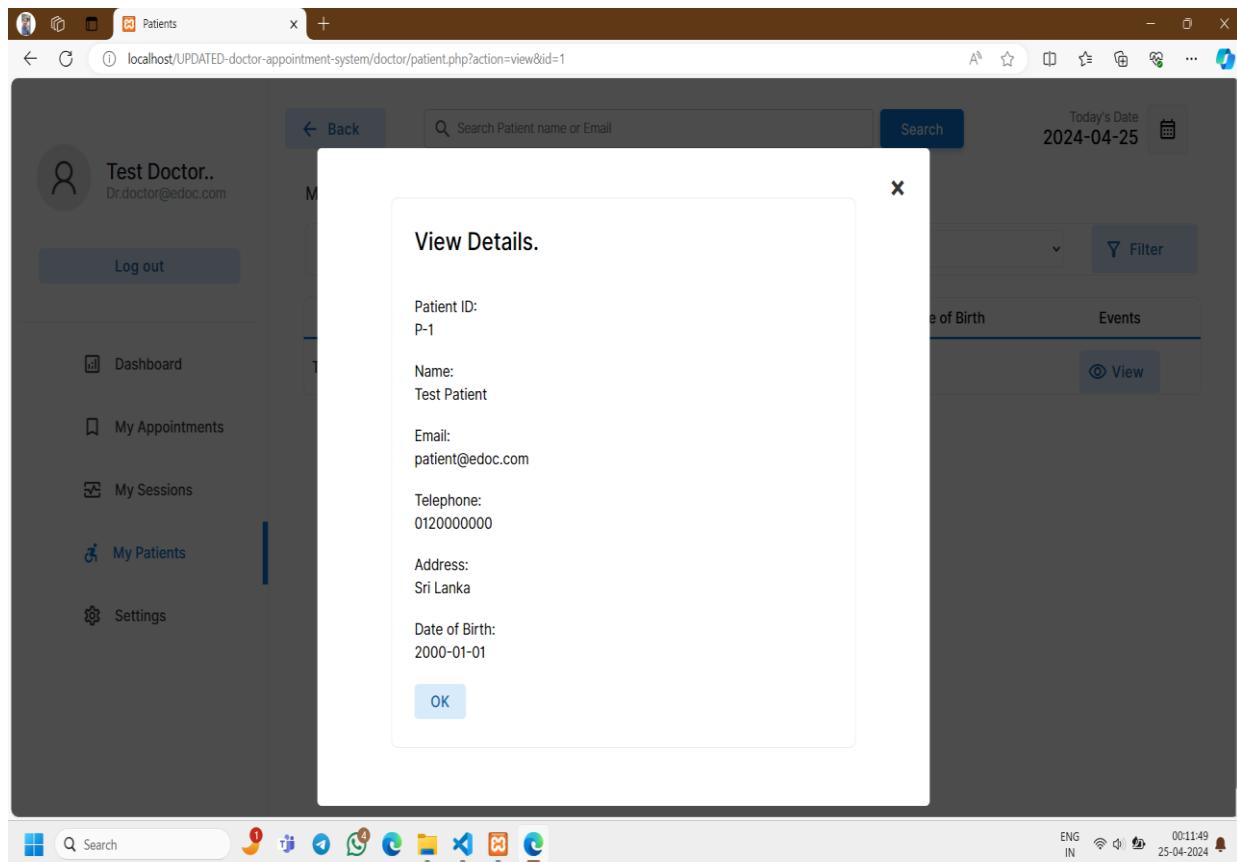
The screenshot shows the 'My Patients' section of the doctor's dashboard. At the top, there is a search bar with the placeholder 'Search Patient name or Email'. To the right of the search bar is a 'Search' button and a date field showing 'Today's Date 2024-04-25'. Below the search bar, the heading 'My Patients (1)' is displayed. Underneath this, there is a table with columns: Name, Telephone, Email, Date of Birth, and Events. One row is visible for 'Test Patient' with details: Telephone 0120000000, Email patient@edoc.com, Date of Birth 2000-01-01, and an 'Events' column containing a 'View' link. Above the table, there is a dropdown menu labeled 'Show Details About: My patients Only'. On the far left, a sidebar lists navigation options: Dashboard, My Appointments, My Sessions, My Patients (selected), and Settings. The 'My Patients' option is highlighted with a blue underline.

Name	Telephone	Email	Date of Birth	Events
Test Patient	0120000000	patient@edoc.com	2000-01-01	View

Screenshot 9.2.2.8 My Patient Activity (My Patient)

- When we move to forward in menu bar of doctor dashboard there is my patient activity
- In this activity there is date at right top corner, front of that there is search bar for search directly through patient name or email
- Below that there is section of details of patient name, telephone number, email id, date of birth and event session

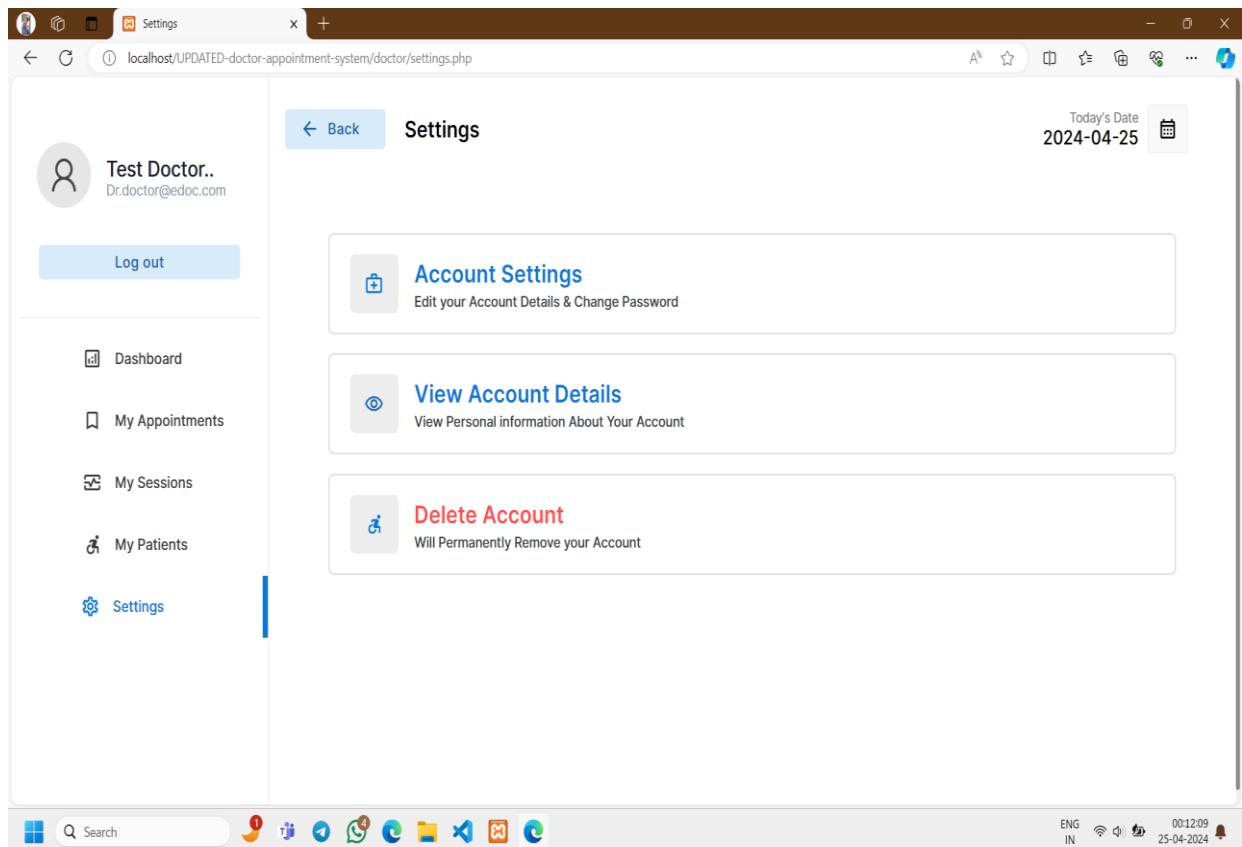
➤ My Patients Activity (View Patient Details)



Screenshot 9.2.2.9 My Patient Activity (View Patient Details)

- After click on the view button of event section in the patient activity of doctor panel its open the popup tab for the view of details for particular patient from that list
- In that tab there is patient id, patient name, email, telephone number, address and date of birth

