

A
PROJECT REPORT
ON
BOOK YOUR DOCTOR

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE ACADEMIC
REQUIREMENT FOR THE AWARD OF DEGREE OF
MASTER OF COMPUTER APPLICATION
COMPUTER SCIENCE AND APPLICATION

Submitted
By
AMIYA RANJAN ROUT (1805106002)
PRAGYA PARIMITA PANDA (1805106018)
MANAS RANJAN PARIDA (1805106014)

Under the Esteemed guidance of
Dr. JIBITESH MISHRA

Head of Department

Department of CSA



DEPARTMENT OF COMPUTER SCIENCE & APPLICATION
COLLEGE OF ENGINEERING AND TECHNOLOGY
(Techno Campus ,PO-Ghatikia, Mahalaxmi Vihar,Bhubaneswar-751029)

2019-2020

COLLEGE OF ENGINEERING AND TECHNOLOGY

(Techno Campus, PO- Ghatikia, Mahalaxmi Vihar,Bhubaneswar-751029)

DEPARTMENT OF COMPUTER SCIENCE & APPLICATION



CERTIFICATE

This is to certify that the Thesis/Dissertation entitled **BOOK YOR DOCTOR** are being submitted by **AMIYA RANJAN ROUT, PRAGYA PARIMITA PANDA, MANAS RANJAN PARIDA** with regd.no **1805106002,1805106018,1805106014** respectively in partial fulfilment of the requirement for the award of the degree of MCA. In to the College of Engineering & Technology is a record of bonafide work carried out by us under our guidance and supervision from July to December 2019.

The results presented in this thesis have been verified and are found to be satisfactory. The results embodied in this thesis have not been submitted to any other University for the award of any other degree.

INTERNAL GUIDE

Dr. Jibitesh Mishra

(Associate Professor)

HOD

Dr. Jibitesh Mishra

(Associate Professor)

COLLEGE OF ENGINEERING & TECHNOLOGY

(Techno Campus, PO- Ghatikia, Mahalaxmi Vihar,Bhubaneswar-751029)

DEPARTMENT OF COMPUTER SCIENCE & APPLICATION



ACKNOWLEDGMENT

We are extremely grateful to **Professor Dr.Jibitesh Mishra, H.O.D** and as our Internal Guide, Department of CSA, College of Engineering & Technology for their encouragement and valuable guidance in bringing shape to this dissertation.

We will be failing in duty if we do not acknowledge with grateful thanks to the author of the references and other literature referred in this Project.

We express our thanks to all staff members and friends for all the help and coordination extended in bringing out this Project successfully in time.

Finally, we are very much thankful to our parents who guided us for every step.

Date: **Amiya Ranjan Rout**
(1805106002)

Place: **Pragya Parimita Panda**
(1805106018)
Manas Ranjan Parida
(1805106014)

COLLEGE OF ENGINEERING & TECHNOLOGY
(Techno Campus, PO- Ghatikia, Mahalaxmi Vihar,Bhubaneswar-751029)

DEPARTMENT OF COMPUTER SCIENCE & APPLICATION



DECLARATION

We **Amiya Ranjan Rout, Pragya Parimita Panda, Manas Ranjan Parida** bearing Roll No: **1805106002, 1805106018 , 1805106014** respectively a Bonafede student of **College of Engineering and Technology**, would like to declare that the project titled "**BOOK YOUR DOCTOR**" A partial fulfilment of MCA Degree course of College of Engineering & Technology is my original work in the year 2019 under the guidance of **Dr. Jibitesh Mishra**, Associate Professor, Computer Science & Application Department and it has not previously formed the basis for any degree or diploma or other any similar title submitted to any university.

Date:

Student Name

ABSTRACT

Now a day's healthcare issue is one of the tough challenges for our country. For metro cities, smart cities it may not be a major issue but for small cities and especially for rural areas this issue is enhancing day by day because of lack of experienced doctors. So, our main focus is to develop this project to reduce health care issues in those areas. We aim to develop an application that would act like a medium between Doctor and Patient. As we know some experienced doctor from bigger cities used to visit their private clinic or some pharmacies situated in small cities and rural area once or more than once in a week. But people in their respective areas are unable to know about Doctor's availability. And even if they get to know the Doctor's availability, they are facing a lot of problems to get a ticket. For e.g. they have to stand in queue from dawn to get an early ticket. Through our platform, our main target is to collaborate with those pharmacies and provides some better solution for this issue. This application will provide availability of registered Doctors. And Patients can also search doctors as per their specialty and book an appointment for the same. And if later on they changed their mind they can also cancel the appointment.

INDEX

Chapter 1: Introduction-----	1
1.1 Introduction-----	1
1.2 Project Scope-----	1
1.3 Existing System-----	2
1.4 Proposed System-----	2
Chapter 2: Literature Survey/Review of Literature-----	3
2.1 Introduction-----	4
2.2 Related Work-----	6
Chapter 3: System Requirement-----	7
3.1. Software Requirement-----	8
3.2. Hardware Requirement-----	9
Chapter 4: Software Requirement Analysis-----	10
4.1. System planning and the initial Investigation-----	11
4.2. Information Gathering-----	12
4.3. Feasibility Study-----	13
4.3.1. Technical Feasibility-----	13
4.3.2. Economic Feasibility-----	13
4.3.3. Cost Benefit Analysis-----	14
4.3.4. Legal Feasibility-----	14
Chapter 5: Project Plan & Approach-----	15
Chapter 6: Project Schedule-----	17

Chapter 7: Software Design-----	19
7.1. Scope-----	20
7.2. System model specification-----	22
7.3. Class Diagram-----	23
7.4. Use Case Diagram-----	24
7.5. ER Diagram-----	25
7.6. System Architecture-----	26
Chapter 8: Use Case Description-----	27
Chapter 9: Coding /Code Templates-----	37
Chapter 10: Testing-----	56
Chapter 11: Database Design View-----	64
Chapter 12: Output Screens-----	71
Chapter 13: Conclusion-----	80
Chapter 14: References/Bibliography-----	84

Chapter 1. INTRODUCTION

1.1 INRODUCTION

BookUrDoctor is more than an app for finding doctors. Whether it's viewing doctors' profile, booking doctor appointments, confirming appointments, cancelling appointments, **BookUrDoctor** helps you find online medical services & solutions to enable you to take better care of yourself. Briefly, the main goal of this project is: Finding the right doctors for your good health can be a challenge. With an expansive list of verified doctors, specialists, surgeons from neighborhood clinics & top hospitals to choose from, we help you get appointments with the doctors using the **BookUrDoctor** app.

1.2. Project Scope

- The system allows a doctor to add their details such as their name along with their designation, specialization, experiences and also with a brief description of them.
- User can search for top experienced doctors as per their specialization.
- User can search for top experienced doctors as per their nearby location.
- User can find right doctors by viewing the doctor's profile.
- User can book an appointment with their specialized doctors.
- User can see their all appointment letters at a one place.
- User can also cancel their appointments.

1.3. EXISTING SYSTEM

In the existing system it only working on metro cities and smart cities. Then what about for small town and especially for rural area? And also, it's main focus on doctor's private clinic which is very rare in rural areas.

The following are the drawbacks present in the existing system:

1. Only for smart cities and for metro cities.
2. No scope for small towns.
3. No scope for rural areas.
4. Only focus on doctor's private clinic.

1.4. Proposed solution

In the proposed system our main focus is to collaborate with those pharmacies in which some experienced doctors from bigger cities used to visit once or more than once in a week which are situated in small cities and rural areas. Our proposed system provides solution to existing system by extending its facilities as follows:

1. Extending the facility to the small-town areas.
2. Extending the facility to the rural areas as well.
3. Collaborating with pharmacies.

Chapter 2: Literature Survey/Review of Literature

2.1. Introduction

A literature review is a search and evaluation of the available literature in your given subject or chosen topic area. A lot of projects and researches has been done aiming in proper solving the problem of users.

2.2. Related Work

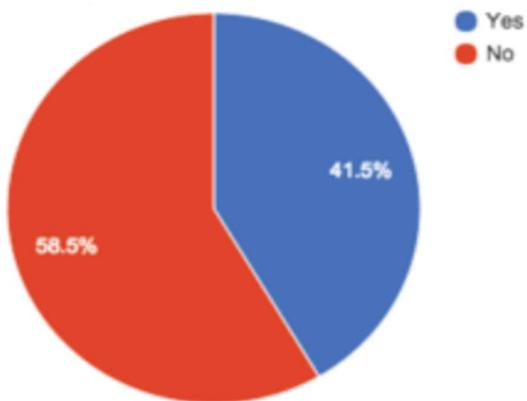
Here is some related work of the book your doctor system

2.2.1. Doctor Booking

In this project user must register an account and book a doctor in order to fix aqn appointment with the doctor. The system allow the user to register and as soon as they register an account with unique phone number, id is created in the system. The user must require an Android application or web platform to book a doctor by mobile or desktop. The status of the booking is updated only by respective authority. The user has to wait till the appointment letter has not been displayed. Bookings can be easily done by the user and can be tracked with the unique phone number provided at the time of registration. The user can then view the details of the booking. This system will not only save the time of the booking fillers but also lead to track their booking and its status in a regular interval of time. Once the date of the booking is done, the user will have the appointment letter on his smartphone and email address as well.

