

Project Report on

Patient doctor admin management in hospital

A Project Work

By

Saswata Chakraborty (001910503004)
149867 of 2019-20
Exam roll number : MCA226004

In partial fulfillment of requirements for the award of
degree Master of Computer Application
in Computer Science and Engineering(2022)

Under the Guidance of

Dr. Diganta Saha
Department of
Computer Science and
Engineering
Jadavpur University

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

Certificate of Recommendation

This is to certify that the dissertation entitled “**Patient doctor admin management in hospital**” has been carried out by **Saswata Chakraborty** (University Registration No.: 149867 of 2019-20, Examination Roll No.: **MCA226004**) under my guidance and supervision and be accepted in partial fulfilment of the requirement for the Degree of Master of Computer Application.

.....

Dr. Diganta Saha (Thesis Supervisor)

Department of Computer Science and Engineering

Jadavpur University, Kolkata-32

Countersigned

.....

Prof. Anupam Sinha

Head, Department of Computer Science and Engineering,

Jadavpur University, Kolkata-32.

.....

Prof. Chandan Majumdar

Dean, Faculty of Engineering and Technology,

Jadavpur University, Kolkata-32.

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

Certificate of Approval

This is to certify that the thesis entitled “**Patient doctor admin management in hospital**” is a Bonafede record of work carried out by **Saswata Chakraborty** in partial fulfilment of the requirements for the award of the degree of Master of Computer Application in the Department of Computer Science and Engineering, Jadavpur University during the period of September 2019 to July 2022. It is understood that by this approval the undersigned does not necessarily endorse or approve any statement made, opinion expressed, or conclusion drawn therein but approves the thesis only for the purpose for which it has been submitted.

.....

Signature of Examiner 1

Date:

.....

Signature of Examiner 2

Date:

*Only in case the thesis is approved

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

Declaration of Originality and Compliance of Academic Ethics

I hereby declare that this thesis entitled “**Patient doctor admin management in hospital**” contains a literature survey and original research work by the undersigned candidate, as part of his degree in Master of Computer Application.

All information has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Saswata Chakraborty

Registration No: 149867 of 2019-20

Examination Roll No.: MCA226004

Thesis Title: **Patient doctor admin management in hospital**

.....

Signature with Date

Acknowledgement

I would like to start by thanking the holy trinity for helping me deploy all the right resources and for shaping me into a better human being. I would like to express my deepest gratitude to my advisor, **Dr. Diganta Saha**, Department of Computer Science and Engineering, Jadavpur University for his admirable guidance, care, patience and for providing me with an excellent atmosphere for doing research. Our numerous scientific discussions and his many constructive comments have greatly improved this work.

I take this opportunity to acknowledge indebtedness and deep sense of gratitude to Mr **Subhendu Saha**, whose valuable guidance and kind supervision throughout the course shaped the present work as it shows.

I am also thankful to my project partner **Ms. Sougata Banerjee** who was always present to motivate me.

Most importantly none of this would have been possible without the love and support of my family. I extend my thanks to my parents, especially to my mother whose forbearance and whole-hearted support helped this endeavor succeed.

This thesis would not have been completed without the inspiration and support of a number of wonderful individuals — my thanks and appreciation to all of them for being part of this journey and making this thesis possible.

.....
Saswata Chakraborty

Registration No: 149867 of 2019-20

Examination Roll No.: MCA226004

Jadavpur University

TABLE OF CONTENTS

Content	Page No.
Abstract	
1. Introduction	
1.1 General Overview of the Problem	1
1.2 Literature Survey	2
1.3 Problem Definition	3
1.4 Analysis of Problem	3
1.5 Proposed Solution Strategy	3
1.6 SRS	4
2. Organisation of the report	5
3. Design Strategy	
3.1 Architecture Diagram	6
3.2 ER Diagram	7
4. Implementation Details	
4.1 Pseudo Code	8
4.2 Flowchart for all use cases	11
5. Detailed Test Plan	16
6. Results and Discussions	21
7. Summary and Conclusion	
7.1 Summary of Achievements	33

7.2 Main difficulties encountered and how they are tackled	33
7.3 Limitations of the project	33
7.4 Future scope of the project	34
 8. References	 35

LIST OF FIGURES

Figure No.	Figure Name	Page No.
1	Architecture Diagram	6
2	ER Diagram	7
3	Patient Authentication	11
4	Patient seeing all doctors list	11
5	Patient booking appointments	12
6	Patient cancelling appointments	12
7	Patients updating their accounts	13
8	Doctors viewing their appointments	13
9	Admin adding admin	14
10	Admin deleting doctor	14
11	Admin generating report	15
12	Home Page	21
13	Patient Registration Form	21
14	Patient Login Form	22
15	Patient Dashboard	23
16	Doctors List	23
17	Appointment Booking	24
18	Appointment Slots	25
19	Doctor Login Form	26
20	Doctor Dashboard	26
21	Admin Login Form	27
22	Admin Dashboard	28
23	Add Admin Form	29
24	Add Doctor Form	30
25	Generate Report	30

LIST OF TABLES

Table No.	Title Name	Page No.
1	Literature Survey	2
2	Test Cases	16

ABSTRACT

The aim of this project is to design every element during development of the web-based project, a hospital management system, that can handle patients, doctors and admins, along with the detail understanding of the data flow and to make sure that no confusions remain on the customer part regarding the structural design, architecture and data flow for the management system.

The landing page is the home page, where there will be login options for patients, doctors and admins separately. If a patient is registered, he will log in, otherwise he will have to register for the system, and then login. Upon arrival on the patient's dashboard after login, he will see the list of doctors, from which he can book an appointment for a doctor on an available slot, and cancel an upcoming appointment, if allowed. The patient profile can also be updated.

A doctor can login to the system, and upon his arrival on the dashboard, he will see the upcoming appointments for the next 7 days.

An admin can manage (enable and disable) the doctor and add other admins or doctors to the system, and can generate appointment reports, weekly or monthly.

1.INTRODUCTION

1.1 General Overview of the Problem

Since the start, the managers of a lot of doctors in the hospital, are maintaining the doctors' and patients' information in Excel files, along with some unformatted file structures like .txt files and pictures. The management was mostly legacy based. Some of the information for appointments are also taken down manually, on notebook pages and etcetera. Since most of the appointments are mainly made on telephone calls, both to the manager and the doctors themselves, it was getting increasingly difficult for the hospital to maintain all the information in a unified way, and was thereby creating the problem of concurrency and time-slot clashes, thus leading to misunderstandings between the manager and the doctors, as well as the patients.

Hence the requirement of creating a digitized appointment making system, that would contain the patients' and doctors' information in a structured manner, make sure that two appointments with the same doctor do not clash, and let both parties have a look at the schedule that they have upcoming.

The Medical Management System will manage the information of the patients, along with the doctors, who are predefined. The patients can register / log in (if they are already registered) to their respective dashboards, where they can select from a drop-down the specialization of the doctors they want to visit (or can also search doctors by their name). They will then be shown a list of doctors of that specialization (or name).

The patients can book an appointment for any doctor, where the patient will have to put in a date inside the next 7 days, and he / she will be shown the list of available time slots, from which one or more can be booked.

The doctor can access a report of his schedules for the coming 7 days, anytime he wishes for, and update his/her profile.

1.2 Literature Survey

Sl No.	Author	Paper and Publication Details	Findings	Relevance to project
1	Raimonda Staniute and Dmitrij Sesok	Hospital Management System. International journal for Research in Engineering Application & Management (IJREAM), ISSN (2015). [1]	Existing systems are reviewed and their pros and cons are highlighted.	It help us in clear understanding of the project and way to achieve the goal of the project
2	Youssef Mroueh	Medical Management System. In 2015 Global Summit on Computer & Information Technology (GSCIT), pp. 1-6. IEEE, 2015. [2]	It described the use cases of the system.	The working of the system is quite similar to what I want to achieve.
3	Kumaran, Kalai Selvi, Deepak	A Study of Advanced Hospital	How large scale health	The literature about implementation

		Management. in IOSR- JDMS, 2020. [3]	systems operate.	and evaluation of Health Information System is reviewed and summarized.
--	--	---	---------------------	--

Table 1: Literature Survey

1.3 Problem Statement

The main aim of this project is to develop a system that coordinate and integrate all the inherent activities involved in the management and running of a healthcare facility.

In the past all details like patient and doctor details have been stored in legacy formats that are not conventional. There are also many loopholes due to this like appointment redundancy, time slot clashes, and when I look at the security of the existing system I can notice that it is very vulnerable to attacks.

1.4 Analysis of the problem

The problem in this project is to develop a system which will help an admin in maintain patient and doctor details along with appointments. The system must be free-flowing UI along with fast data extraction.

1.5 Proposed solution strategy

In the proposed solution, an integrated system will be developed which will automate the entire patient doctor management process. To achieve this I will focus on the 3 main points below:

- **Data Collection:** For the data collection part, I first created a MySQL

database which will contain doctor, patient and admin details. The data will be generated through the application using the user registration functionality. This also act as a mock database for the implementation of the ManagementSystem. Appointment details will be populated in the database as and when an user books an appointment.

- **UI/UX design:** The UI is designed in JSP and served by the Spring MVC application running on a Tomcat Server providing the user and the admin a GUI to access the application and view/update information.
- **Implementation in Spring MVC:** The Spring MVC application contains the main business logic (Repository layer, Service Layer and Controller Layer) and all the middleware components linking the database (using JPA) and the UI (JSP). Authentication and authorization for the users and administrators are also handled, so that data security is maintained through password encryption.