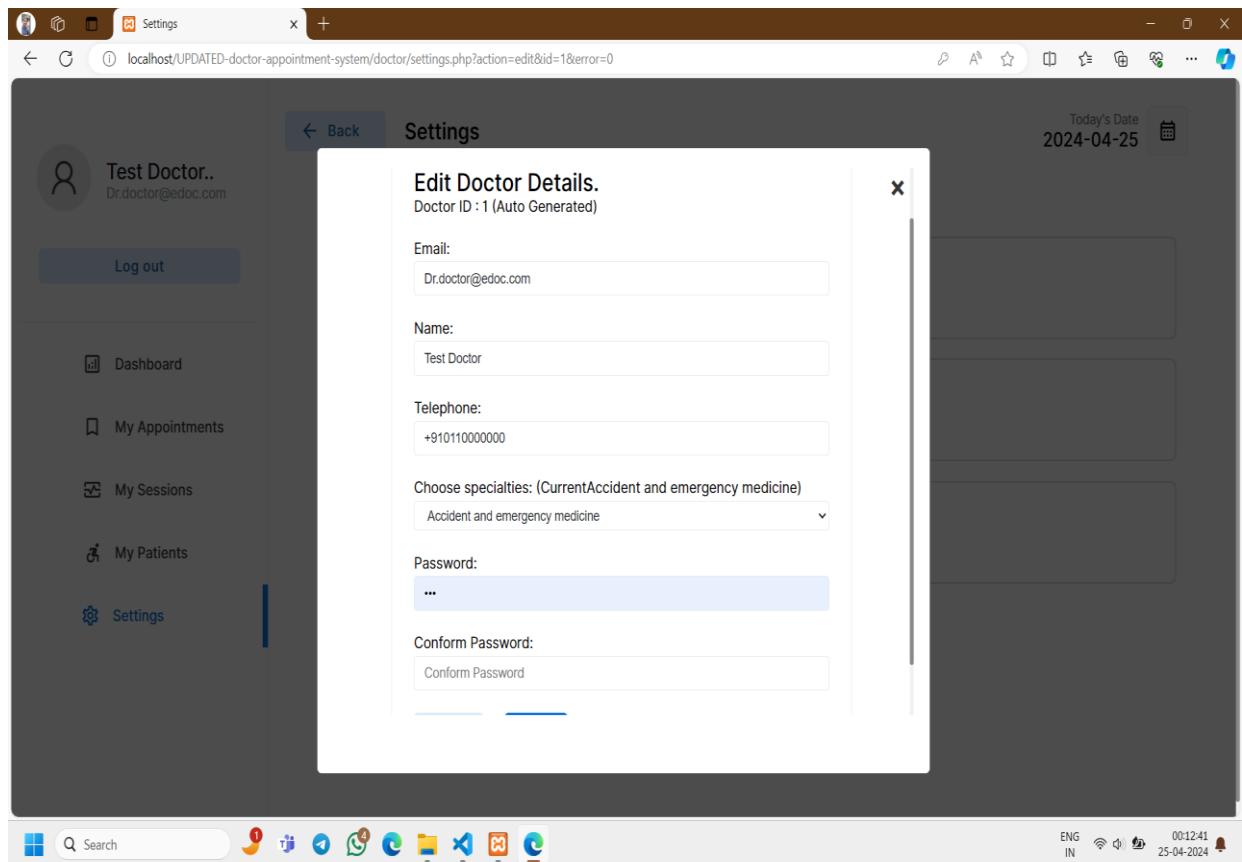
➤ Settings Activity



Screenshot 9.2.2.10 Settings Activity

- When we move to forward in menu bar of doctor dashboard there is setting activity
- In this activity there is account setting, view account details and delete account

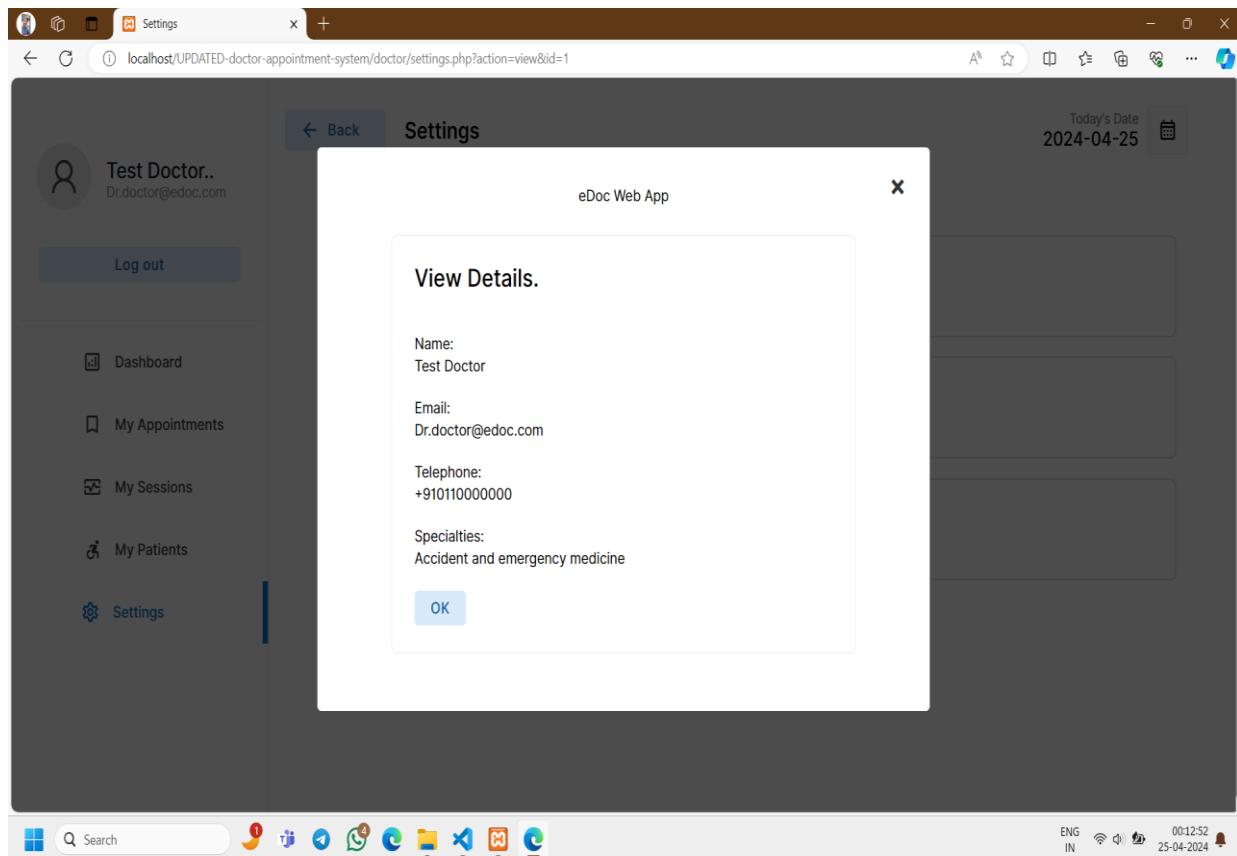
➤ **Settings Activity (Edit Doctor Details)**



Screenshot 9.2.2.11 Settings Activity (Edit Doctor Details)

- After click on the account setting its open popup tab for edit the doctor details
- Doctor can update his record details like email, name, telephone number, specialties and password
- There also two buttons for the cancel and conform the changes