2.2.1. Online appointment Booking survey Reporting

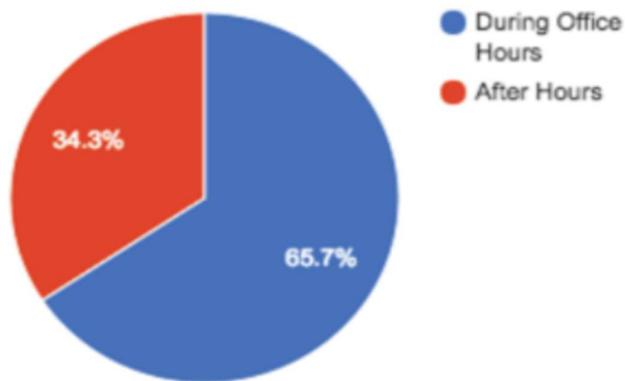
Nearly half (42 percent) of survey respondents say they would like online appointment scheduling on the healthcare website, versus scheduling by calling the practice. However, only 17 percent of respondents said they currently have the option to schedule their appointment online.



PatientPop Survey: I would prefer to book an appointment with the doctor online with a website or mobile app without having to go to the clinic.

Given the chance to schedule online 24/7, about 34 percent of new patients were able to make an appointment outside of standard business hours.

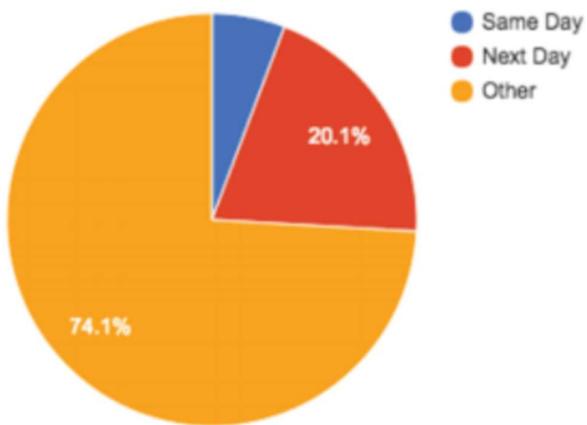
PatientPop Data: 34.4% of appointment scheduled online were scheduled during closing hours.



When patients have access to schedule their appointment online, 26 percent are able to choose appointments the same day or the next day, filling empty slots.

PatientPop Data: Online scheduling allows patient to fill open same day and next day slots.

- 6% of appointment scheduled online are for the same day slots
- 20% are of next day slots
- 26% of appointment scheduled online were for same day or next day slots



Chapter 3. System Requirement

3.1. Software Requirement

For Android Platform

3.1.1. Frontend

3.1.1.1. Android Development Tools

3.1.1.2. Advance Java

3.1.1.3. XML

3.1.2. Backend

3.1.2.1. Firebase

3.1.3. IDE

 3.1.3.1. Android Studio

3.2. Hardware Requirement

3.2.1. Application Requirement

3.2.1.1. Android version 6.0 (Marshmallow onwards)

3.2.1.2. 4GB RAM

 3.2.1.3. 1.2 GHZ Processor

3.2.2. Development Environment Requirement

 3.2.2.1. Intel i3 5th Generation Processor

 3.2.2.2. Windows 10 / UBUNTU 14.0.4

 3.2.2.3. Android Studio Installed

For Web Platform

3.1.1. Frontend

3.1.1.1. Web Development Tools

3.1.1.2. HTML, CSS

3.1.1.3. BOOTSTRAP

3.1.2. Backend

3.1.2.1. MySQL

3.1.3. IDE

3.1.3.1. Brackets, Sublime

3.2.1. Development Environment Requirement

3.2.1.1. Intel i3 5th Generation Processor

3.2.1.2. Windows 10 / UBUNTU 14.0.4

3.2.1.3. Brackets or Sublime Installed

Chapter 4: Software Requirement Analysis

4. Software Requirement Analysis

Analysis is a detailed study of the various operations performed by a system and their relation within and outside the system. A key question is what must be done to solve the problem. One aspect of the analysis defining the boundaries of the system and determining whether or not a candidate system should consider other related system. During analysis data are collected on the available files decision points and transaction handled by the parent system. Some logical system models and tools are used in the analysis. Data flow diagrams, interviews, onsite observation and questionnaires are examples, the interview is commonly used in analysis. It requires special skill and sensitivity to the subject being interviewed bias in data collection and interpretation can be a problem.

Training experience and common sense are required for collections of the information needed to do analysis. Once analysis is completed, the analyst has firm understanding of what is to be done. The next step is to decide how the problem might be solved. Thus, in system design, we move from the logical to the physical of the life cycle.

4.1. System Planning and the initial investigation

The most critical phase of managing system projects is planning to launch a system investigation, we need a plan detailing the steps to be taken, the people to be questioned and the outcome expected. The initial investigation has the objective of determining whether the user's request has potential merit. The major steps are defining user requirements. When the initial investigation is completed, the user receives a proposal summarizing the findings and the recommendation of the analyst.

4.2. Information Gathering

A key part of feasibility analysis is gathering information about the present system. The analyst knows what information to gather, where to find it, how to collect it and what to make of it. The proper use of tools for gathering information is the key to successful analysis. The tools are the traditional interview,

questionnaire, and on-site observation. We need to know, for example how to structure an interview, what makes up a questionnaire, and what to look for on-site observations. These tools when learned help analysis assess the effectiveness of the present system and provide the groundwork for recommending a billing system.

4.3. Feasibility Study

The main objective of the preliminary analysis is to identify the problem, evaluate the system concept of feasibility, and perform the economic and technical analyses perform the cost benefit analysis. After the clarification analysis the solution proposed it is checked that it is practical to implement that solutions. This is done through the feasibility study. It is checked for various aspect whether the proposed solution is technically or economically feasible or not. On the basis of which it has been categorized into four classes viz

- 1.) Technical
- 2.) Economic
- 3.) Legal
- 4.) Operational

The outcome of the preliminary analysis should be clear so that an alternate way to do the job can be found out.

4.3.1. Technical Feasibility

During the technical feasibility studies following issues are taken into consideration

- 1.Whether the required technology is available or not?
2. Required resources are available or not? (Manpower, programmer, software and hardware etc)

Once the technical feasibility is established, it is important to consider the monetary factors also. Since it might happen that developing a particular system may be technically possible but it may require huge investments and benefits may be less. For evaluating this, economic feasibility of the proposed system is carried out.

As in our proposed system our team has technically trained manpower with knowledge of developing the system. We are going to use web technology in our system, which is readily available. Software to be used is also available easily. So technically the project is feasible.

4.3.2. Economic Feasibility

For any system if the expected benefits equal or exceed the expected costs, the system can be judged to be economically feasible. In economic feasibility, cost benefit analysis is done in which expected costs and benefits are evaluated.

Economic analysis is used for evaluating the effectiveness of the proposed system. In economic feasibility, the most important is cost benefit analysis. As the name suggests, it is an analysis of the cost to be incurred in the system and benefits derivable out of the system.

As in our institute the hardware and software required for this type of system is already available so economically our project is feasible.

4.3.3. Cost Benefit Analysis

Developing an IT application is an investment. Since after developing that application it provides the organization with profits. Profits can be monetary or in the form of the improved working environment. However, it carries risks, because in some cases an estimate can be wrong. And the project might not actually turn out to be beneficial.

Cost Benefit Analysis helps to give management a picture of cost, benefits and risks. It usually involves comparing alternate investments.

Cost Benefit determines the benefits and saving that are expected from the system and compare them with the expected costs.

The cost of an information system involves the development cost and maintenance cost. The development costs are one-time investments whereas the

maintenance cost is recurring. The development cost is basically the cost incurred during the various stages of the system development.

Each phase of the life cycle has a cost. Some examples are:

1. Personal
2. Equipment
3. Supplies
4. Overheads etc.

In proposed system all hardware and Software are available in the company so cost is 0%.

4.3.4. Legal Feasibility

It includes study concerning contracts, liability, violation and legal other traps frequently unknown to the technical staff.

Chapter 5: Project Plan & Approach

5. Project Plan & Approach

Software development team consists of 3 members.

Phase-1: Requirement Gathering and Analysis

Gathering system requirement and prepare a System Requirement Specification document.

After collect the information it is analysed that the available resources can fulfil all the requirements. And it is also be examined that what resource will be used.

Phase-2: System Design

Make a detailed analysis of the system and prepare a System Design Document on the basis of SRS.

Phase-3: Prepare UTC/STC for testing the software

Prepare Unit Test Cases document. This document will be used to verify whether the functional requirements of the system have been met.

Phase-4: Develop the software

Develop the planned system

Phase-5: Test the software using the prepared UTC/STC and Rework if needed

Run your software programs using the respective UTC to verify & test the software.

Phase-6: Demonstrate the software to users & Implement it

Chapter 6: Project Schedule

PROJECT SCHEDULE

<i>Step #</i>	<i>Time Frame</i>
Phase 1. Requirement Gathering	18 Days
Phase 2. System Design	25 Days
Phase 3. Prepare UTC & STC for Testing the software	15 Days
Phase 4. Develop the software	32 Days
Phase 5. Test the software	13 Days
Phase 6. Demonstrate the software to users & implement it	12 Days

Chapter 7: Software Design

7.1. Scope

In this section we define the scope of the design effort. The design phase is an important part of the system development phase. A good design of the system needs creativity and flair from the designer and is the key to effective and successful engineering. The following are the basic objectives of the software design process:

- To describe the process of software design where informal ideas are transformed into detailed implementation description.
- Introduction of different stages in the design process.
- Understanding whether an Object Oriented or a Functional Oriented approach or both should be applied to the software.
- Determining and improving, cohesion control and coupling within subsystems.