1.6 SRS

A. Technologies used

- Front-end: HTML, CSS, JavaScript
- Middleware: Spring-web-MVC, Hibernate
- Back-end database: MySQL

B. Tools and Software required to develop

- Apache Tomcat 9.0
- Eclipse IDE / Spring tool suite 4
- Apache POI
- MySQL

C. Minimum Hardware Specifications

- Processor: Intel Pentium 4
- RAM: 4 GB
- OS: Windows XP
- Hard Disk: 5 GB

2. Organisation of the report

Chapter 1 Introduction discusses about the general overview of the problem, literature survey done, problem statement and the solution strategy proposed to overcome the prevailing problem.

Chapter 2 Design strategy for the solution discusses the design (architecture diagram, ER diagram) that is made that will tackle the problem mentioned above.

Chapter 3 Detailed test plan contains the test cases used to test the developed system.

Chapter 4 Implementation details discusses the pseudo code for the solution developed and also contains the flowchart for all the use cases.

Chapter 5 Results and discussions contains the screenshots of My final application and explains each screenshot.

Chapter 6 and beyond concludes My project report and mentions the achievements, the difficulties faced, the limitations of My project and the future scope of My work.

3.DESIGN STRATEGY

3.1 Architecture diagram

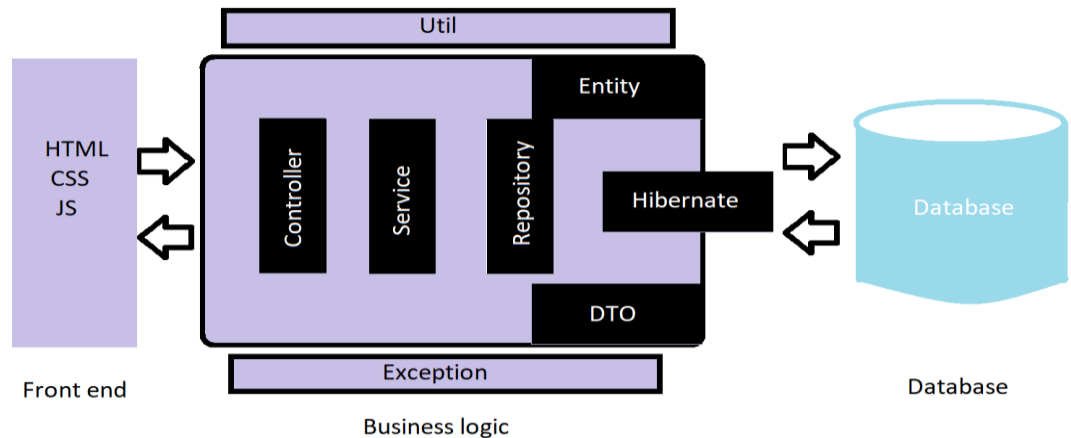


Figure 1: Architecture Diagram

- **Front end:**

- Front End is My presentation layer and it encapsulates all the UI logic within it.

- **Middle tier:**

- Middle tier consists of 3 layers: Controller, Service and Repository. It also contains all the entity, Util and custom exception classes within it, to handle exceptions, and date management issues.
- **Controller:** It is responsible for processing incoming http requests and returning the view to be rendered as a response.
- **Service:** This layer contains the business logic that drives the application's core functionalities.
- **Repository:** This layer is responsible for interacting with the database and hence encapsulates the ORM specific logic.
- **Entity:** Entities in JPA are POJOs representing data that can be persisted to the database. An entity represents a table stored in a database. Every instance of an entity represents a row in the table. The entities are patient, doctor, admin and appointment.

- **Hibernate:** Hibernate is a Java framework that simplifies the development of Java applications to interact with the database. It is an ORM (Object Relational Mapping) tool. Hibernate implements the specifications of JPA (Java Persistence API) for data persistence.

- **Backend:**

- The backend consists of the database which stores the data.

3.2 ER diagram

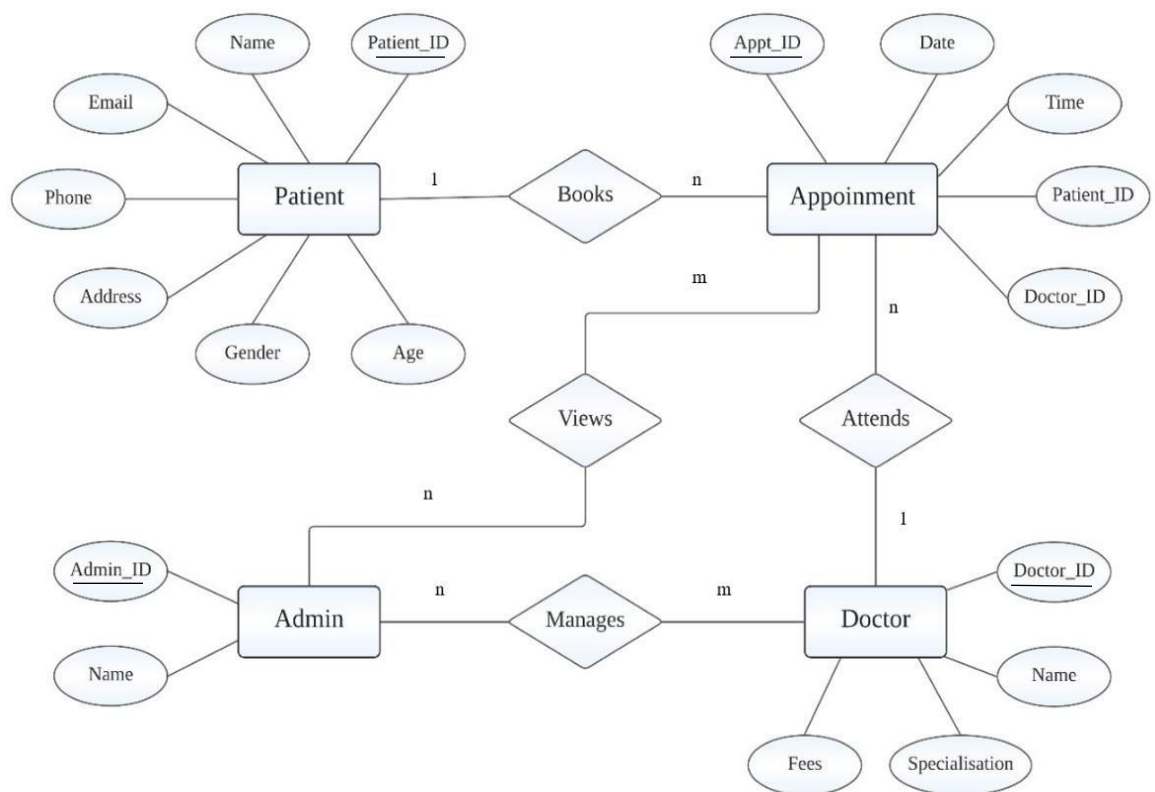


Figure 2: ER Diagram

4. IMPLEMENTATION DETAILS

4.1 Pseudo Code

- Patient Authentication: Auth system takes 2 parameters, a request(req) and a response(res).

Check if req==register

 redirect to create a new user(/register)

 take in user_details

 user.register(user_details)

 if res ==success, Redirect to /login

 else redirect to /register with msg “Failed to register”

Check if req==login

 redirect to login route(/login)

 take in user_details

 password.authenticate(user_details)

 if res==success, Redirect to /patDash

 else redirect to /register with msg “Failed to login”

- Patient Dashboard

Check if req==addappointment

 redirect to addappointment route(/addappointment)

 take in appointment_details

 user.addappointment(user_details)

 if res==success, Redirect to /patDash

 else redirect to /addappointment with msg “Failed to add appointment”

Check if req==patientprofile

 redirect to patientprofile route(/patientprofile)

 display user.patientprofile

Check if req==doctorlist

 redirect to doctorlist route(/doctorlist)

 display user.doctorlist

- Doctor Authentication: Auth system takes 2 parameters, a request(req) and a response(res).

Check if req==login

redirect to login route(/login)

take in user_details

password.authenticate(user_details)

if res==success, Redirect to /doctordashboard

else redirect to /register with msg “Failed to login”

- Doctor Dashboard: The doctor can see his/her patient appointments and profile

Check if req==doctorprofile

redirect to doctorprofile route(/doctorprofile)

user.(user_details)

if res ==success, Redirect to /login

else redirect to /register with msg “Failed to register”

- Admin Dashboard: The admin can add admins, doctors, enable, disable doctors as well as generate reports.