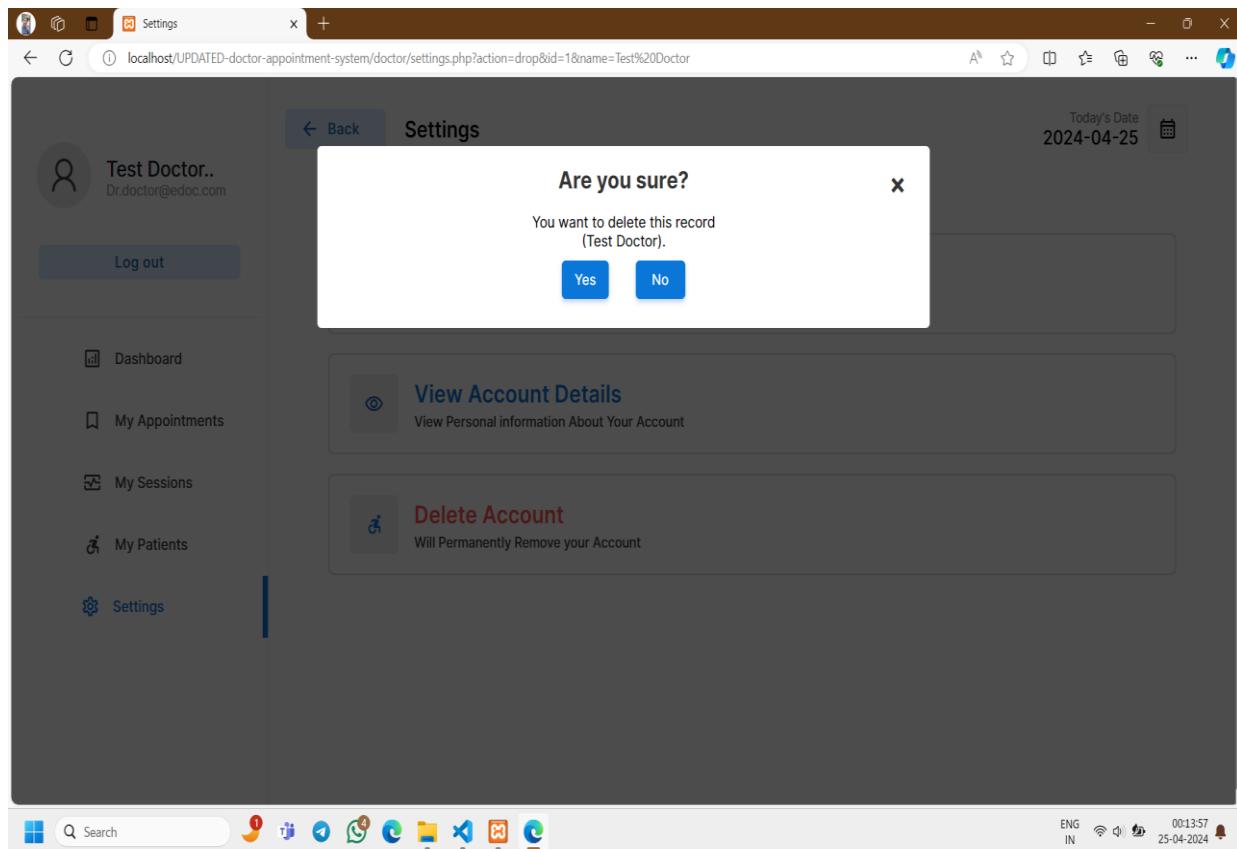
➤ **Settings Activity (View Account Details)**



Screenshot 9.2.2.12 Settings Activity (View Account Details)

- After click on the view account details its open popup tab for view the doctor details
- Doctor can watch his record details like email, name, telephone number, specialties

➤ Settings Activity (Delete Account)

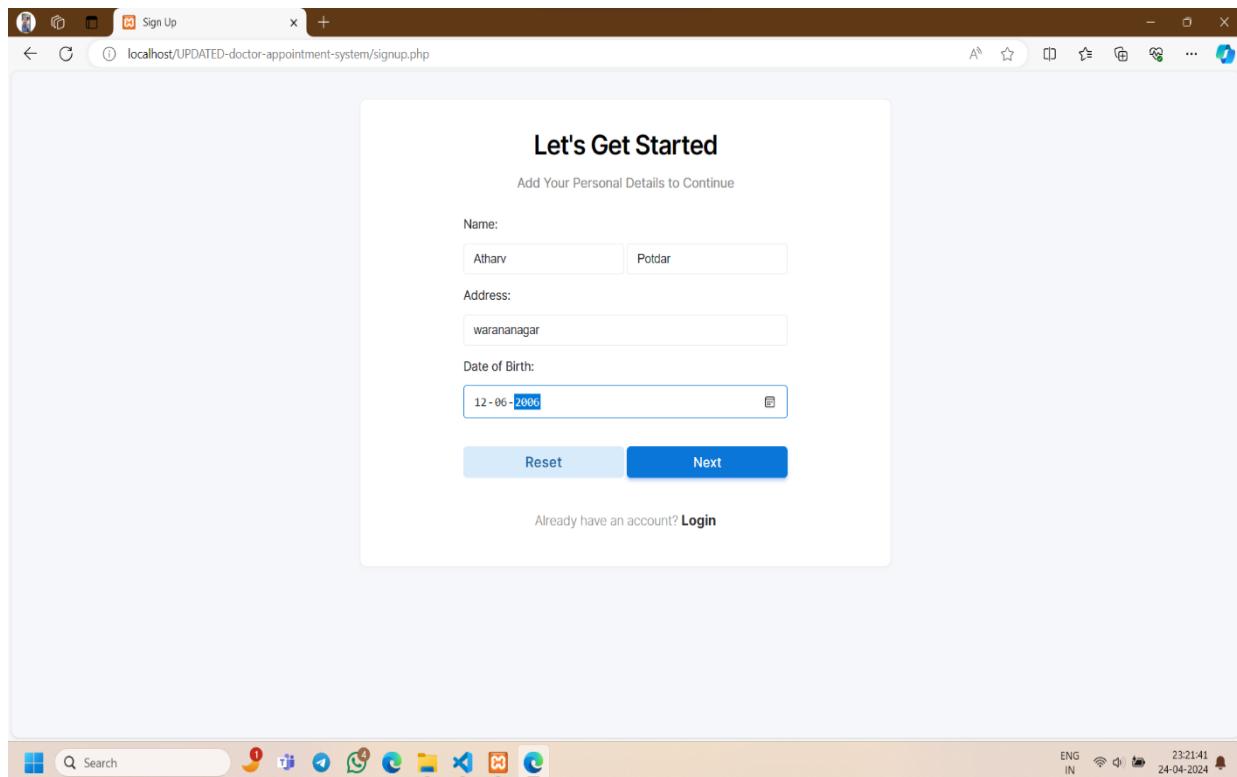


Screenshot 9.2.2.12 Settings Activity (Delete Account)

- After click on the delete account its open popup tab for conform delete account
- Within that tab there is two option for user “yes” or “no” to conformation

9.2.3 Patient Panel

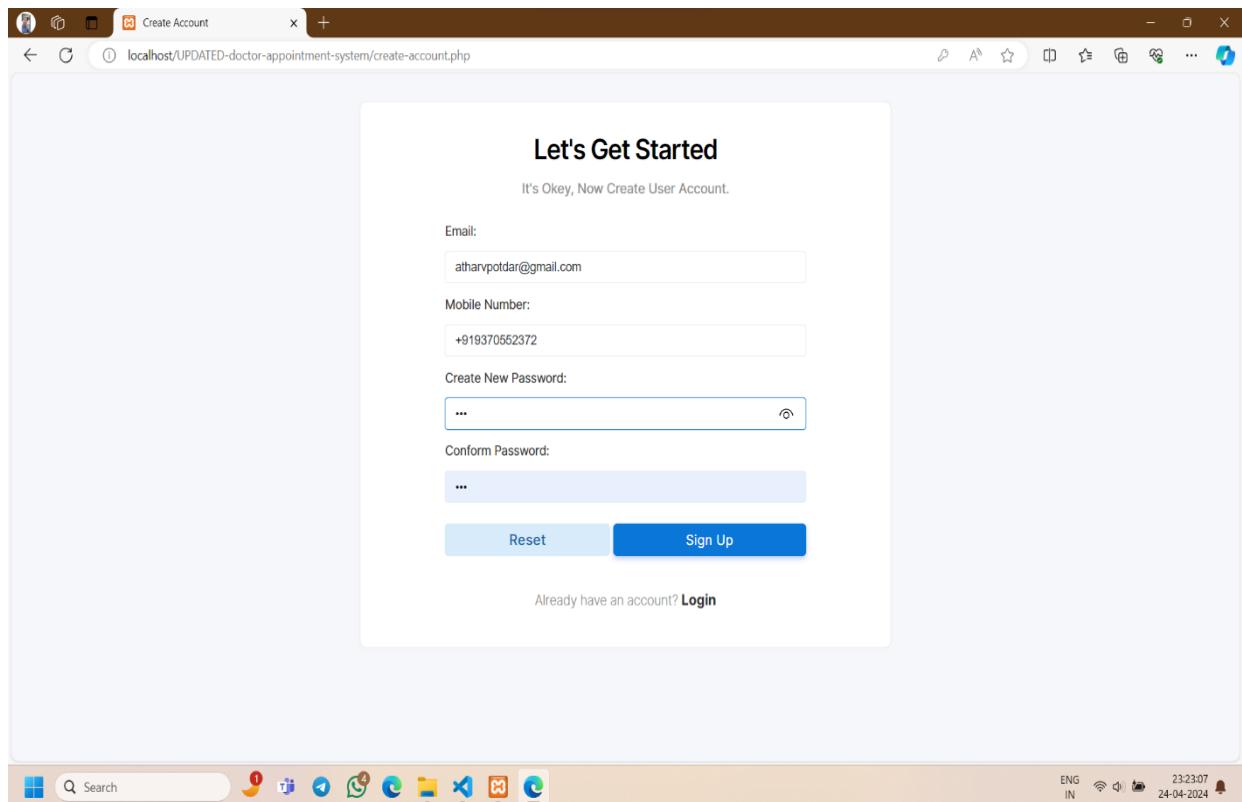
➤ Sign up Page



Screenshot 9.2.3.1 Patient Panel (Sign Up Page)