In the project, the design phase has been identified as one of the most crucial documents. In this phase, we have identified the various aspect of the “USER MANAGEMENT”, which have to be implemented as subsystems and their further components.

7.2. System model specification

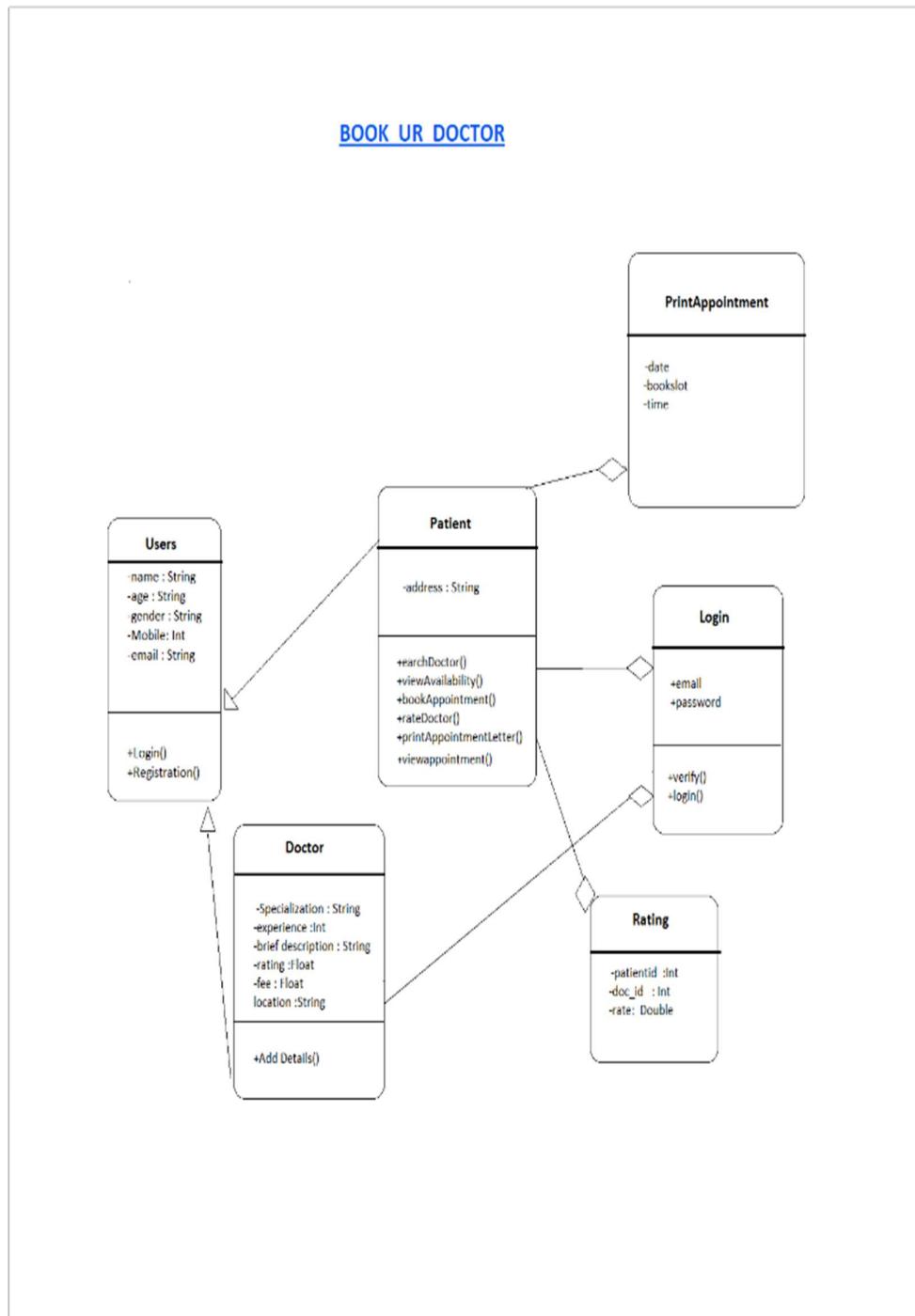
The system model chosen for the software project is the prototyping model. It begins with requirement gathering. The overall objectives of the software are well defined, known requirements are identified and an area where further definition is mandatory has been outlined. After this the software process is initiated and a prototype is built and then it is evaluated by the customer and used to refine requirements for the software to be developed. A prototype is a

working system that is developed to test the ideas about the new system and prototyping is a process of building a model of the system to be developed.

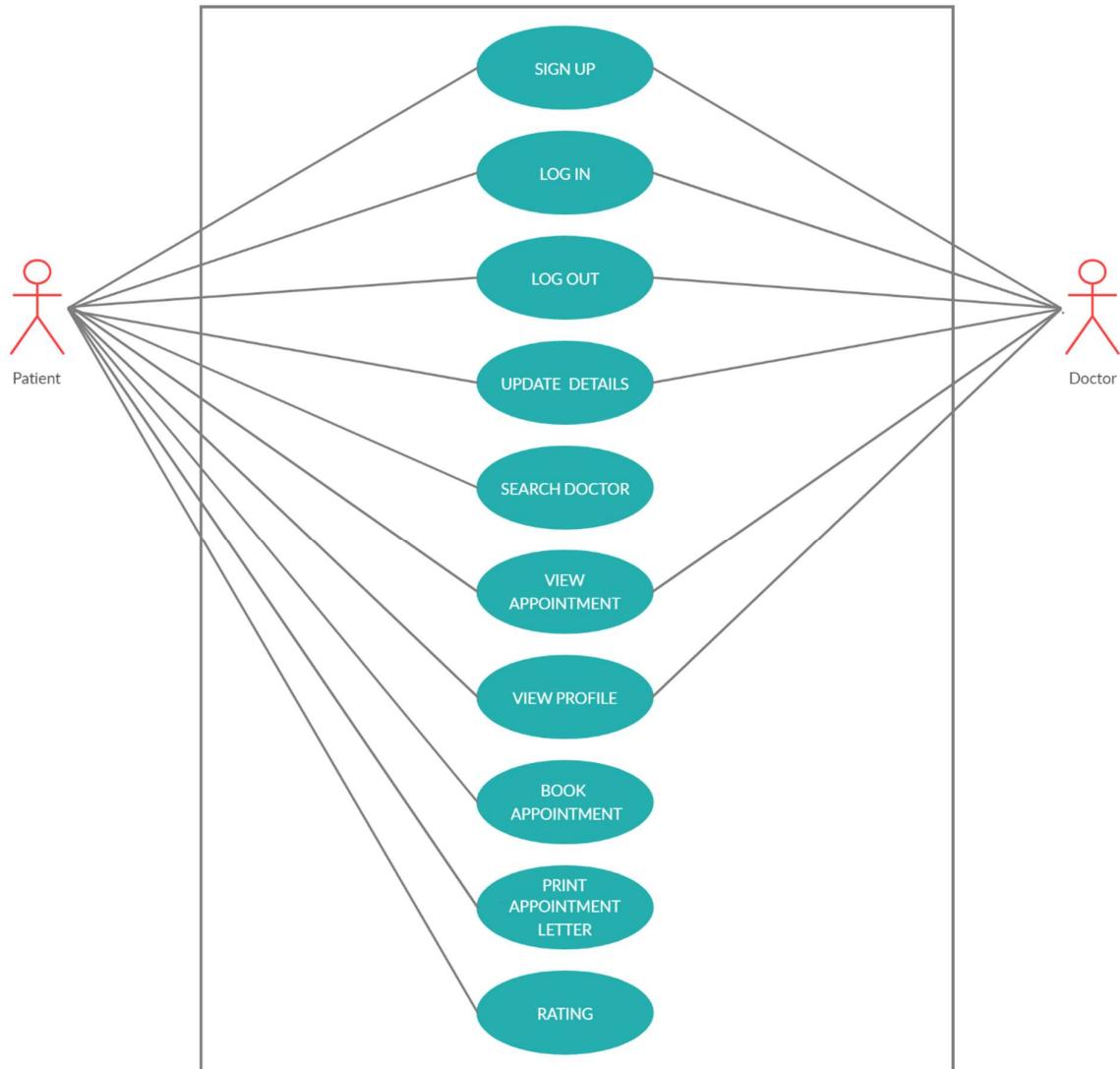
This approach is used in our project because it is difficult to know all the requirements in the beginning of the project. Such situation arises when no other system like proposed one is built earlier.