Check if req==addadmin

redirect to addadmin route(/addadmin)

take in admin_details

admin.addadmin(admin_details)

if res ==success, Redirect to /admindashboard

else redirect to /addadmin with msg “Failed to add admin”

Check if req==adddoctor

redirect to adddoctor route(/adddoctor)

take in doctor_details

admin.adddoctor(doctor_details)

if res ==success, Redirect to /admindashboard

else redirect to /adddoctor with msg “Failed to add doctor”

Check if req==enabledoctor

 redirect to enabledoctor route(/enabledoctor)

 take in doctor_details

 admin.enable(doctor_details)

 if res ==success, Redirect to /admindashboard

 else redirect to /addadmin with msg “Failed to enable doctor”

Check if req==disabledoctor

 redirect to disabledoctor route(/disabledoctor)

 take in doctor_details

 admin.disable(doctor_details)

 if res ==success, Redirect to /admindashboard

 else redirect to /addadmin with msg “Failed to disable doctor”

Check if req==report

 redirect to report route(/report)

 take in report_details

 admin.apply(report_details)

 if res ==success, generate report Redirect to /report

 else redirect to /report with msg “Failed to generate report”

4.2 Flowchart for all use cases

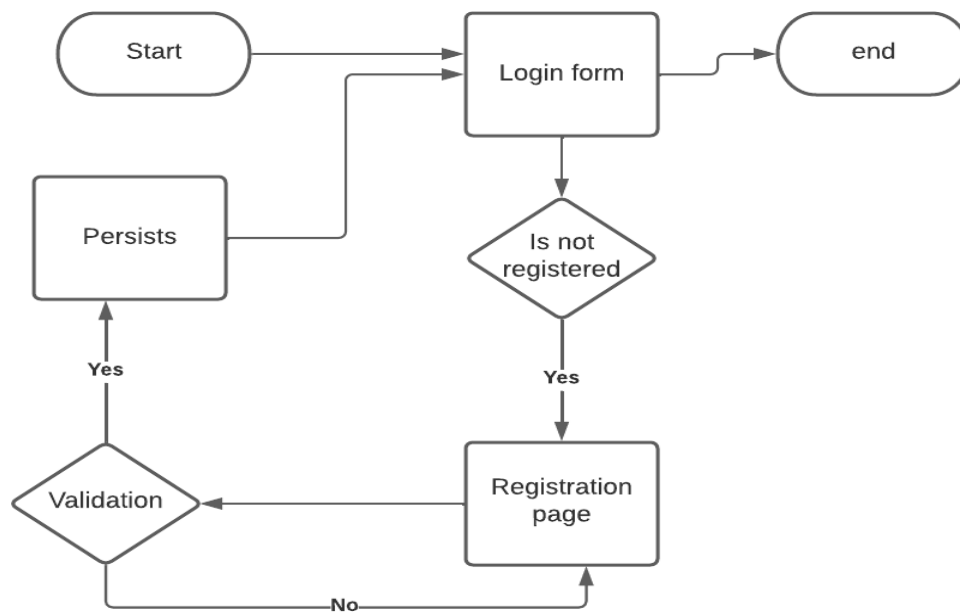


Figure 3: Patient authentication

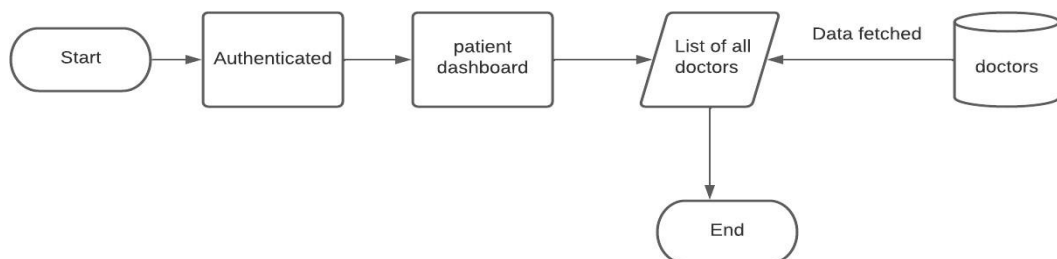


Figure 4. Patient seeing all doctor list

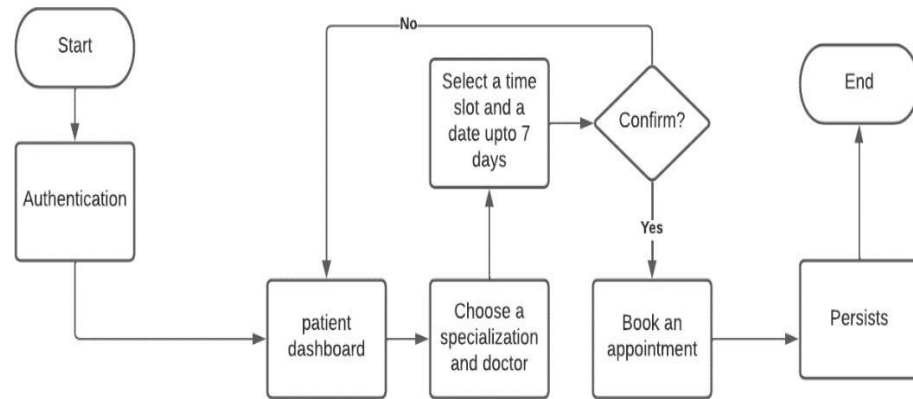


Figure 5. Patient booking appointments

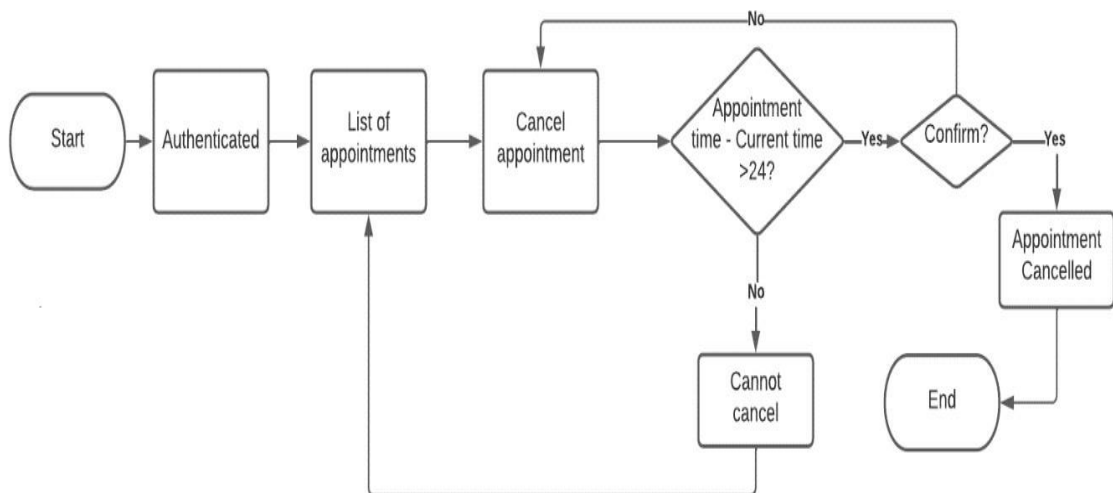


Figure 6. Patient cancelling appointments

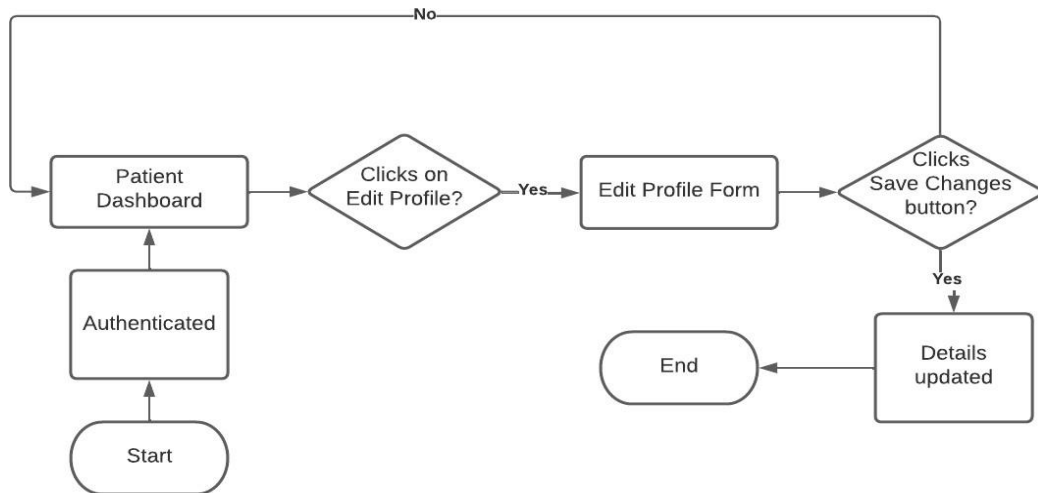


Figure 7: Patients updating their account

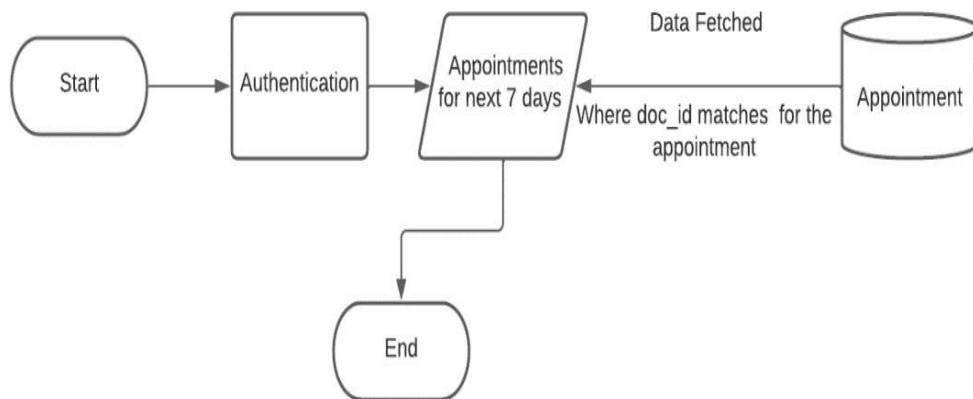


Figure 8: Doctors viewing their schedule

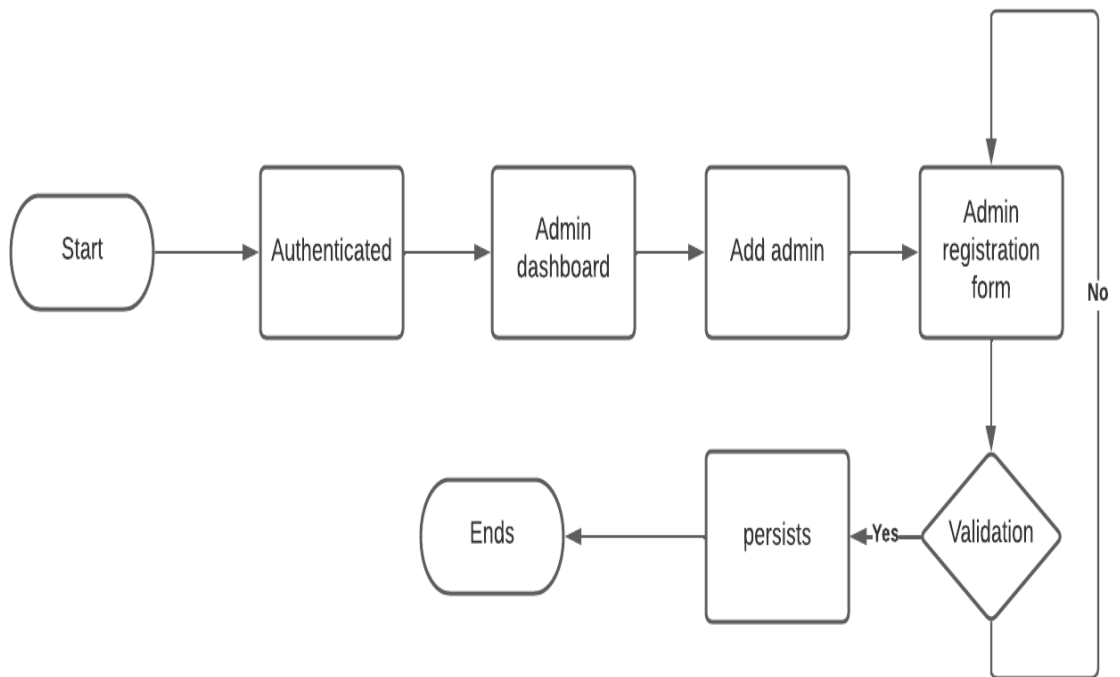


Figure 9: Admin adding admin

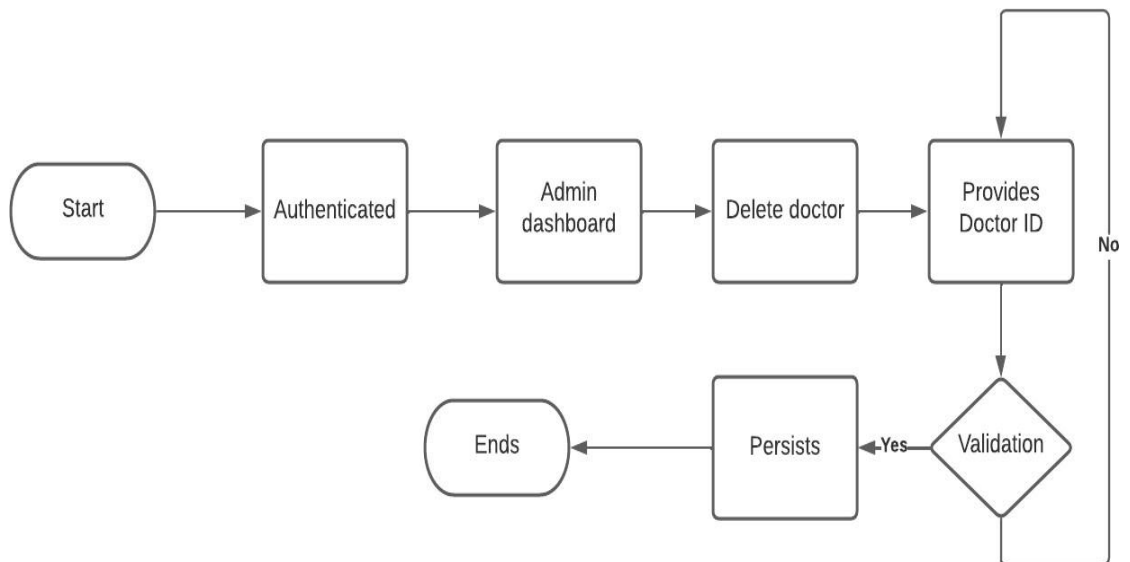


Figure 10: Admin deleting doctor

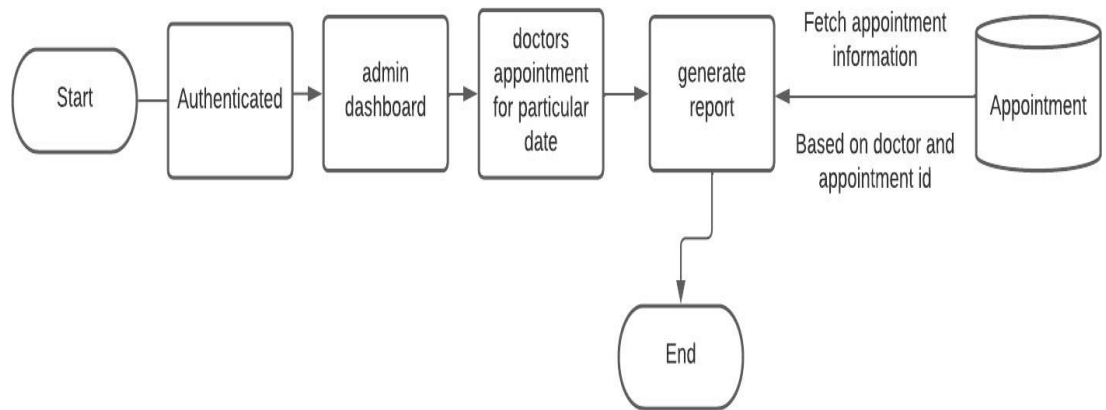


Figure 11: Admin generating reports

5.DETAILED TEST PLAN

The main plan for testing is to run test cases. I have prepared a few test cases below which covers all the functionalities.

Test Case ID	Description	Steps
MMS01	A patient can register himself/herself into the system	a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on register here c. Fill in the mandatory fields of name, email, phone number, password, age, gender. The address field is not mandatory d. Click on register
MMS02	Patient can log in	a. Open GUI using the link "a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Fill correct email and password in proper boxes. c. Click on login
MMS03	Patient can see the list of all doctors	a. Open GUI using the link "a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Fill correct email and password in proper boxes. c. Click on login d. In the dashboard, click on doctors
MMS04	Patient cannot book an appointment for a past date	a. Open GUI using the link "a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser c. Click on login d. In the dashboard, click on book appointment e. Select a specialization, and click on search f. Choose any doctor and click on book g. Provide a past date and select any time slot