- After click on sign up button of home page (Dashboard Page) it jumps to sign up process for login in
- On that page there is textbox for name, surname, address, date of birth and two buttons for the reset and continues next process
- There is one more option for direct login to patient dashboard for the existing account

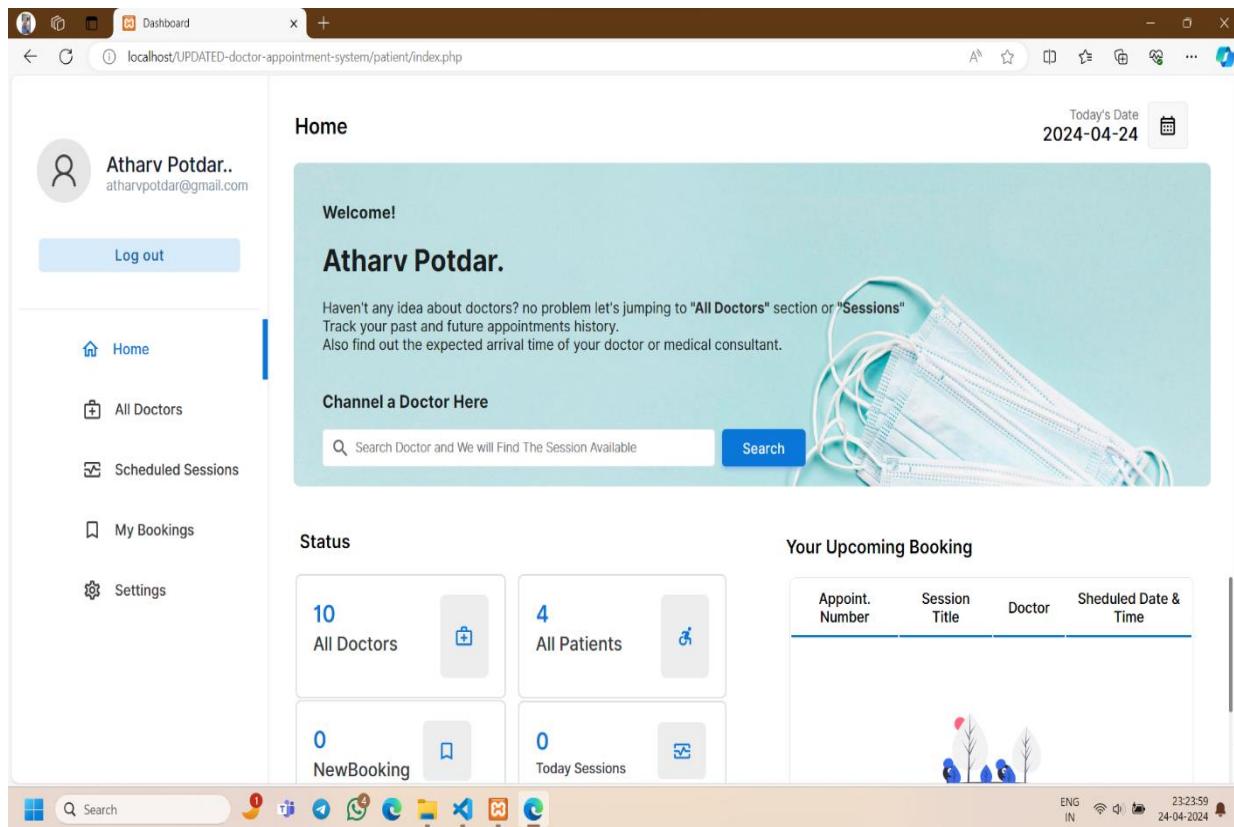
➤ Create User Account



Screenshot 9.2.3.2 Create User Account

- After we continues next process there is section for the create the user(patient) account
- Within this there is textbox for email id, mobile number, password and conform password
- Below that there are two buttons for reset and sign up for create account

➤ Patient Dashboard



Screenshot 9.2.3.3 Patient Dashboard

- After successfully login to patient account its open patient dashboard
- Within that dashboard there is menu bar at left side, date at top side of right corner, below that there is search bar for search doctor, next that there is status which give overall report of all doctor, all patient, total booking appointment, and today sessions
- Front of that there is upcoming appointment if any booked appointment

➤ All Doctors Activity (List of Doctors)

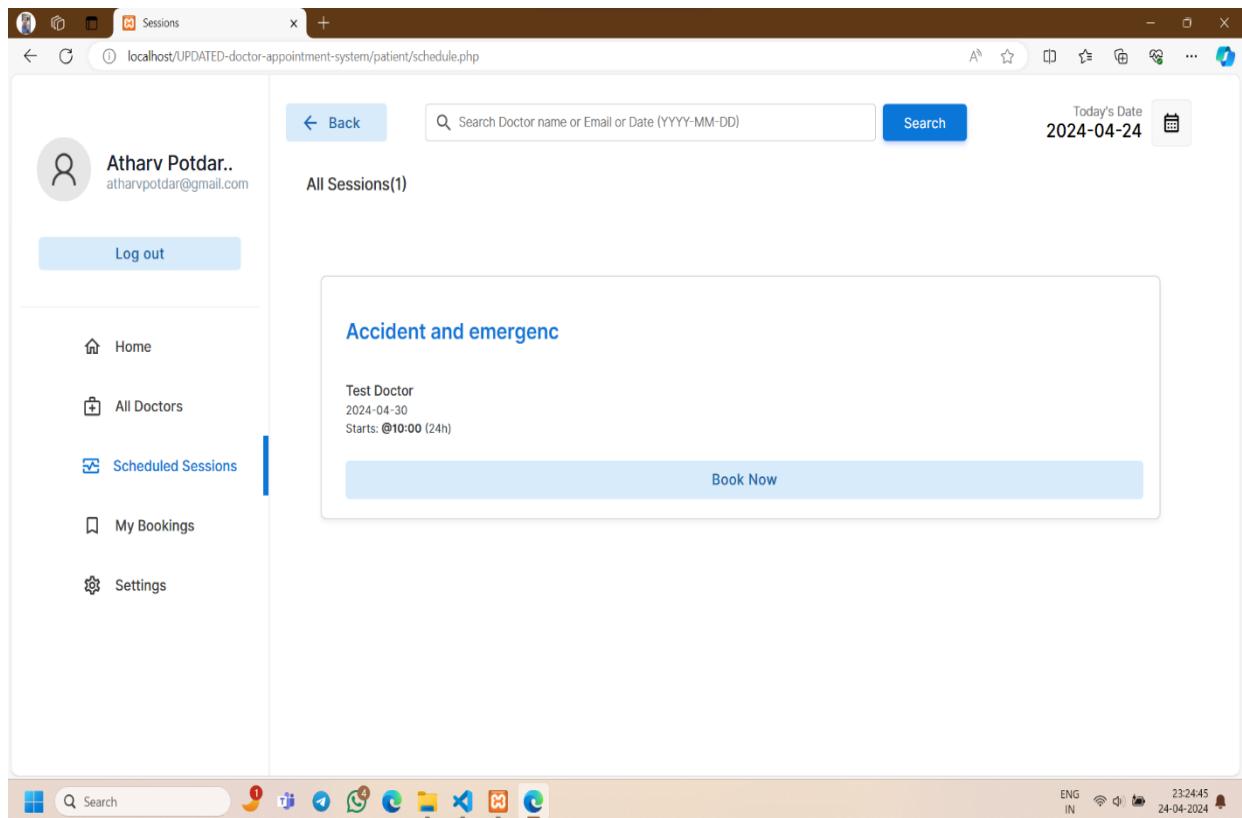
The screenshot shows a web browser window titled "Doctors" with the URL "localhost/UPDATED-doctor-appointment-system/patient/doctors.php". The page header includes a search bar ("Search Doctor name or Email") and a date field ("Today's Date: 2024-04-24"). On the left, a sidebar for the user "Atharv Potdar.." (atharvpotdar@gmail.com) offers links to "Home", "All Doctors" (which is selected and highlighted in blue), "Scheduled Sessions", "My Bookings", and "Settings". The main content area displays a table titled "All Doctors (10)" with the following data:

Doctor Name	Email	Specialties	Events
Miss Dr. Vedhika	Dr.Vedhika@gmail.com	Dental, oral and max	View Sessions
Miss Dr. Anupa	Dr.Anupa@gmail.com	Cardiology	View Sessions
Dr. Sunny	Dr.Sunny@gmail.com	Public health and Pr	View Sessions
Dr. Keran	Dr.Keran@gmail.com	Cardiology	View Sessions
Dr. Adarsh	Dr.Adarsh@gmail.com	Accident and emergen	View Sessions
Dr. Raj	Dr.Raj@edoc.com	General surgery	View Sessions
Dr. Kunal	Dr.kunal000@gmail.co	General surgery	View Sessions
Miss Dr. Komal	Dr.Komal000@gmail.co	Child psychiatry	View Sessions
Miss Dr. Arpita	Dr.kolearpita000@gma	Dental, oral and max	View Sessions

Screenshot 9.2.3.4 All Doctors Activity (List of Doctors)

- After click on the all doctors form menu bar its open the page of listed doctors
- On that page there is date at top right convers
- Below that there is chart for the present doctors in this chart doctors name, email id, specialties and event like view and session

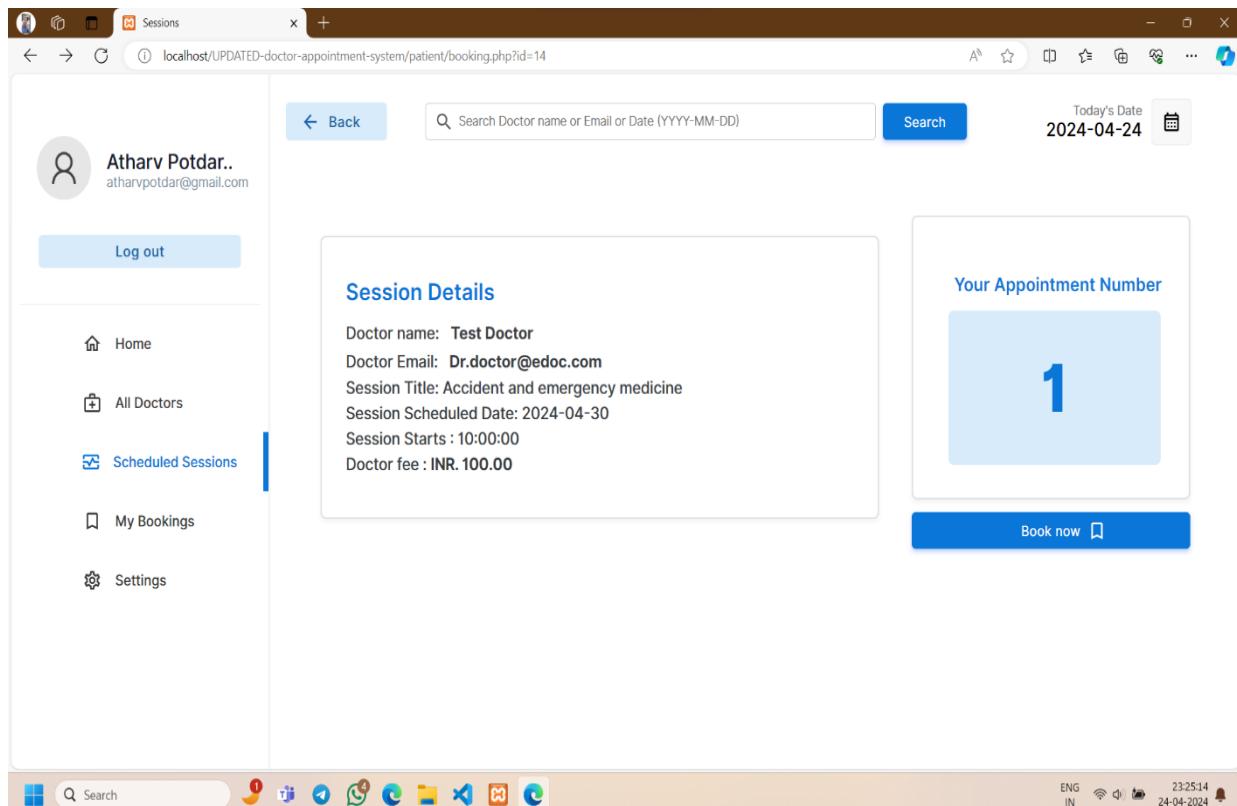
➤ Scheduled Session Activity (All Sessions)



Screenshot 9.2.3.5 Scheduled Session Activity (All Sessions)

- After click on the scheduled session form menu bar its open the page of listed sessions of all doctors
- On that page there is date at top right convers and search bar for the doctor name, email id and date
- Below that there is all session of all doctors this session contains session title, doctors name, date & time and one button for book appointment

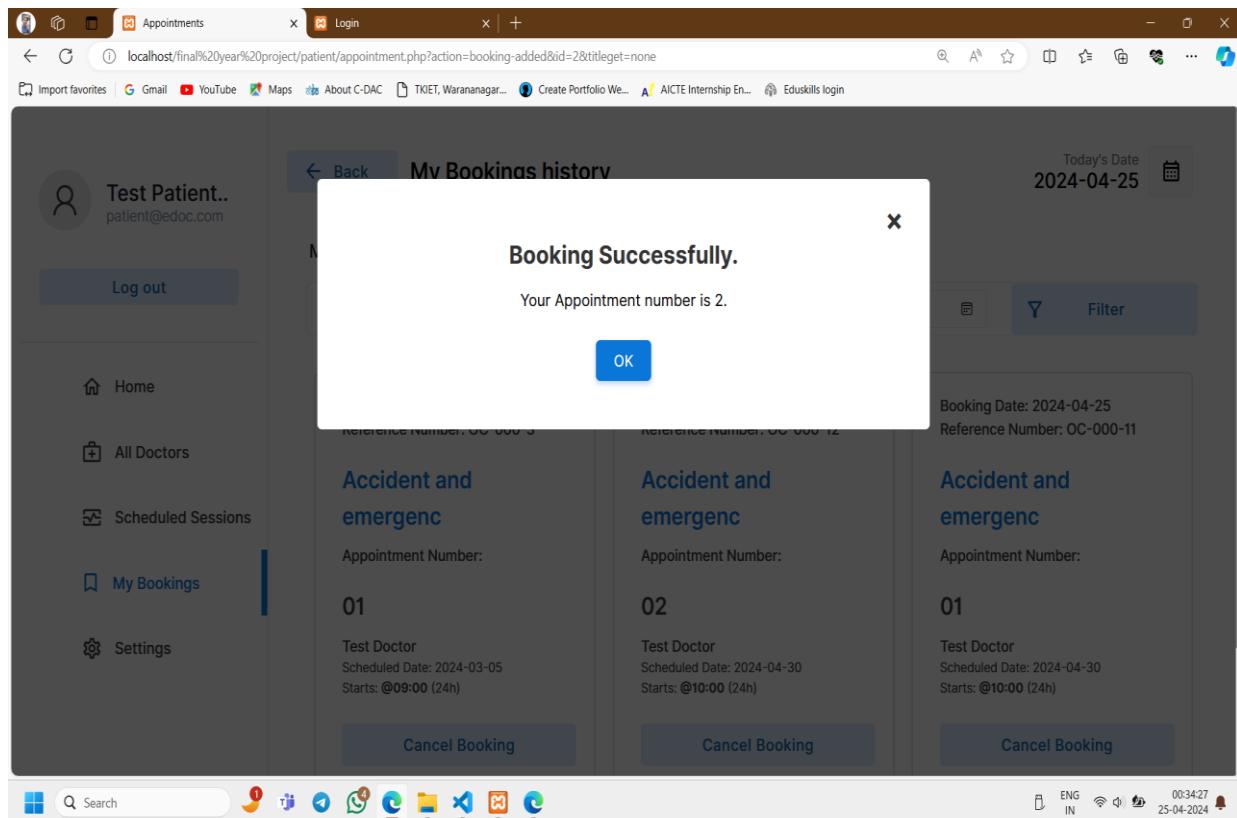
➤ Scheduled Session Activity (Sessions Details)



Screenshot 9.2.3.6 Scheduled Session Activity (Sessions Details)

- After click on book now it's open that particular session details
- Within this there is doctor name, doctor email, session title, schedule date, time and doctors fee
- At front of that there is appointment number and one button for book now appointment