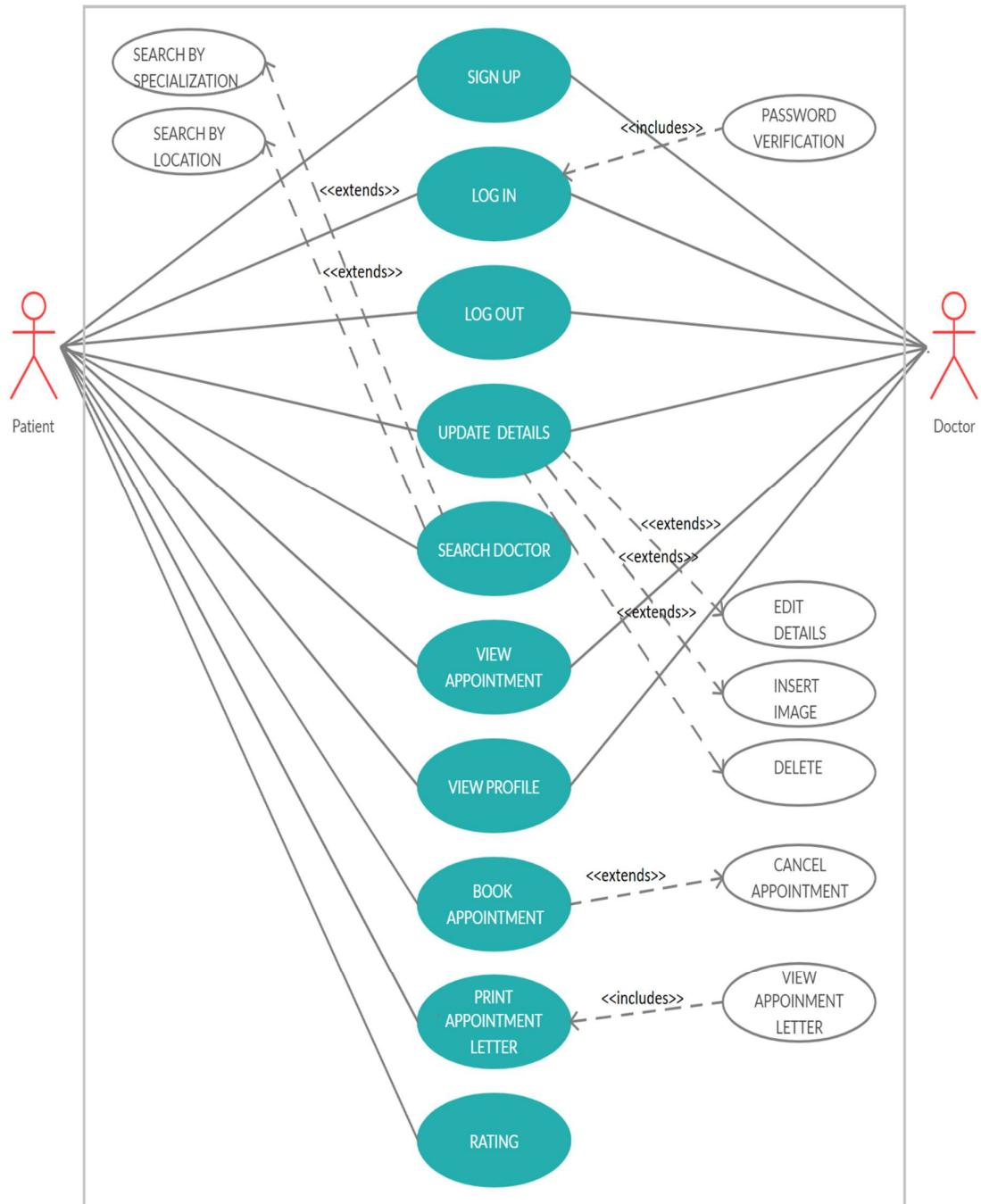
The complete system of our project is based on iterative model where the user keeps specifying the requirements according to the model made available and the changes are incorporated into the system to build a better model. In this type of approach, we will develop a prototype and show it to the user. The user verifies the prototype. In case if there are some suggestions from user then again that functionality is added to the system and again the user evaluates it. This cycle gets repeated till the time the user is completely satisfied with the prototype. Then the actual project is built and implemented thereafter.

7.3. Class Diagram



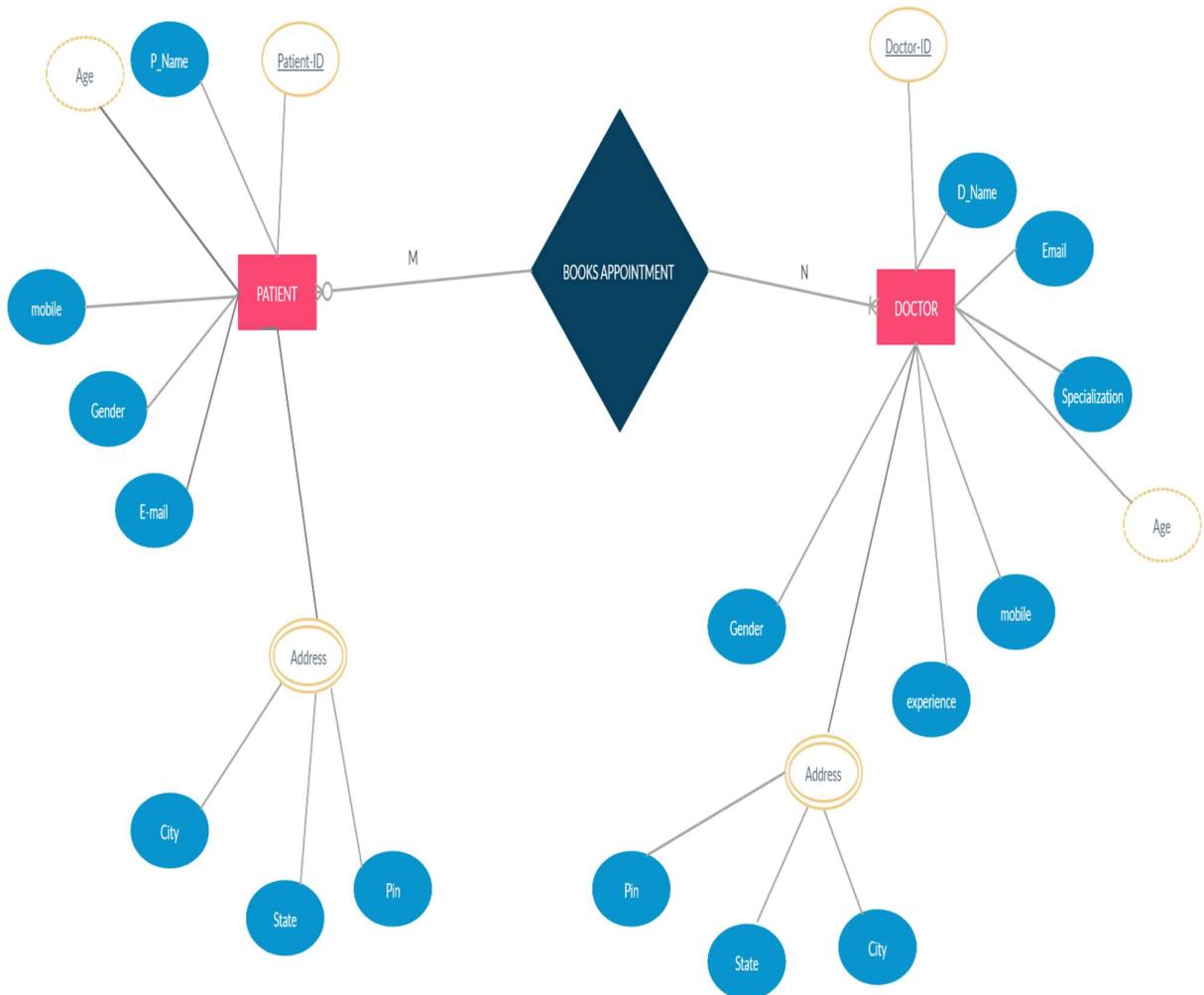
7.4 Use Case Diagram





7.5 ER Diagram

E-R Diagram of BookUrDoctor



7.6. Architectural Design

Large system is generally decomposed to smaller subsystems that account for functionality of the complete software system.

This process of identifying the sub systems and establishing a framework for sub system control and communication is called architectural design.

Architectural design is an important phase, as a bad architectural design cannot be rescued by good implementation. Following activities were performed during this stage:

- Principle sub systems that are functionally independent were identified and distinguished.
- A general model of control relationships between system parts was established.

Each subsystem was further decomposed into their sub function.

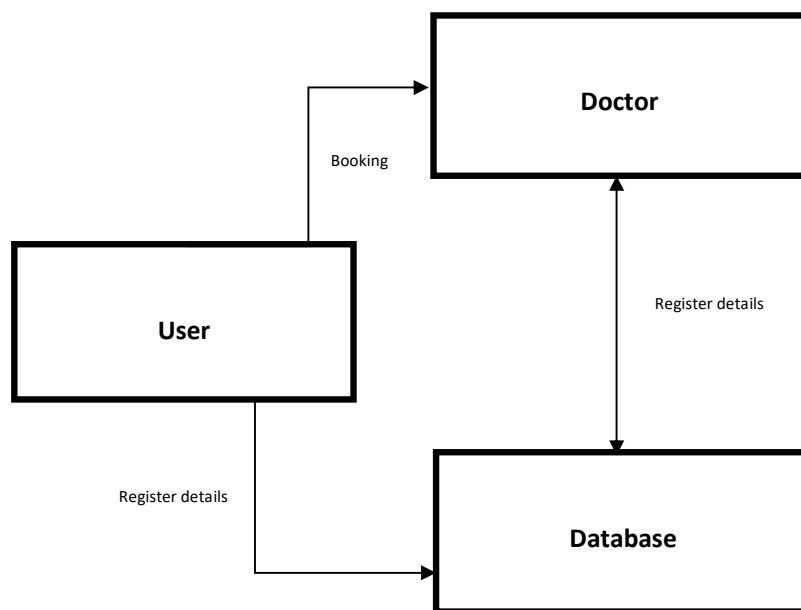


Fig. Architecture of BookUrDoctor System

8. Use case Description

Use Case Name: Signup

Actors: Doctor & Patient

Description: Doctor & Patient must sign up using their name, mobile no, email & password.