MMS05	Patient cannot book an appointment on excess of 7 days	<p>a. Open GUI using the link "a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser</p> <p>c. Click on login</p> <p>d. In the dashboard, click on book appointment</p> <p>e. Select a specialization, and click on search</p> <p>f. Choose any doctor and click on book</p> <p>g. Provide a date in excess of 7 days and select any time slot</p>
MMS06	Patient can book an appointment for a doctor in the coming 7 days for a slot if the slot is open	<p>a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser</p> <p>c. Click on login</p> <p>d. In the dashboard, click on book appointment</p> <p>e. Select a specialization / or put a name pattern, and click on search</p> <p>f. Choose any doctor and click on book</p> <p>g. Provide a date inside of 7 days and select any time slot that is open</p> <p>h. Click on confirm</p>
MMS07	Patient cannot book an appointment for a doctor in the coming 7 days for a slot if the slot is already taken	<p>a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser</p> <p>c. Click on login</p> <p>d. In the dashboard, click on book appointment</p> <p>e. Select a specialization, and click on search</p> <p>f. Choose any doctor and click on book</p> <p>g. Provide a date inside of 7 days and select any time slot that is taken</p>
MMS08	A patient cannot cancel an appointment if less than 24 hours are left	<p>a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser</p> <p>c. Click on login</p> <p>d. In the dashboard, click on cancel for an appointment that is less than 24 hours away</p>
MMS09	A patient can cancel an appointment if less than 24 hos are left	<p>a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser</p> <p>c. Click on login</p> <p>d. In the dashboard, click on cancel for an appointment that is more than 24 hours away</p>

MMS10	A patient cannot book two appointments with two doctors in same date time slot	a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser c. Click on login d. In the dashboard, click on book appointment e. Select a specialization, and click on search f. Choose any doctor and click on book g. Provide a date inside of 7 days and select any time slot that is open h. Try to book another doctor for same date time slot who is available
MMS11	A date time slot appointment, cancelled by one patient, can be booked by another	a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser c. Click on login d. In the dashboard, click on cancel for an appointment that is more than 24 hours away e. Log in with another patient and try to book the doctor for the same date time slot
MMS12	Patient can edit his or her profile	a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser c. Click on login d. In the dashboard, click on edit profile e. Provide new email, phone number, address and password (if password is not to be changed, retype the password)
MMS13	Doctor can log in	a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on doctor login c. Fill correct email and password in proper boxes. d. Click on login
MMS14	A doctor can view his schedules for coming 7 days	a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on doctor login c. Fill correct email and password in proper boxes. d. Click on login e. Go to dashboard
MMS15	Admin can login to the system	a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login