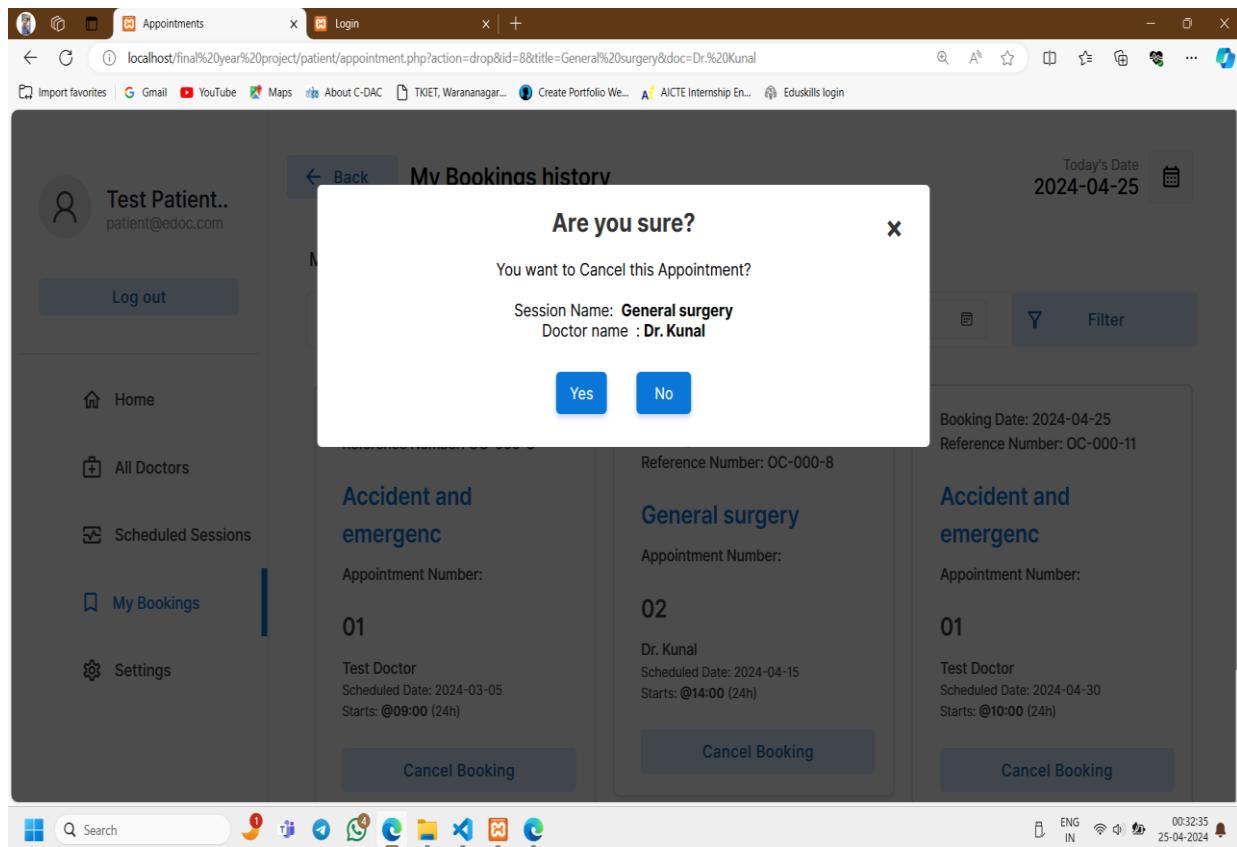
➤ My Booking Activity (Booking History)



Screenshot 9.2.3.7 My Booking Activity (Booking History)

- After click on the book now it gives the popup notification of conform booking with the appointment number
- And move to my booking activity

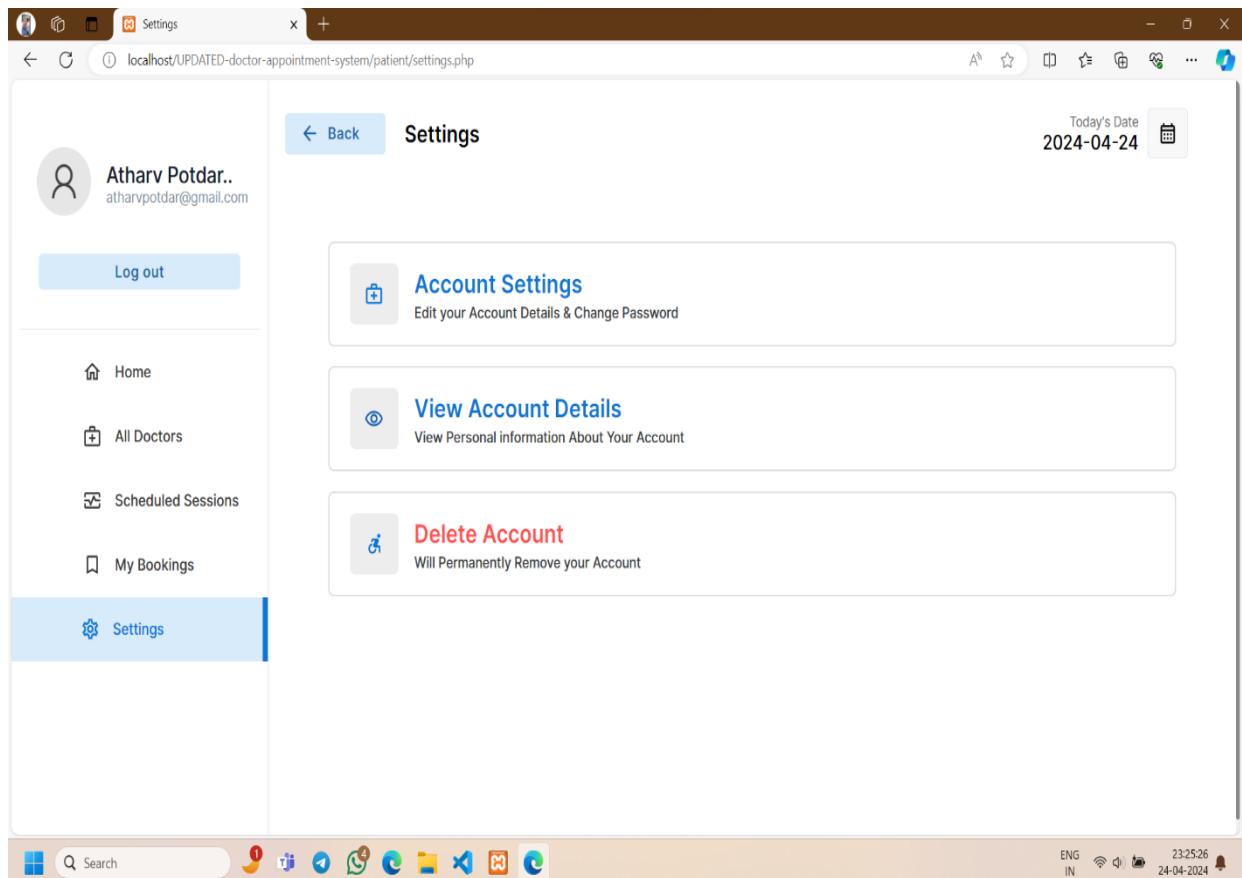
➤ My Booking Activity (Cancel Booking)



Screenshot 9.2.3.8 My Booking Activity (Cancel Booking)

- After conform booking it jump to my booking activity section page
- On that page there is history of all booking with their details
- After click on the cancel booking it gives the popup notification of conform cancel booking with the doctor name and session title
- There are also two buttons for “yes” or “no” to the conformation