The flow of Events:

A. Basic Flow

- 1)The user chooses to Sign up Now Button.
- 2)The system prompts for registration from.
- 3)The user must fill the form using their name and their unique mobile no, email & password.
- 4)The system verifies the unique mobile no & email if those are never been used in that system before. After getting the unique mobile no & email the system will create a new account of that user.

B. Alternative Flows

- 1) Invalid Mobile No/Email: If the mobile no or email entered by Doctor or Patient have already registered before or if it's invalid then it displays registration unsuccessful.

Special Requirements:

- 1) Minimum password length is 8 characters and must include a combination of characters including at least one non-alphabetic character.

Pre-conditions:

User must have a valid email and mobile no.

Post-conditions:

1. Signup Success: The system displays the main interface from which additional actions can be initiated by the user.
2. Signup Failure: If the Signup fails as described in any of the alternative flows above, an appropriate message is displayed and the user is not considered authenticated.

Use Case Name: Login

Actors: Doctor & Patient

Description: A registered user can login to the system and once authenticated, can initiate subsequent actions.

Flow of Events

A. Basic Flow

- 1) The registered user chooses to Login.
- 2) The system prompts for username (registered email/mobile no) and password.
- 3) The registered user submits username and password.
- 4) The system verifies the username and password combination. It authorizes the registered user according to the role(s) to which the user has been assigned. Then it displays the main interface and awaits subsequent action from the user.

B. Alternative Flows

- 1) **Invalid Username/Password:** The system displays the Authentication Failed message.

Pre-conditions

User must have registered.

Post-conditions

1. **Login Success:** The system displays the main interface from which additional actions can be initiated by the user.
2. **Login Failure:** If the Login fails as described in any of the alternative flows above, an appropriate message is displayed and the user is not considered authenticated.

Use Case Name: LogOut

Actors: Doctor & Patient

Description: A registered user can logout from the system and once directed, can drawback from subsequent actions.

Flow of Events

A. Basic Flow

- 1) The registered user chooses to Logout.
- 2) The user clicks on the logout button.
- 3) The system verifies the user and signs him out of the system. Then it displays the main interface and awaits subsequent action from the user.

B. Alternative Flows

- 1) **Invalid Username:** The system displays the logout Failed message.

Pre-conditions

User must have registered.

Post-conditions

1. **Logout Success:** The system displays the main interface from which again the login actions can be initiated by the user.
2. **Login Failure:** If the Logout fails as described in any of the alternative flows above, an appropriate message is displayed and the user is not signed out.

Use Case Name: Update Details

Actors: Doctor & Patient

Description: Doctors will be able to add their details such as their name, age, gender, phone no, email along with their specialization, fee and also with a brief description of them. Apart from those the doctor must update their time table that on which day and they are coming to which clinic.

Patient will be able to add their details such as their name, age, gender, phone no, email along with their address.

The flow of Events:

A. Basic Flow

- 1)The user chooses to Login Now Button.
- 2)The system displays the main interface from which doctors could view their profile and choose the update details button.
- 3)The doctor must update the details by their name, and along with their designation, specialization, experiences and also with a brief description of them.The user must update the details by their name, and along with their address.
- 4)The doctor must update time to time that on which day and on which date they are coming to which clinic.
- 5)After filling the form user should press the update details button to save their details in the system.

B. Alternative Flows

- 1) Invalid Details: If the details entered by User is invalid then the system should notify him.

Pre-conditions:

- 1)The doctor must have registered.
- 2)The doctor must have unique registration no.

Post-conditions

1. Add Details Successfully: The system displays profile preview with updated details successfully.
2. Add Details Failure: If the Update details fail as described in any of the alternatives flows above, an appropriate message is displayed and the details are not updated.

Use Case Name: Search Doctor

Actors: Patient

Description: Patient can search for top experienced doctors from dermatologists, medicine specialists, gynecologists, oncologists and more as well as the renewed surgeon using this app or by popular doctor's specialization or by their location.

The flow of Events:

A. Basic Flow

- 1)The user chooses to Login Now/Skip Now Button.
- 2)The system displays the main interface from which the patient could search for the doctor by their name.
- 3)The patient could search for doctors according to their specialisation.
- 4)The patient could search for doctors according to their location.

B. Alternative Flows

- 1) Search Unsuccessful: If the name entered by the patient is not found in the database then the system will show a message as not found.

Pre-conditions:

- 1) The patient should enter a valid name in the search engine.

Post-conditions:

1. Search Successful: The system displays the profile of those doctors searched by the patient.
2. Search Failure: If the searching fails as described in any of the alternatives flows above, an appropriate message is displayed as please input the correct name.

Use Case Name: View Appointment

Actor: Patient & Doctor

Description: A registered user will look for the scheduled appointments.

Flow of Events:

A. Basic Flow

1. Once the slot is booked a preview of the booked slot will be displayed.
2. The preview will be having a my appointments navigation option.

3. The registered user selects to view and the appointment preview will be displayed.

B. Alternative Flow:

1. If there is no appointment the preview will display no appointments message.

Pre-Condition:

1. The user must be registered.
2. The user must be having appointments.

Post-Condition:

1. Success: The appointments are displayed according to the schedule.
2. Failure: The appointment preview page is not displayed.

Use Case Name: View Profile

Actors: Patient and Doctor

Description: User could view their profiles.

Flow of Events

A. Basic Flow:

- The registered user chooses to view profile .
- The system displays the profile page of the user.
- The user can update the profile.
- The user can delete his profile.

B. Alternate Flow

The system will not prompt the profile page.

Special Requirements:

The user must have added his details.

Pre-Conditions:

The user must be a registered user.

Post-Conditions:

The system directs the user to the profile page of the doctor.

Use Case Name: Book Appointment

Actors: Patient

Description: A registered user can book an appointment for his required date and time.

Flow of Events

A. Basic Flow

1.register user chooses on book appointment.

- system prompts a search icon to enter either disease name or the doctor name and also below the icon the system displays some of the frequent disease or doctor name.
- user submits either the name of the doctor or the disease name.
- system generates a list of available slots for booking an appointment.
- The user chooses on a slot of his requirement among the list and proceed further.
- If 15 patient already have registered then appropriate message will be displayed.
- The system generates appointment no. for future reference.
- The system asks for another appointment if any.

B. Alternative Flow

If the user chooses to cancel the appointment process in between an appropriate message will be prompted.

Pre-Conditions:

1.The user must have registered.

Post-Conditions:

1.Success: The system generates the appointment preview.

2.Failure: The system cancels the appointment if any delay arises during the transaction.

Use Case Name: Print Appointment Letter

Actor: Patient

Description: A registered user will print the appointment letter.

Flow of Events:**A. Basic Flow**

1. Once the slot is booked a preview of the booked slot will be displayed.
2. The preview will be having a view appointment letter option.
3. The appointment letter will be having a print appointment letter option.
4. The registered user selects to print and the appointment preview will be printed.

B. Alternative Flow:

1. If there is no printer the appointment preview will be downloaded as a softcopy.

Special Requirements:

1. The user should be having a printer.

Pre-Condition:

1. The user should have a booked slot.

Post-Condition:

1. Success: The document is downloaded or printed successfully.
2. Failure: The document is neither printed nor downloaded.

Use Case Name: Rating**Actors:** Patient

Description: A registered user rate the doctor after the appointment date.

Flow of Events**A. Basic Flow**

1. register user chooses on rating.
2. system prompts a new window where the patient can rate .
3. user submits the rate value.
4. system generates a alert box where “Thanks for rating will be prompted”.

5. The system redirects to the previous page.
6. The patient can give rating to other doctor if any.

B. Alternative Flow

The user can cancel the rating process in between.

Pre-Conditions:

1. The user must have registered.

Post-Conditions:

1. Success: The system directs to the doctors page.

Chapter 9. Coding/Code Templates

Android Platform:

Home Page:

Frontend:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#ffff"
    android:padding="10dp"
    tools:context=".MainActivity">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="90dp"
        android:layout_marginTop="30dp"
        android:src="@drawable/logo"
        android:scaleType="fitCenter"
    ></ImageView>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:weightSum="2"
        android:layout_marginTop="50dp"
        android:layout_weight="3">

        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:textAlignment="center"
            android:orientation="vertical">

            <TextView
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:textAlignment="center"
                android:fontFamily="@font/abhaya_libre_extrabold"
                android:text="Are you a doctor? "
                android:textColor="#050505"
                android:textSize="20sp"
                android:textStyle="bold" />

            <Button
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:layout_alignParentEnd="true"
                android:layout_centerVertical="true"
                android:padding="10dp"
                android:drawableTop="@drawable/ic_doctor"
                android:background="@drawable/round_btn"
                android:fontFamily="@font/maven_pro_medium"
                android:onClick="doctorRegister btn" />
        
```

```

        android:textStyle="bold"
        android:layout_marginRight="2dp"
        android:layout_marginTop="5dp"
        android:textColor="#FF6C02"
        android:text="R e g i s t e r
Here"
        android:textAlignment="center"
        android:textSize="13sp"/>

    </LinearLayout>

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:orientation="vertical">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:fontFamily="@font/abhaya_libre_extrabold"
            android:text="Hello, User"
            android:textAlignment="center"
            android:textColor="#050505"
            android:textSize="20sp"
            android:textStyle="bold" />