MMS16	Admin can add doctors	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login e. Click on add doctor f. Provide name, email, specialization, fees and password, phone, gender and age of the doctor. Click on add doctor
MMS17	Admin can update doctor fees	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login e. Click on edit doctor, and click on edit for any doctor f. Provide new fees and save changes
MMS18	Admin can add admins	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login e. Click on add admin, provide name, password, gender, phone and own password. Click on add.
MMS19	Admins can generate reports on any doctor weekly or monthly	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login e. Click on generate report. Choose any doctor from the dropdown, choose whether a pdf or excel report is needed and choose whether a monthly or weekly report is needed. Click on generate report.
MMS20	An admin can enable a disabled doctor	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login e. On the dashboard, click on enable button for a doctor whose status is inactive

MMS21	Admin can disable a doctor who has no upcoming appointment	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login e. On the dashboard, click on enable button for a doctor whose status is active and has no appointment
MMS22	Admin cannot disable a doctor who has an upcoming appointment	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Fill correct ID and password in proper boxes. d. Click on login e. On the dashboard, click on enable button for a doctor whose status is active and also has an appointment
MMS23	Admin can change his / her own password	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. Click on change password d. provide old password correctly, and provide new password (in case of no change, retype the old password)
MMS24	Doctor can change his / her own password	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on doctor login c. Click on change password d. provide old password correctly, and provide new password (in case of no change, retype the old password)
MMS25	patient can provide rating for his last appointment	<ul style="list-style-type: none"> a. Open GUI using the link "http://localhost:8095/MedicalManagementSystemMMS/" from any browser b. Click on admin login c. A modal will come up, asking for review for the last appointment d. The patient can provide the ratings or skip it

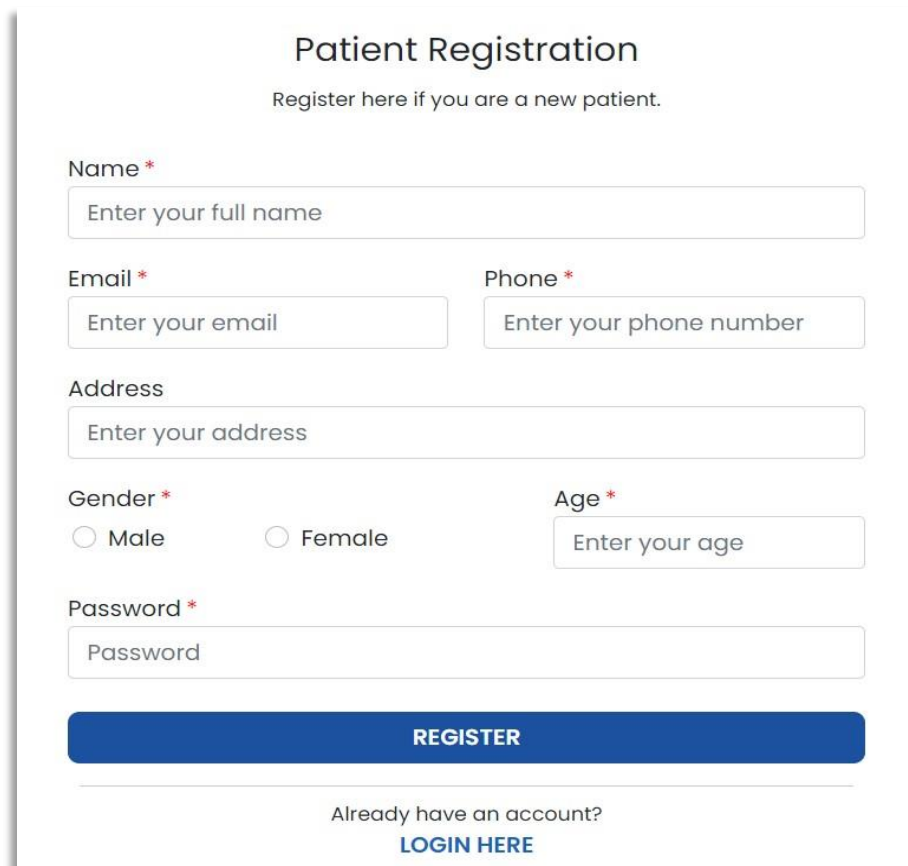
Table 2: Test Cases

6.RESULTS AND DISCUSSIONS



Figure 12: Home Page

The page above is the main page of My application which is displayed when I try to access the system. It contains links for the user to click on namely register as a patient or login as a patient or doctor.



The form is titled "Patient Registration" in a bold, black font. Below the title is a subtitle "Register here if you are a new patient." in a smaller, regular black font. The form contains several input fields: a single-line text box for "Name" with a red asterisk, a single-line text box for "Email" with a red asterisk, a single-line text box for "Phone" with a red asterisk, a single-line text box for "Address", a radio button for "Male" and a radio button for "Female" under the "Gender" label with a red asterisk, a single-line text box for "Age" with a red asterisk, and a single-line text box for "Password" with a red asterisk. A large blue button with the text "REGISTER" in white is positioned below the input fields. At the bottom, there is a link "LOGIN HERE" in blue text, preceded by the text "Already have an account?" in a smaller, regular black font.

Patient Registration

Register here if you are a new patient.

Name *

Enter your full name

Email *

Enter your email

Phone *

Enter your phone number

Address

Enter your address

Gender *

☐ Male ☐ Female

Age *

Enter your age

Password *

Password

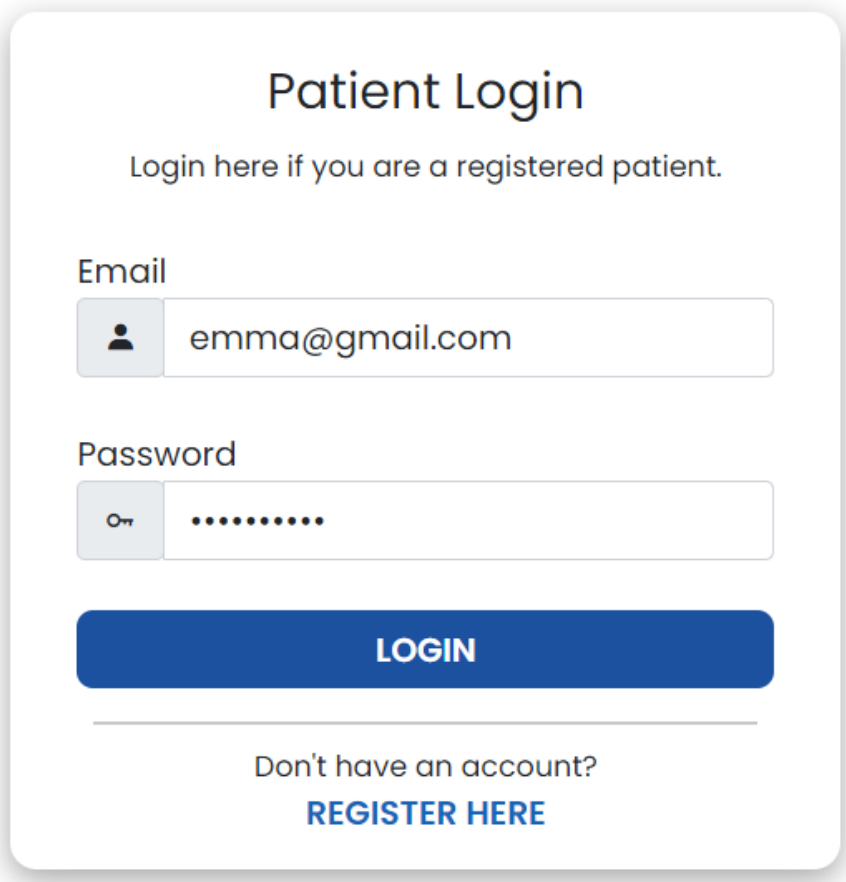
REGISTER

Already have an account?

[LOGIN HERE](#)

Figure 13: Patient Registration Form

This is the registration form displayed to the user when they click on the register button shown in the home page (Figure 12). It contains a few fields like Name, Email, Phone Number etc. of which fields like Name, Email, Phone Number, Gender, Age and Password are mandatory which is indicated by the “*” symbol.



The image shows a 'Patient Login' form. At the top, the title 'Patient Login' is centered in a large, dark font. Below it, a subtitle reads 'Login here if you are a registered patient.' in a smaller, lighter font. The form contains two input fields: 'Email' and 'Password'. The 'Email' field has a user icon on the left and the text 'emma@gmail.com' inside. The 'Password' field has an eye icon on the left and a series of dots representing a masked password. Below these fields is a prominent blue button with the word 'LOGIN' in white capital letters. Underneath the button, there is a horizontal line, followed by the text 'Don't have an account?' and a blue link that says 'REGISTER HERE'.

Patient Login

Login here if you are a registered patient.

Email

emma@gmail.com

Password

.....

LOGIN

Don't have an account?

[REGISTER HERE](#)

Figure 14: Patient Login Form

This is the registration form which is displayed when the user clicks on the login button on the home page (Figure 12). It takes in Email and Password, sends it for verification and then lets the user to access their respective dashboard. If the patient is not registered then they are also provided with a **REGISTER HERE** button which will take them back to the Patient Registration Form (Figure 13).