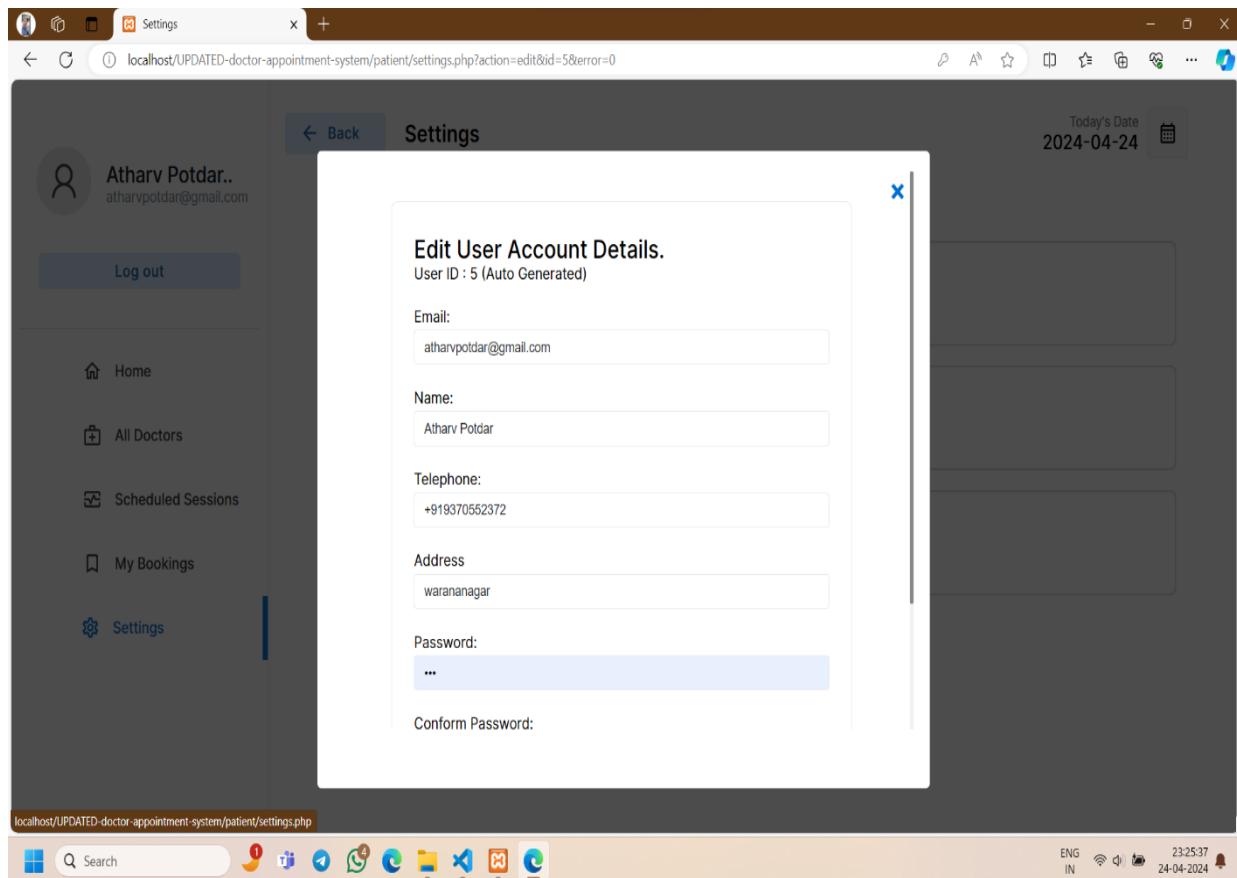
➤ Settings Activity



Screenshot 9.2.3.9 Settings Activity

- After move forward to the settings it open setting activity
- Within this page there is date it top right convers and below that account setting, view account details and delete account tabs

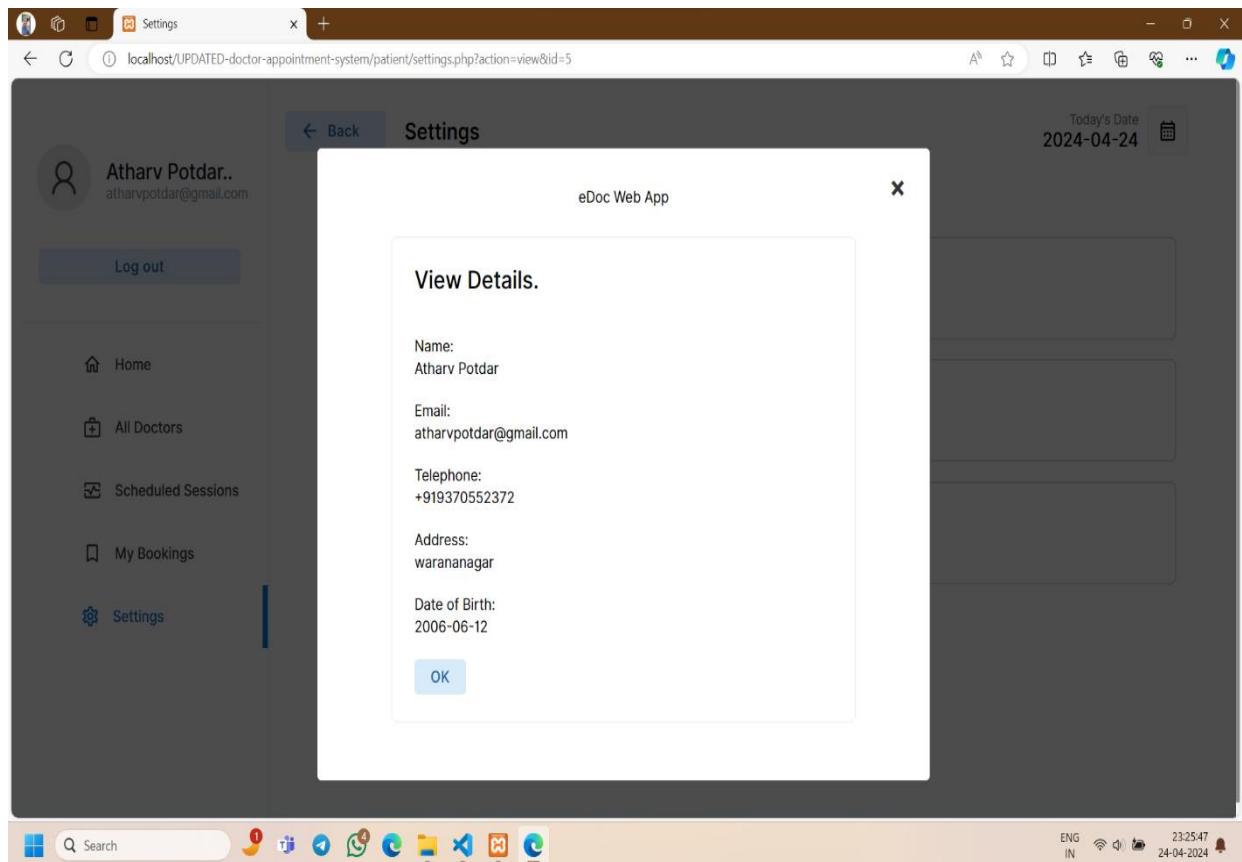
➤ Settings Activity (Account Setting)



Screenshot 9.2.3.10 Settings Activity (Account Setting)

- After click on account setting it open popup tab for edit user account details
- In that tab there is email id, name, telephone number, address, password and one button for conform details

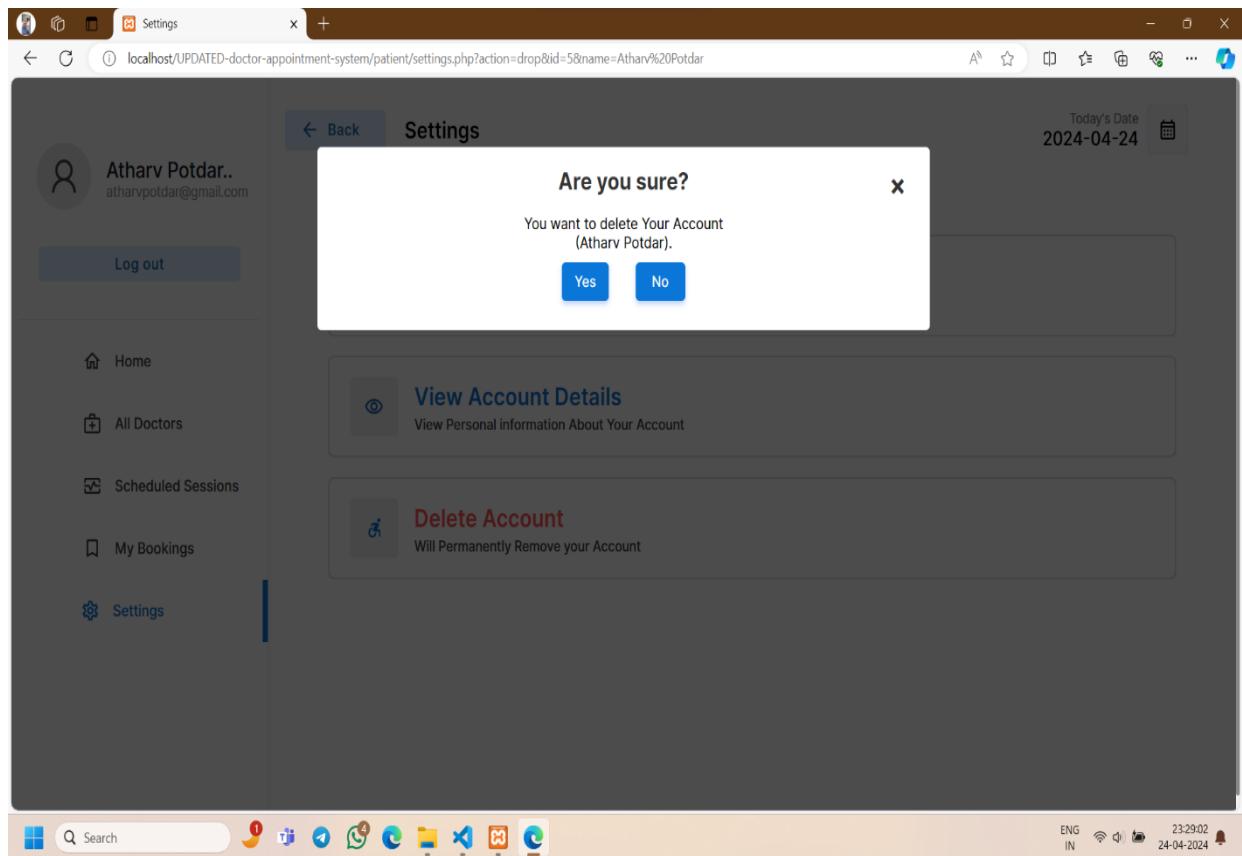
➤ **Settings Activity (View Account Details)**