        <Button
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_alignParentEnd="true"
            android:layout_centerVertical="true"
            android:layout_marginLeft="2dp"
            android:padding="10dp"
            android:layout_marginTop="5dp"
            android:drawableTop="@drawable/ic_man"
            android:background="@drawable/round_btn"
            android:fontFamily="@font/maven_pro_medium"
            android:id="@+id/main_join_now_btn"
            android:text="S i g n   U p
Now"
            android:textAlignment="center"
            android:textStyle="bold"
            android:textColor="#FF6C02"
            android:textSize="13sp" />

    </LinearLayout>

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:fontFamily="@font/abhaya_libre_extrabold"
    android:text="OR"
    android:layout_marginBottom="10dp"
    android:textAlignment="center"
    android:textColor="#9B9FBB"
    android:textSize="15sp"
    android:textStyle="bold" />

<TextView
    android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        android:fontFamily="@font/abhaya_libre_extrabold"
        android:text="Already have an account ?"
        android:layout_marginBottom="10dp"
        android:textAlignment="center"
        android:textColor="#050505"
        android:textSize="20sp"
        android:textStyle="bold" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textAlignment="center"
        android:layout_centerVertical="true"
        android:layout_alignParentEnd="true"
        android:fontFamily="@font/maven_pro_medium"
        android:id="@+id/main_login_btn"
        android:background="@drawable/round_btn"
        android:textColor="#FF6C02"
        android:textStyle="bold"
        android:text="Login Now"
        android:layout_marginBottom="80dp"
        android:textSize="13sp" />

    <!--
    <Space
        android:layout_width="match_parent"
        android:layout_height="10dp" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="@font/abhaya_libre_extrabold"
        android:text="Be My Visitor ?"
        android:textAlignment="center"
        android:textColor="#050505"
        android:textSize="17sp"
        android:textStyle="bold" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_alignParentEnd="true"
        android:fontFamily="@font/maven_pro_medium"
        android:onClick="skip_btn"
        android:textAlignment="center"
        android:background="@drawable/round_btn"
        android:textColor="#FF6C02"
        android:layout_marginBottom="30dp"
        android:textStyle="bold"
        android:text="skip Now"
        android:textSize="13sp" />
-->
</LinearLayout>
```

Backend:

```
package com.example.demo;
```

```

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.app.AlertDialog;
import android.content.Context;
import android.content.Intent;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import com.example.demo.Model.Users;
import com.example.demo.Prevalent.Prevalent;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import io.paperdb.Paper;

public class MainActivity extends AppCompatActivity {

    private Button joinNowButton, loginButton;
    private ProgressDialog loadingBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        checkConnection();

        joinNowButton = (Button) findViewById(R.id.main_join_now_btn);
        loginButton = (Button) findViewById(R.id.main_login_btn);
        loadingBar = new ProgressDialog(this);

        Paper.init(this);

        loginButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view)
            {
                Intent intent = new Intent(MainActivity.this, LoginActivity.class);
                startActivity(intent);
            }
        });

        joinNowButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view)
            {
                Intent intent = new Intent(MainActivity.this, SignupActivity.class);
                startActivity(intent);
            }
        });
    }

    String UserPhoneKey = Paper.book().read(Prevalent.UserPhoneKey);
    String UserPasswordKey = Paper.book().read(Prevalent.UserPasswordKey);

    if (UserPhoneKey != "" && UserPasswordKey != "")
    {
        if (!TextUtils.isEmpty(UserPhoneKey) &&

```



```

                startActivity(intent);
            }
        } else {
            loadingBar.dismiss();
            Toast.makeText(MainActivity.this, "Password is incorrect.", Toast.LENGTH_SHORT).show();
        }
    } else {
        Toast.makeText(MainActivity.this, "Account with this " + phone + " number do not exists.", Toast.LENGTH_SHORT).show();
        loadingBar.dismiss();
    }
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
    ...
}
}

public void doctorRegister_btn(View view) {
    Intent intent=new Intent(MainActivity.this,DoctorRegistrationForm.class);
    startActivity(intent);
}

/*
public void skip_btn(View view) {
    Intent intent=new Intent(MainActivity.this,CreateDoctorProfile.class);
    startActivity(intent);
}

*/
}

```

Doctor Add Details:

Frontend:

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/background"
    android:orientation="vertical"
    tools:context=".Upload_Doctors">

    <LinearLayout
        android:layout_width="match_parent"
        android:padding="30dp"
        android:orientation="vertical"
        android:layout_height="wrap_content">

```

```
<ImageView
    android:layout_width="match_parent"
    android:layout_height="200dp"
    android:scaleType="centerCrop"
    android:id="@+id/iv_doctorImage"
    android:src="@drawable/imagenotavailable"
/>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Full Name"
    android:textSize="15sp"
    android:layout_marginTop="20dp"
    android:id="@+id/txt_doctor_name"
/>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Specialization"
    android:layout_marginTop="10dp"
    android:textSize="15sp"
    android:id="@+id/text_specialization"
/>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Gender"
    android:textSize="15sp"
    android:layout_marginTop="10dp"
    android:id="@+id/txt_doctor_gender"
/>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="15sp"
    android:hint="Brief Description (Experience,Age,Achievements etc..)"
    android:layout_marginTop="10dp"
    android:id="@+id/txt_doctor_location"
/>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Days"
    android:textSize="15sp"
    android:layout_marginTop="10dp"
    android:id="@+id/txt_doctor_days"
/>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Price"
    android:inputType="number"
    android:textSize="15sp"
    android:layout_marginTop="10dp"
    android:id="@+id/txt_doctor_price"
/>

<EditText
    android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:hint="Locations (With Full Address Of That Pharmacy)"
        android:layout_marginBottom="10dp"
        android:layout_marginTop="10dp"
        android:textSize="15sp"
        android:id="@+id/text_description"
    />
<TextView
    android:layout_marginTop="5dp"
    android:layout_marginBottom="5dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="#F6C02"
    android:fontFamily="@font/amethysta"
    android:text="*All fields are mandatory"
    android:textAlignment="textEnd"
    android:textSize="10sp"
    android:textStyle="bold" />

<Button
    android:layout_marginTop="13dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="S e l e c t   I m a g e"
    android:textSize="13sp"
    android:textAlignment="center"
    android:fontFamily="@font/maven_pro_medium"
    android:background="@drawable/round_button"
    android:textColor="#fff"
    android:onClick="btnSelectImage"
/>

<Space
    android:layout_width="match_parent"
    android:layout_height="10dp" />
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="U p l o a d"
    android:textAlignment="center"
    android:fontFamily="@font/maven_pro_medium"
    android:textSize="13sp"
    android:background="@drawable/round_button"
    android:textColor="#fff"
    android:onClick="btnUploadDoctors"
/>

</LinearLayout>
</ScrollView>
```

Backend:

```

package com.example.demo;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.app.AlertDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
```

```

import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.storage.FirebaseStorage;
import com.google.firebaseio.storage.StorageReference;
import com.google.firebaseio.storage.UploadTask;

import java.text.DateFormat;
import java.util.Calendar;

public class Upload_Doctors extends AppCompatActivity {

    ImageView doctorImage;
    Uri uri;
    EditText txt_name, txt_specialization, txt_gender, txt_locations, txt_days,
    txt_price, txt_description;
    String imageUrl;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_upload_doctors);

        doctorImage = (ImageView) findViewById(R.id.iv_doctorImage);
        txt_name = (EditText) findViewById(R.id.txt_doctor_name);
        txt_specialization = (EditText) findViewById(R.id.text_specialization);
        txt_gender = (EditText) findViewById(R.id.txt_doctor_gender);
        txt_locations = (EditText) findViewById(R.id.txt_doctor_location);
        txt_days = (EditText) findViewById(R.id.txt_doctor_days);
        txt_price = (EditText) findViewById(R.id.txt_doctor_price);
        txt_description = (EditText) findViewById(R.id.text_description);
    }

    public void btnSelectImage(View view) {

        Intent photoPicker = new Intent(Intent.ACTION_PICK);
        photoPicker.setType("image/*");
        startActivityForResult(photoPicker, 1);
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        super.onActivityResult(requestCode, resultCode, data);

        if (resultCode == RESULT_OK) {
            uri = data.getData();
            doctorImage.setImageURI(uri);
        } else Toast.makeText(this, "You Have Not Picked Any Image",
        Toast.LENGTH_SHORT).show();
    }

    public void uploadImage() {

        StorageReference storageReference =
        FirebaseStorage.getInstance().getReference().child("DoctorsImage")
        .child(uri.getLastPathSegment());

        final ProgressDialog progressDialog = new ProgressDialog(this);
        progressDialog.setMessage("Details Uploading....");
    }
}