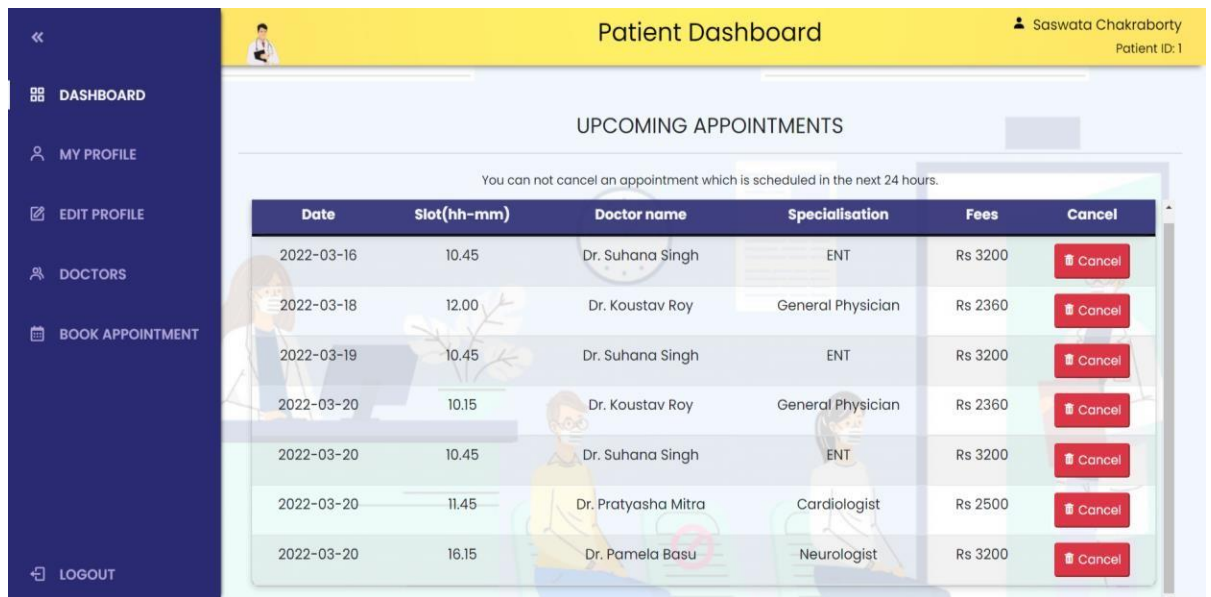


Figure 15. Patient Dashboard

This is the page that is displayed when the patient is verified and successfully logs in. Here there is a table in the centre that shows the appointments that the patient has in the upcoming days. There are CANCEL buttons placed beside each appointment for the patient to use in case they want to cancel their appointment. There are also links to various other functionalities like MY PROFILE tab for the patient to check their account details, BOOK APPOINTMENT tab for appointment booking etc.

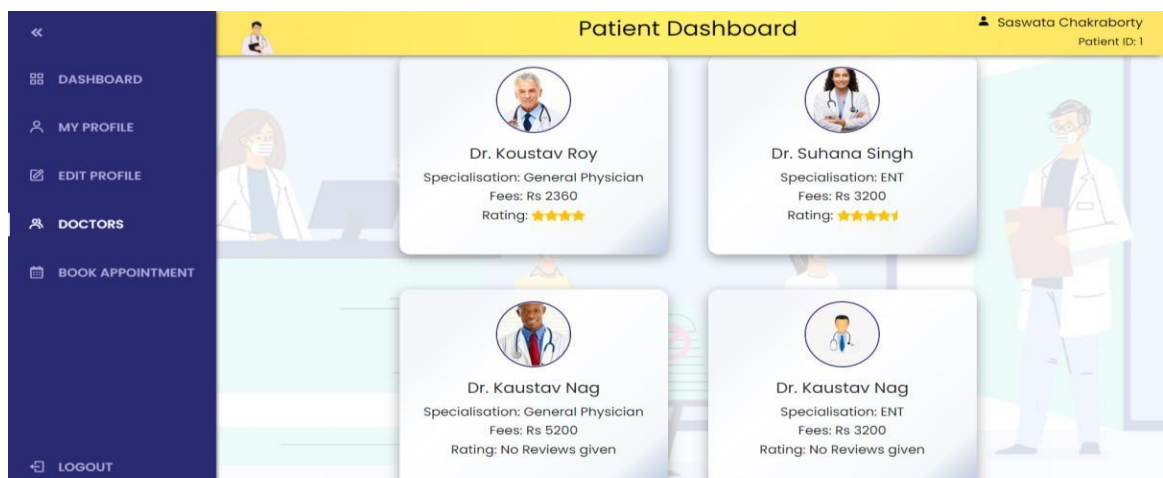


Figure 16: Doctor List

This is a list of all doctors present in the database which is displayed when the patient clicks on the DOCTORS tab in the side panel of their dashboard.

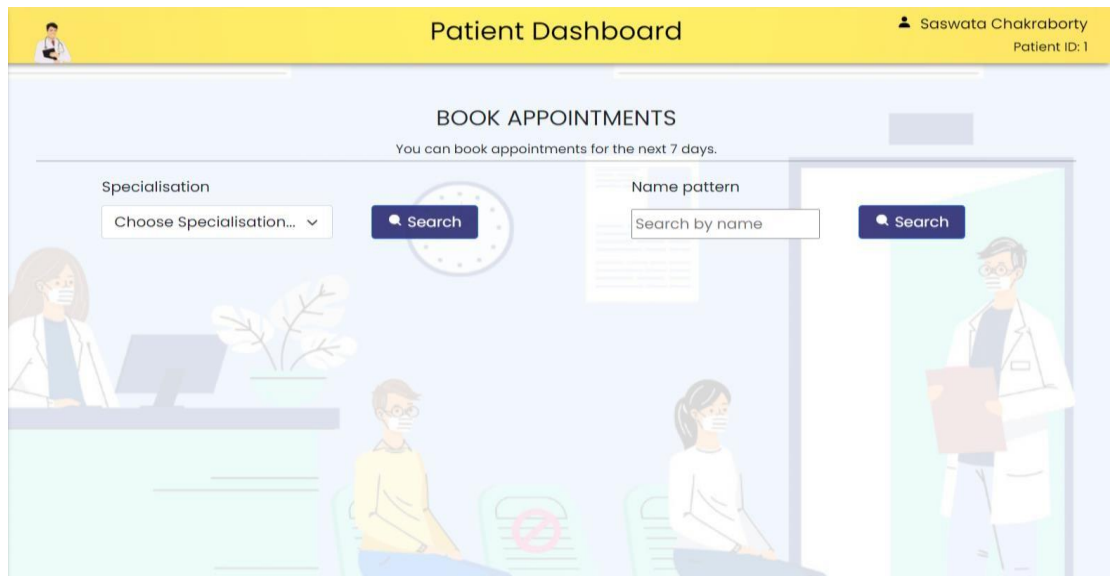


Figure 17: Appointment booking

Upon clicking on the BOOK APPOINTMENT button on the side panel, the patient is redirected to a page where they can select the appointment date. Once the date is selected they are taken to a page (Figure 17) where they can filter doctors present in the system based on their specialisation or their name. Upon click search they will be shown all the doctors satisfying that filtering criteria along with a BOOK button.

Appointment Details

Singh
General Physician

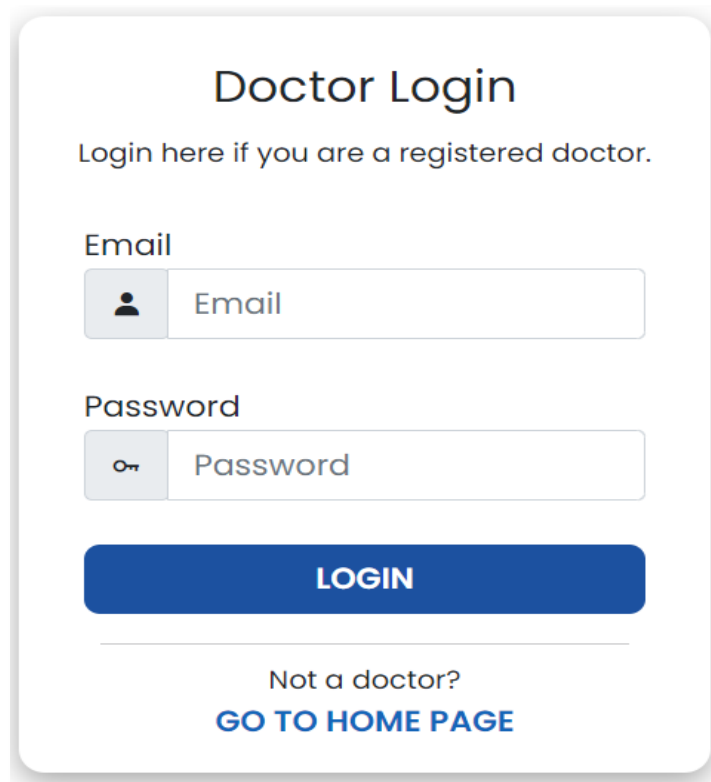
Slots Available for 2022-03-16

10.00	10.15	10.30	10.45	11.00	11.15
11.30	11.45	12.00	12.15	12.30	12.45
14.00	14.15	14.30	14.45	15.00	15.15
15.30	15.45	16.00	16.15	16.30	16.45

Confirm

Figure 18: Appointment Slots

After the patient clicks on the BOOK button of a particular doctor they are shown this modal which contains all the slots in the selected date. **RED** means the slot is either booked by some other patient or the patient trying to book the appointment already has an appointment on that date and time slot. **GREEN** bordered slots mean the slots are available for booking. Upon clicking the CONFIRM button, the user is redirected to the dashboard and the appointment details (date, time slot and patient ID) are stored in the database. The appointment is then visible on the appointments table (Figure 15).