Screenshot 9.2.3.11 Settings Activity (View Account Details)

- After click on view account details its open popup tab for view user account details
- In that tab there is email id, name, telephone number, address, date of birth and one button for conform details

➤ **Settings Activity (Delete Account Details)**



Screenshot 9.2.3.12 Settings Activity (Delete Account Details)

- After click on delete account its open popup tab for delete account details
- After click on that tab there is popup notification for conform it and there is also two buttons for the “yes” or “no” to conform delete account

Chapter 10

Conclusion and Future Scope

10.1 Conclusion

The development of an online doctor appointment system represents a pivotal advancement in healthcare technology, with far-reaching implications for patients, healthcare providers, and the healthcare industry as a whole. Through the convergence of digital innovation and medical practice, this project aims to streamline the cumbersome process of scheduling doctor appointments, enhancing accessibility, efficiency, and overall patient experience.

At its core, the online doctor appointment system serves as a catalyst for patient empowerment, enabling individuals to take control of their healthcare journey with greater ease and convenience. By providing a user-friendly platform accessible via web or mobile devices, patients can effortlessly browse through a comprehensive directory of healthcare professionals, view their profiles, specialties, and availability, and book appointments at their preferred time slots—all from the comfort of their homes or on the go. This not only eliminates the need for tedious phone calls and physical visits but also mitigates the risk of missed appointments and scheduling conflicts, thereby optimizing resource utilization and maximizing patient-provider interaction.

Moreover, the system fosters a paradigm shift in the doctor-patient relationship, fostering greater communication, transparency, and trust. Patients gain access to relevant information about their healthcare providers, including their qualifications, credentials, and patient reviews, empowering them to make informed decisions tailored to their unique needs and preferences. Additionally, built-in features such as appointment reminders, automated notifications, and real-time updates keep patients informed and engaged throughout their healthcare journey, promoting adherence to treatment plans and fostering continuity of care.

From the perspective of healthcare providers, the online appointment system represents a transformative tool for practice management and patient engagement. By digitizing the appointment scheduling process, healthcare facilities can streamline administrative tasks, optimize appointment scheduling, and allocate resources more effectively, leading to improved operational efficiency and cost savings. Furthermore, the system facilitates better patient flow management, enabling providers to anticipate patient volumes, allocate staff resources

accordingly, and minimize wait times, thus enhancing patient satisfaction and retention.

Beyond its immediate benefits for patients and healthcare providers, the online doctor appointment system holds immense potential to drive systemic improvements in the healthcare ecosystem. By harnessing data analytics and machine learning algorithms, the system can generate valuable insights into patient preferences, healthcare trends, and resource utilization patterns, empowering stakeholders to make data-driven decisions, optimize service delivery, and enhance healthcare outcomes. Moreover, the aggregation of anonymized patient data offers opportunities for population health management, epidemiological research, and public health interventions, contributing to the advancement of evidence-based medicine and preventive healthcare initiatives.

In conclusion, the development and implementation of an online doctor appointment system represent a pivotal step towards the realization of a patient-centric, digitally enabled healthcare ecosystem. By leveraging technology to overcome traditional barriers to access and efficiency, this project not only enhances the patient experience but also empowers healthcare providers to deliver high-quality care in a cost-effective and sustainable manner. As we continue to navigate the evolving landscape of healthcare, the online doctor appointment system stands as a testament to the transformative power of innovation in shaping the future of healthcare delivery.

10.2 Future scope

In envisioning the deep future scope for an online doctor appointment system, we can anticipate a range of advancements and enhancements that could revolutionize healthcare delivery and patient experience. Here's a speculative look into what the future might hold:

- **Integration of AI and Machine Learning:** As AI and machine learning continue to advance, future online doctor appointment systems could incorporate intelligent algorithms to assist patients in diagnosing common ailments, triage appointments based on urgency, and even predict potential health issues based on historical data and genetic information.
- **Virtual Reality Consultations:** With the proliferation of virtual reality (VR) technology, patients could have immersive virtual appointments with healthcare providers. VR consultations could offer a more lifelike interaction, enabling doctors to better assess patients' conditions remotely and provide more personalized care.
- **Telemedicine and Remote Monitoring:** Online doctor appointment systems could evolve into comprehensive telemedicine platforms, allowing patients to consult with specialists

from anywhere in the world. Additionally, remote monitoring devices and wearable sensors could seamlessly integrate with these systems, enabling real-time health monitoring and proactive interventions.

- **Blockchain for Health Records:** To address concerns about data security and interoperability, future systems might leverage blockchain technology to securely store and share patients' health records. Blockchain-based health records could ensure data integrity, enhance patient privacy, and facilitate seamless access to medical information across different healthcare providers.
- **Predictive Analytics for Population Health Management:** By analyzing large datasets of anonymized patient information, online doctor appointment systems could help identify trends and patterns in population health. Predictive analytics algorithms could forecast disease outbreaks, assess the effectiveness of public health interventions, and inform healthcare policy decisions.
- **Personalized Medicine and Genomic Profiling:** As the cost of genomic sequencing continues to decline, online appointment systems could incorporate genetic profiling to deliver personalized medicine. By analyzing patients' genetic makeup, doctors could tailor treatments and preventive measures to individual genetic predispositions, optimizing health outcomes and reducing the risk of adverse reactions to medications.
- **Augmented Reality for Surgical Planning and Training:** Surgeons could use augmented reality (AR) technology to visualize patient anatomy in three dimensions, allowing for more precise surgical planning and intraoperative guidance. Moreover, AR-based training modules could provide aspiring surgeons with realistic simulations and hands-on practice opportunities, enhancing surgical skills and improving patient safety.
- **Health Chatbots and Virtual Assistants:** Intelligent chatbots and virtual assistants could serve as the first point of contact for patients seeking medical advice. These conversational AI systems could answer common health-related questions, schedule appointments, and provide personalized recommendations based on patients' symptoms and medical history.
- **Collaborative Care Networks:** Online doctor appointment systems could facilitate seamless collaboration among healthcare providers within integrated care networks. Through secure messaging, shared care plans, and multidisciplinary consultations, providers could work together to coordinate patient care more effectively and achieve better health outcomes.

- **Continuous Innovation and User-Centric Design:** In the deep future, online doctor appointment systems must remain agile and responsive to evolving healthcare needs and technological advancements. Continuous innovation, user-centric design principles, and ongoing stakeholder feedback will be essential for ensuring that these systems continue to enhance access to quality care, improve patient experiences, and drive positive health outcomes for all.

Referred Journal/Conference Papers –

[1] Dependable online appointment booking system for NHIS outpatient in Nigerian teaching hospitals by Adebayo Peter Idowu, Olajide Olusegun Adeosun, Kehinde Oladipo Williams -In this fast-driven society, where the climate in the healthcare sector demands efficiency and patient's satisfaction and medical care delivery.

2.3.1 Web Reference –

[1] Apache Official Website: <https://httpd.apache.org/>

[2] MySQL official Website: <https://www.mysql.com/>

[3] Visual Studio Code Official Website: <https://code.visualstudio.com/>

[4] Microsoft's Web Development Documentation:

<https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-5.0>