```

```

        progressDialog.show();

        storageReference.putFile(uri).addOnSuccessListener(new
OnSuccessListener<UploadTask.TaskSnapshot>() {
            @Override
            public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

                Task<Uri> uriTask = taskSnapshot.getStorage().getDownloadUrl();
                while (!uriTask.isComplete());
                Uri urlImage = uriTask.getResult();
                imageUrl = urlImage.toString();
                uploadDoctors();

                progressDialog.dismiss();
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                progressDialog.dismiss();
            }
        });
    }

    public void btnUploadDoctors(View view) {

        if (TextUtils.isEmpty(txt_name.getText().toString())) {
            Toast.makeText(this, "Please provide appintro_search_doctor full name.", Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(txt_specialization.getText().toString())) {
            Toast.makeText(this, "Please provide appintro_search_doctor specialization.", Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(txt_gender.getText().toString())) {
            Toast.makeText(this, "Please provide gender.", Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(txt_locations.getText().toString())) {
            Toast.makeText(this, "Please provide a short description.", Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(txt_days.getText().toString())) {
            Toast.makeText(this, "Please provide on which days you are going to this clinic.", Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(txt_price.getText().toString())) {
            Toast.makeText(this, "Please provide your fee.", Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(txt_description.getText().toString())) {
            Toast.makeText(this, "Please provide the location of that pharmacy with its full address.", Toast.LENGTH_SHORT).show();
        } else {
            uploadImage();
        }
    }

    public void uploadDoctors(){

        DoctorDetails doctorDetails = new DoctorDetails(
            txt_name.getText().toString(),
            txt_specialization.getText().toString(),
            txt_gender.getText().toString(),
            txt_locations.getText().toString(),
            txt_days.getText().toString(),
            txt_price.getText().toString(),
            txt_description.getText().toString(),
            imageUrl
        );
    }
}

```

```
String myCurrentDateTime = DateFormat.getDateTimeInstance()
    .format(Calendar.getInstance().getTime());
```

```
FirebaseDatabase.getInstance().getReference("Doctors")
```

```
.child(myCurrentDateTime).setValue(doctorDetails).addOnCompleteListener(new
OnCompleteListener<Void>() {
    @Override
    public void onComplete(@NonNull Task<Void> task) {

        if (task.isSuccessful()){
            Toast.makeText(Upload_Doctors.this, "Doctor Details Upload
Successfully", Toast.LENGTH_SHORT).show();
            finish();
        }

    }
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {
        Toast.makeText(Upload_Doctors.this, e.getMessage().toString(),
Toast.LENGTH_SHORT).show();
    }
});
```

Web Platform:

Home Page:

Frontend:

```
<?php
session_start();
    include 'Header.php'
?>

<!DOCTYPE html>
<html>
    <head>
        <title>HOME PAGE</title>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
        <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
        <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>
        <script type="js/bootstrap.js"></script>
    </head>
    <body>
        <div style=" border-style: none none solid none; border-width: 1px; border-color:#C2BABA;"></div>
        <br>
        <div class="container">
            <div class="row">
                <form action="Search.php" method="post" class="form-inline">
                    <div class="input-group">
                        <input name="locval" style="height:50px; width:300px; border:1.5px solid gray;" class="form-control" placeholder="Location">
                        <span class="input-group-addon"><button name="loc" class="btn" style="border:1.5px solid gray; height:50px; width:100px;"><i style="" class="fa fa-map-marker fa-2x"></i></button></span>
                    </div>
                    <div class="input-group">
                        <input name="spval" style="height:50px; width:500px; border:1.5px solid gray;" class="form-control" placeholder=" Doctors">
                        <span class="input-group-addon"><button name="sp" class="btn" style="border:1.5px solid gray; height:50px; width:100px;"><i class="fa fa-search fa-lg"></i></button></span>
                    </div>
                </form>
            </div>
            </div>
        <br><br>
```

```
<div class = "myslide" style="margin-left:200px; margin-right:200px;">
<div id="demo" class="carousel slide" data-ride="carousel">

    <ul class="carousel-indicators">
        <li data-target="#demo" data-slide-to="0" class="active"></li>
        <li data-target="#demo" data-slide-to="1"></li>
        <li data-target="#demo" data-slide-to="2"></li>
    </ul>

    <div class="carousel-inner">
        <div class="carousel-item active">
            
        </div>
        <div class="carousel-item">
            
        </div>
        <div class="carousel-item">
            
        </div>
    </div>

    <a class="carousel-control-prev" href="#demo" data-slide="prev">
        <span class="carousel-control-prev-icon"></span>
    </a>
    <a class="carousel-control-next" href="#demo" data-slide="next">
        <span class="carousel-control-next-icon"></span>
    </a>
</div>
<br>
<br>
```

```

<br>
<div class="doctors" style="margin-left:200px; margin-right:200px;">
<center>
    <div class = "container">
        <div class="row">
            <h3 style ="color:black;font-size:400%; font-family:Book Antiqua">Find Doctor in top specialities</h3>
        </div>
    </div>
    <br>
    <br>
<div class="doctors" style="margin-left:200px; margin-right:200px;">
    <div class = "container">
        <div class = "row">
            <div class="col-md-3">
                <br>
                <p><b><a href="#">DENTIST</a></b></p>
                Oral issues?<br>Schedule a dental Checkup</div>

                <div class="col-md-3">
                    <br>
                    <p><b><a href="Gynaecologist.php">GYNECOLOGIST/OBSTETRICIAN</a></b></p>
                    Explore for women's health, pregnancy and infertility treatment</div>

                    <div class="col-md-3">
                        <br>
                        <b><a href="#">DIETITIAN/NUTRITION</a></b></p>
                        Get guidance on eating right, weight management and sports nutrition</div>

                    <div class="col-md-3">
                        <br>
                        <p><b><a href="#">PHYSIOTHERAPIST</a></b></p>
                    Pulled a muscle?<br>Get it treated by physiotherapist</div>
                </div>
            </div>
        </div>
    </div>
</div>
<br>

```

```

<div class = "row">
    <div class="col-md-3">
        <br>
        <p><b><a href="#">GENERAL SURGEON</a></b></p>
    Need to get operated?<br>Find the right surgeon</div>

    <div class="col-md-3">
        <br>
        <p><b><a href="#">ORTHOPEDIST</a></b></p>
    For bone and joint issues, spinal injuries and more</div>

    <div class="col-md-3">
        <br>
        <p><b><a href="#">GENERAL PHYSICIAN</a></b></p>
    Find the right family doctor</div>

    <div class="col-md-3">
        <br>
        <p><b><a href="#">PEDIATRICIAN</a></b></p>
    Child specialist and Doctors for infant</div>
</div>
</div>
<br>
<br>
<div class="doctors" style="margin-left:200px; margin-right:200px;">
    <div class = "container">
        <div class = "row">
            <div class="col-md-3">
                <br>
                <p><b><a href="#">IMMUNOLOGIST</a></b></p>
            Fight your allergy<br>Get your immune system ready</div>

            <div class="col-md-3">
                <br>
                <p><b><a href="Cardiologist.php">CARDIOLOGIST</a></b></p>
                Unhealthy heart?<br>We will take care</div>
            </div>
        </div>
    </div>
</div>

```

Book Appointment:

Frontend:

```
<script>
$(document).ready(function(){
    var date_input=$('#input[name="date"]'); //our date input has the name "date"
    var container($('.bootstrap-iso form').length>0 ? $('.bootstrap-iso form').parent() : "body");
    var options={
        format: 'dd/mm/yyyy',
        container: container,
        todayHighlight: true,
        autoclose: true,
    };
    date_input.datepicker(options);
})
</script>
</head>
<body>
<br>
<div class="container">

<div id="myf" class="row" style="background:black;">

<form class="col-xs-12 col-sm-12 col-md-12" method="post">
<br>
<h2 align="center">Please provide patient details</h2>
<br>
<div style=" margin-left:25px;" class="form-group">
    <p class="tx">Select Slot</p>

    <input type="radio" name="slot" value="7A.M-9A.M" /><label class="tx">&nbsp;7A.M-9A.M</label>
        <input type="radio" style="margin-left:100px;" name="slot" value="12P.M-2P.M" /><label class="tx">&nbsp;12P.M-2P.M</label>
        <input type="radio" style=" margin-left:100px;" name="slot" value="6P.M-8P.M" /><label class="tx">&nbsp;6P.M-8P.M</label>
</div>
<div class="form-group" <!-- Date input -->
    <input class="form-control" id="date" name="date" placeholder="Appointment Date(DD/MM/YYYY)" type="text"/>
</div>
<br>
```

```
<div class="form-group">
  <input type="text" class="form-control" id="name" placeholder="Patient's Full Name" name="name" required>
</div>
<br>
<div class="form-group">
  <input type="text" class="form-control" id="gender" placeholder="Gender (Male/Female/Others)" name="gender" required>
</div>
<br>
<div class="form-group">
  <input type="text" class="form-control" id="phone" placeholder="Enter age" name="age" required>
</div>
<br>
<div class="form-group">
  <input type="phone" class="form-control" id="phone" placeholder="Phone Number" name="phone" required>
</div>
<br>
<div class="form-group">
  <textarea rows="3" cols="40" class="form-control" id="home" placeholder="Home Address" name="home" required></textarea>
</div>
<br>
<div class="form-group">
  <input type="text" class="form-control" id="city" placeholder="City Name" name="city" required>
  <input type="hidden" class="txt" name="id" value=<?php echo $id; ?>>
</div>
<br>
<br>
```

```
</div>
<br>
<div class="form-group">

    <input type="text" class="form-control" id="city" placeholder="City Name" name="city" required>
    <input type="hidden" class="txt" name="id" value="php echo $id; ?&gt;"&gt;
&lt;/div&gt;
&lt;br&gt;
&lt;br&gt;

    &lt;button type="submit" name="confirm" class="btn btn-success bt-lg" style="width:335px; height:50px; margin-right:140px;"&gt;CONFIRM
        BOOKING&nbsp;&ampnbsp;&ampnbsp;&lt;i class="fas fa-check-circle"&gt;&lt;/i&gt;&lt;/button&gt;

&lt;br&gt;
&lt;br&gt;
&lt;br&gt;
&lt;br&gt;
&lt;/form&gt;
&lt;br&gt;

&lt;/div&gt;
&lt;br&gt;
&lt;br&gt;

&lt;br&gt;
&lt;br&gt;
&lt;br&gt;

&lt;!--
&lt;?php
    include 'Footer.php';
?
--&gt;

&lt;br&gt;
&lt;br&gt;
&lt;br&gt;

&lt;/body&gt;
&lt;/html&gt;</pre
```