The image shows a 'Doctor Login' form. At the top, it says 'Doctor Login' in a large, dark blue font. Below that, a subtitle reads 'Login here if you are a registered doctor.' There are two input fields: 'Email' with a person icon and 'Password' with a key icon. A blue 'LOGIN' button is positioned below the fields. At the bottom, there is a link that says 'Not a doctor? GO TO HOME PAGE'.

Doctor Login

Login here if you are a registered doctor.

Email

Password

LOGIN

Not a doctor?
[GO TO HOME PAGE](#)

Figure 19: Doctor Login Form

On the home page (Figure 12), after click on the LOGIN button there is a button which redirects user to the doctor login form. Here, the user has to enter their registered Email and their Password. Once the credentials are validated, they are taken to the doctor dashboard.



The image shows a 'Doctor Dashboard' for Dr. Koustav Roy (Doctor ID: 1). The dashboard has a yellow header with the doctor's name and ID. A dark blue sidebar on the left contains navigation links: DASHBOARD, MY PROFILE, CHANGE PASSWORD, and LOGOUT. The main content area is titled 'UPCOMING APPOINTMENTS' and features a table with columns for Date, Slot, and Patient Name. The table lists five appointments. The background of the dashboard includes illustrations of a doctor at a desk, a patient sitting, and a doctor walking.

Doctor Dashboard

Dr. Koustav Roy
Doctor ID: 1

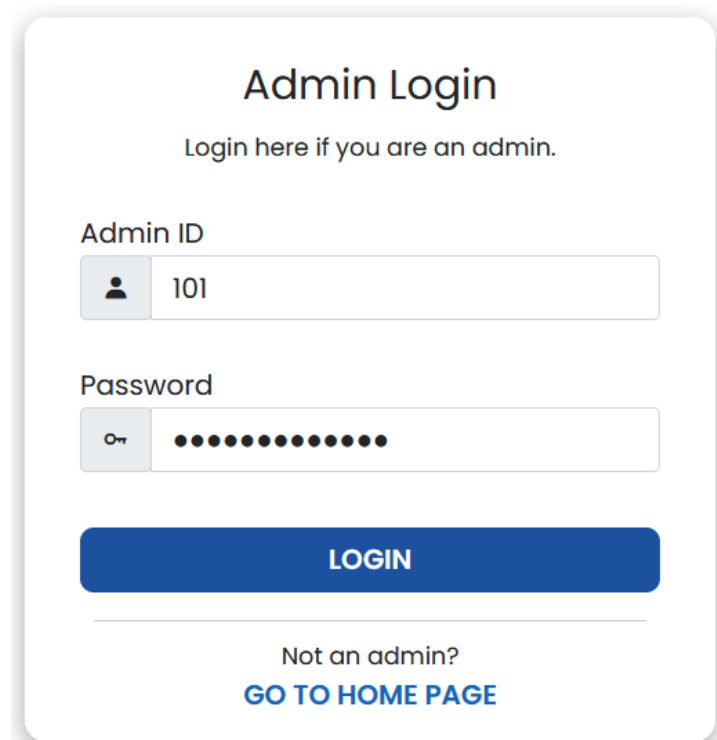
UPCOMING APPOINTMENTS

Date	Slot	Patient Name
2022-03-16	12.00	Shovan Deb Sharma
2022-03-17	10.30	Shovan Deb Sharma
2022-03-18	12.00	Saswata Chakraborty
2022-03-18	12.15	Sampa Chakraborty
2022-03-20	10.15	Saswata Chakraborty

LOGOUT

Figure 20: Doctor Dashboard

Upon successful validation, the doctors are taken to the dashboard where they are shown their upcoming appointments. Also, on the right panel there are buttons like MY PROFILE and CHANGE PASSWORD which give the doctors the functionality to view their account details and change their password.

The image shows a web form titled "Admin Login". Below the title is a subtitle "Login here if you are an admin.". There are two input fields: "Admin ID" with a user icon and the value "101", and "Password" with a toggle icon and masked characters. A blue "LOGIN" button is below the inputs. At the bottom, there is a link "GO TO HOME PAGE" preceded by the text "Not an admin?".

Admin Login

Login here if you are an admin.

Admin ID

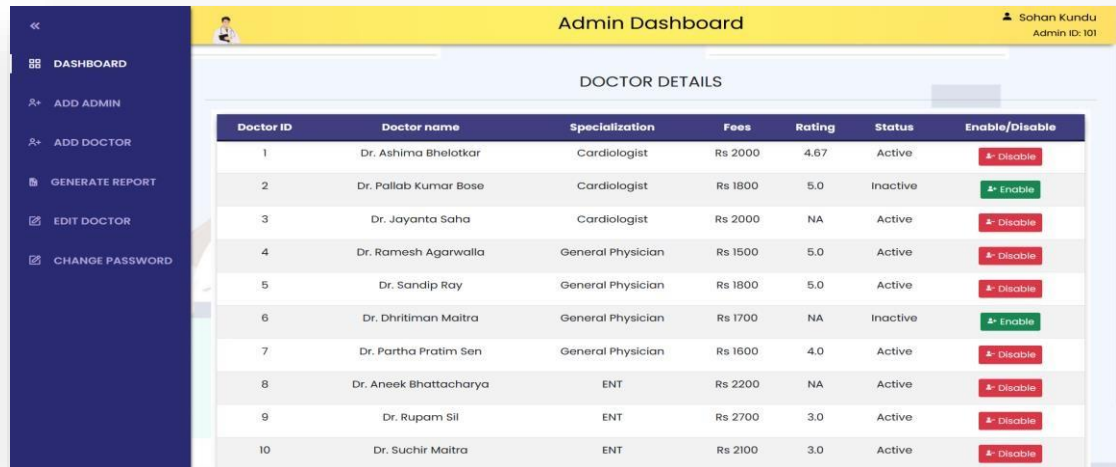
Password

LOGIN

Not an admin?
[GO TO HOME PAGE](#)

Figure 21: Admin Login Form

Once again, on the home page (Figure 12) once the user presses on the LOGIN button they have the ability to click on button that says ADMIN LOGIN. Upon click on that button they are presented this form which takes Admin ID and Password as inputs, verifies them and then lets the user into the next page.

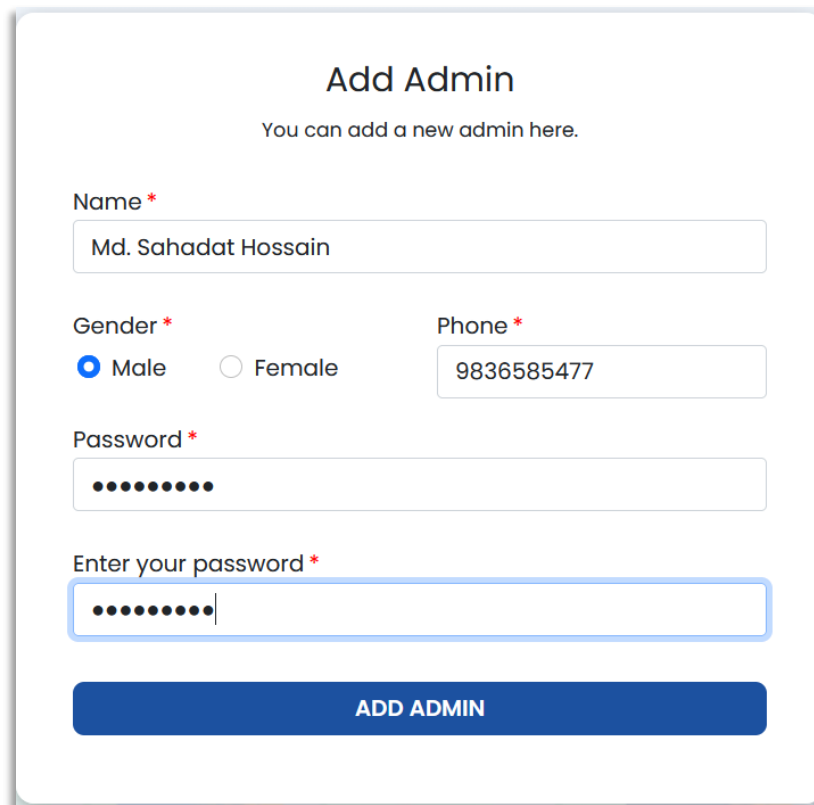


The screenshot shows the Admin Dashboard interface. On the left is a dark blue sidebar with navigation links: DASHBOARD, ADD ADMIN, ADD DOCTOR, GENERATE REPORT, EDIT DOCTOR, and CHANGE PASSWORD. The main content area has a yellow header with 'Admin Dashboard' and user info 'Sohan Kundu Admin ID: 101'. Below the header is a 'DOCTOR DETAILS' section containing a table of 10 doctors. Each row in the table includes columns for Doctor ID, Doctor name, Specialization, Fees, Rating, Status, and an 'Enable/Disable' button. The buttons are green with a right arrow for 'Enable' and red with a left arrow for 'Disable', depending on the doctor's status.

Doctor ID	Doctor name	Specialization	Fees	Rating	Status	Enable/Disable
1	Dr. Ashima Bhelotkar	Cardiologist	Rs 2000	4.67	Active	Disable
2	Dr. Pallab Kumar Bose	Cardiologist	Rs 1800	5.0	Inactive	Enable
3	Dr. Jayanta Saha	Cardiologist	Rs 2000	NA	Active	Disable
4	Dr. Ramesh Agarwalla	General Physician	Rs 1500	5.0	Active	Disable
5	Dr. Sandip Ray	General Physician	Rs 1800	5.0	Active	Disable
6	Dr. Dhritiman Maitra	General Physician	Rs 1700	NA	Inactive	Enable
7	Dr. Partha Pratim Sen	General Physician	Rs 1600	4.0	Active	Disable
8	Dr. Aneek Bhattacharya	ENT	Rs 2200	NA	Active	Disable
9	Dr. Rupam Sil	ENT	Rs 2700	3.0	Active	Disable
10	Dr. Suchir Maitra	ENT	Rs 2100	3.0	Active	Disable

Figure 22: Admin Dashboard

Upon successful validation, the admin is redirected to the admin dashboard which shows the doctors present in the system in the form of a table. In this table, along each doctor, there is an ENABLE/DISABLE button. If the doctor is enabled, then the DISABLE button will be visible and if the doctor is disabled then the ENABLE button will be visible; If a doctor is having upcoming appointments, he/she cannot be disabled. On the left panel, there are a few buttons like ADD ADMIN, ADD DOCTOR, GENERATE REPORT etc. for the admin to use.



The image shows a web form titled "Add Admin" with a subtitle "You can add a new admin here." The form contains several input fields: a text field for "Name" with the value "Md. Sahadat Hossain", a "Gender" section with radio buttons for "Male" (selected) and "Female", a text field for "Phone" with the value "9836585477", a "Password" field with masked characters, and an "Enter your password" field also with masked characters. At the bottom is a blue button labeled "ADD ADMIN". All fields are marked as mandatory with an asterisk.

Add Admin

You can add a new admin here.

Name *
Md. Sahadat Hossain

Gender *
☒ Male ☐ Female

Phone *
9836585477

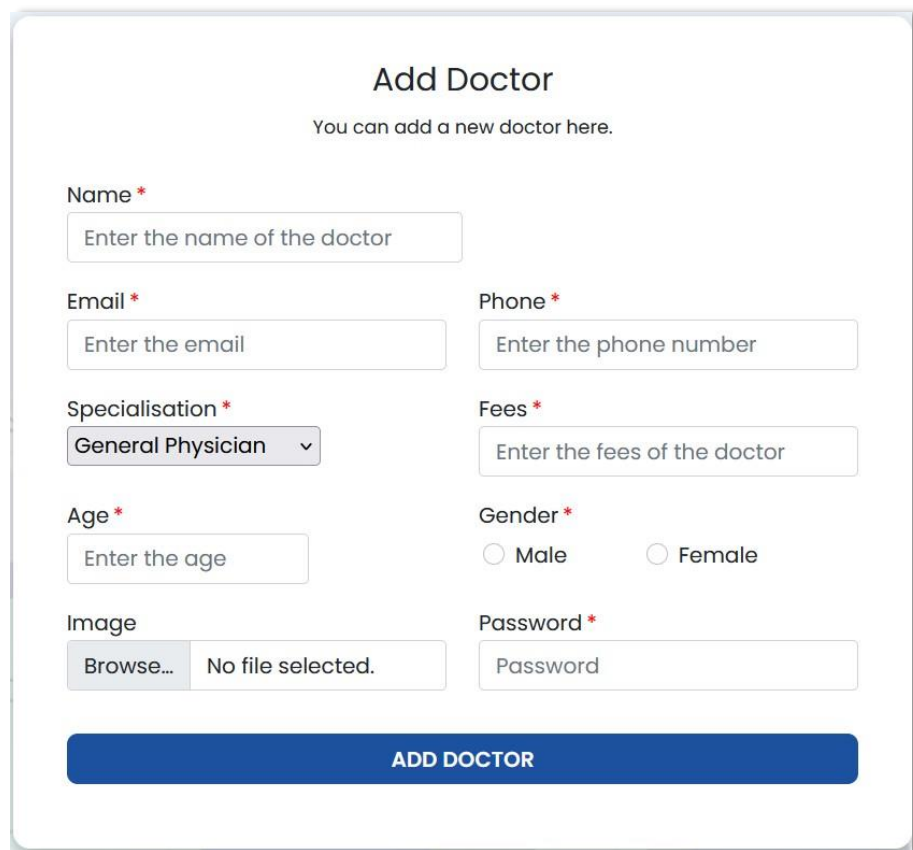
Password *
.....

Enter your password *
.....|

ADD ADMIN

Figure 23: Add Admin Form

Upon clicking the ADD ADMIN button on the admin dashboard (Figure 22), the admin is redirected to this form. This form takes a few fields as input such as Name, Gender, Phone Number etc. Each field in this form is mandatory which is indicated by the “*” symbol. Once they enter the details and click on the ADD ADMIN button a new admin with the enter details is added into the system.



The 'Add Doctor' form is a web interface for adding a new doctor. It features a title 'Add Doctor' and a subtitle 'You can add a new doctor here.' The form contains several input fields: 'Name' (text input), 'Email' (text input), 'Phone' (text input), 'Specialisation' (dropdown menu with 'General Physician' selected), 'Fees' (text input), 'Age' (text input), 'Gender' (radio buttons for 'Male' and 'Female'), 'Image' (file upload area with 'Browse...' button and 'No file selected.' text), and 'Password' (text input). A large blue button labeled 'ADD DOCTOR' is at the bottom.

Figure 24: Add Doctor Form

Similar to the Add Doctor Form (Figure 23), when the admin clicks on the ADD DOCTOR button on the dashboard they are redirected to this form. This form takes a few mandatory fields as inputs such as Name, Email, Phone Number, Specialisation etc. Once all the details are filled and the ADD DOCTOR button is clicked, a new doctor with these details is added into the system.



The 'Generate Report' form is a web interface for generating reports. It features a title 'GENERATE REPORT' and a subtitle 'You can generate the weekly report or monthly of a particular doctor in Excel or PDF format.' The form contains three dropdown menus: 'Choose Doctor' (with '1 - Dr. Ashima Bhelotkar - Cardiologist' selected), 'Report Interval' (with 'Weekly' selected), and 'Report Type' (with 'Excel' selected). A blue button labeled 'Download' is at the bottom right.

Figure 25: Generate Report

On the admin dashboard there is a button on the left panel that says **GENERATE REPORT** (Figure 22). Upon clicking that button, this page is displayed which gives the admin the option to generate report for a particular doctor. The report generated will either be weekly or monthly which can be selected using a dropdown list under **Report Interval**. The report can be generated in two formats, excel and pdf. Once all three fields are selected, the admin can press the **DOWNLOAD** button on the side and the report will be downloaded.

7.SUMMARY AND CONCLUSION

7.1 Summary of achievements

The proposed solution for the problem is successfully developed. The desired features of the application like management of user (patient and doctor) details, successful appointment management without any redundancy in slots, report generation etc. are implemented successfully.

7.2 Main difficulties encountered and how they are tackled

The primary difficulty that I faced was developing a system that takes care of data redundancy and time slot clashes. I solved these by putting validations all across the system. This means that every detail and value is validated to see if it is recurring in the database before storing it.

The second problem I faced was to make the application as user friendly as possible. This was done after implementing multiple changes to My frontend and testing it thoroughly. All these tests made sure that the system is refined and that the system will suit majority of the users.

Time management was a major obstacle faced during the development phase. It took a lot of focus to make sure that I complete My solution on time and make it user friendly.

7.3 Limitations of the project

The current system focuses mainly on storing user details and managing appointments. Due to time constraint there are a few limitations in the project.

In the database, the initial admin i.e. the first admin needs to be added manually by going into the database.

The application was made on large screens i.e. laptops and desktops so majority of My attention was making sure it works on those. Even though the application works on small screens like mobile phones, it isn't very user friendly.

The doctor has been given very little functions in the system namely the ability to view their appointments and change their password.

7.4 Future scope of the project

The project has a lot of scope which can be achieved with continued work and more time.

The system might handle monetary transactions in the future which means along with appointments and user details, it will also store their fees and calculate payment for each doctor. The payments might be made within the system, as well as the refunds for cancellation.

The current system only stores and manages user details. The system can be enhanced to open medical shops, from where medicines can be ordered.

The doctors will be able to view a patients' past appointments, history and make better treatments.

Also the system could support the sharing of appointment details with the users in the form of SMS and email.

Finally, as mentioned above, due to time constraint the application is not very user friendly for small screen devices. With more work on the frontend, it can be refined. This will allow the application to work on both large screen devices (laptops and desktop) and small screen devices (mobile phones) in a much better and user-friendly manner.

8.REFERENCES

- [1] Gadhari, Digvijay H., Yadnyesh P. Kadam, and Parineeta Suman. "Hospital Management System." *International Journal for Research in Engineering Application & Management (IJREAM)*, ISSN (2015): 2494-9150.
- [2] Dridi, Ahmed, Anis Tissaoui, and Salma Sassi. "The medical project management(MPM) system." In *2015 Global Summit on Computer & Information Technology (GSCIT)*, pp. 1-6. IEEE, 2015.
- [3] Kumaran, Kalai Selvi, Deepak, "A Study of Advanced Hospital Management "in *IOSR-JDMS*, 2020.