Book Appointment:

Back end:

```

<?php
session_start();
include 'connection.php';
include 'Header.php';
$_sess=$_SESSION['u_mob'];
if(isset($_POST['book'])){
    $_SESSION['docid']=$_POST['book'];
}
$id=$_SESSION['docid'];
if(isset($_REQUEST['confirm'])){
    $bname=$_REQUEST['name'];
    $bago=$_REQUEST['age'];
    $bgender=$_REQUEST['gender'];
    $bphone=$_REQUEST['phone'];
    $bhome=$_REQUEST['home'];
    $bcity=$_REQUEST['city'];
    $slot=$_REQUEST['slot'];
    $date=$_REQUEST['date'];
    $result = mysqli_query($conn,"select count(bookid) 'ids' from bookdoc where docid='$id' and bookdate='$date' and bookslot='$slot'");
    if($result>1){
        echo "<script>alert('No more slots available.Look for another date or another slot.')</script>";
        echo "<script>location.href='front.php' </script>";
    }
}
else{
    $sql2="INSERT INTO `bookdoc`(`bookid`, `usermob`, `uname`, `uage`, `ugender`, `uphone`, `uaddress`, `ucity`, `bookdate`, `bookslot`, `drid`) VALUES (NULL,'$sess','$bname','$bago','$bgender','$bphone','$bhome','$bcity','$date','$slot','$id')";
    if(mysqli_query($conn,$sql2)){
        echo "<script>location.href='UserAppointment.php' </script>";
    }
    else{
        echo "error";
    }
}
?>

```

Rating:

Front end:

```

35
36  <!DOCTYPE html>
37
38
39
40  <html>
41  <head>
42      <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
43          integrity="sha384-Vkoo8x4CGs03+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">
44      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/rateyo/2.3.2/jquery.rateyo.min.css">
45      <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
46      <script src="https://cdnjs.cloudflare.com/ajax/libs/rateyo/2.3.2/jquery.rateyo.min.js"></script>
47      <script src="rateyo.js"></script>
48      <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css" integrity="
49          sha384-UHRtZLI+pbxtHCWp1t77Bi1L4ZtiqrqD80Kn4Z8NTSRyMA2Fd3n5dQ8lWUE00s/" crossorigin="anonymous">
50
51  <title>RATING SYSTEM</title>
52  </head>
53  <body >
54  <nav class="navbar navbar-expand-lg navbar-light bg-light">
55      <a class="navbar-brand" href="#"></a>
56      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav"
57          aria-expanded="false" aria-label="Toggle navigation">
58          <span class="navbar-toggler-icon"></span>
59      </button>
60      <div class="collapse navbar-collapse" id="navbarNav">
61          <ul class="navbar-nav ml-auto">
62              <li class="nav-item active">
63                  <a class="nav-link" href="front.php">&nbsp;&nbsp;<i class="fa fa-home"></i>&nbsp;HOME&nbsp;&nbsp; <span
64                      class="sr-only">(current)</span></a>
65              </li>
66              <li class="nav-item active">

```


Rating:

Back end:

```
1  <?php
2  session_start();
3  require 'connection.php';
4  if(isset($_POST['rate'])){
5      $docid=$_POST['rate'];
6      //echo $docid;
7      $_SESSION['id']=$docid;
8      //echo $_SESSION['id'];
9  }
10
11 if(isset($_POST['add']))
12 {
13     $rating=$_POST['rating'];
14     //echo $rating;
15     $docid= $_SESSION['id'];
16     $sql="INSERT INTO ratingapp(ratevalue,doc_id) VALUES ('$rating','$docid')";
17
18     if (!mysqli_query($conn,$sql))
19     {
20         echo "Error" .sql. "<br>" .mysql_error();
21
22     }
23     else{
24         header('location:Doctors.php');
25     }
26
27
28
29     mysqli_close($conn);
30 }
31
32
33 ?>
```

Chapter 10: Testing

Test Case ID	Description	Step Name	Step Description	Expected Result	Actual Result	Remarks
TC_001	Verify that Password field does not allow only letters.	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as only lowercase or uppercase letters or combination of both and greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_002	Verify that Password field does not allow only numbers.	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as only numbers [0-9] with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_003	Verify that Password field does not allow only special Characters.	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		

		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as only special symbols with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_004	Verify that Password field does not allow only letters and numbers	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as only lowercase or uppercase letters or combination of both along with numbers [0-9] with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_005	Verify that Password field does not allow only letters and numbers	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		

		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as only lowercase or uppercase letters or combination of both along with numbers [0-9] with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_006	Verify that Password field does not allow only letters and special symbols	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as only lowercase or uppercase letters or combination of both along with special symbols with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_007	Verify that Password field does not allow only numbers and special symbols	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		

		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as numbers [0-9] along with special symbols with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_008	Verify that Password field does not allow only numbers and special symbols	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as numbers [0-9] along with special symbols with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_009	Verify that Password field allows combination of letters (Uppercase + lowercase), numbers and special symbols	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		

		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as combination of: 1. Uppercase letters 2. Lowercase letters 3. Numbers [0-9] 4. Special Symbols with greater than or equals to 8 characters and click on Submit button	New user should get registered successfully and should get displayed with a unique candidate ID	Password is accepted and user registration is successful	Pass
TC_010	Verify that Password field does not allow combination of lowercase letters, numbers and special symbols	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as a combination of lowercase alphabets, numbers [0-9] along with special symbols with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_011	Verify that Password field does not allow combination of uppercase letters, numbers and special symbols	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get		

		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as a combination of uppercase alphabets, numbers [0-9] along with special symbols with greater than or equals to 8 characters and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_012	Verify that Password field does not allow any combination of letters (uppercase and lowercase), numbers [0-9] and special characters if the password length is less than 8 characters.	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter values in Password, Confirm Password fields as any combination of uppercase, lowercase alphabets, numbers [0-9] and special symbols with less than 8 characters in length and click on Submit button	Validation error should get displayed as "Password is invalid. Should contain at least 1 uppercase letter, one lower case letter, 1 number and one special symbol and greater than or equals to 8 characters"	Validation error gets displayed as expected	Pass
TC_013	Verify that appropriate validation error gets displayed when the values in Password and	Step 1	Launch the application of "Book Your Doctor System"	Home page should get displayed		

	Confirm Password fields don't match					
		Step 2	Navigate to New User Registration UI	User Registration UI should get displayed		
		Step 3	Enter valid values in all the mandatory fields and enter different values in Password, Confirm Password fields as any combination of uppercase, lowercase alphabets, numbers [0-9] and special symbols with greater than or equal to 8 characters length and click on Submit button	Validation error should get displayed as "Values of Password and Confirm Password must match"	Validation error gets displayed as expected	Pass

Chapter 11: Database Design View

Android Platform: OverView:

```
bookurdoctor-491ec
  - Booking
    +-- Dec 20, 2019 9:54:18 PM
    +-- Jan 17, 2020 10:20:17 PM
    +-- Jan 17, 2020 12:30:35 PM
  - Doctors
    +-- Jan 16, 2020 11:09:21 PM
    +-- Jan 17, 2020 12:00:34 PM
    +-- Jan 17, 2020 12:13:18 PM
    +-- Jan 17, 2020 12:15:57 PM
    +-- Jan 17, 2020 12:18:39 PM
    +-- Jan 17, 2020 12:21:28 PM
    +-- Jan 17, 2020 12:23:00 PM
    +-- Jan 17, 2020 12:25:01 PM
    +-- Jan 17, 2020 12:26:28 PM
    +-- Jan 17, 2020 12:28:04 PM
  - Users
    +-- 8594946388
```

Users:

```
- Users
  - 8594946388
    address: "APJ CET BOYS HOSTEL, Bhubaneswa
    image: "https://firebasestorage.googleapis.com/v0/b/bookurdoctor-491ec/a...
    name: "Amiya"
    password: "123456"
    phone: "8594946388"
    phoneOrder: "8594946388"
```

Doctors:

```
[-] Doctors
  [-] Jan 16, 2020 11:09:21 PM
    ... days: "Tuesday, Wednesday"
    ... description: "Maa kali pharmacy near ram mandir Sahid nagar ,"
    ... doctorImage: "https://firebasestorage.googleapis.com/v0/b/boc..."
    ... doctorName: "Dr. Manas Jena"
    ... gender: "Male"
    ... locations: "A renowned doctor having 7 years of experience"
    ... price: "500"
    ... specialization: "Dentist"
  [+]-] Jan 17, 2020 12:00:34 PM
  [+]-] Jan 17, 2020 12:13:18 PM
  [+]-] Jan 17, 2020 12:15:57 PM
  [+]-] Jan 17, 2020 12:18:39 PM
  [+]-] Jan 17, 2020 12:21:28 PM
  [+]-] Jan 17, 2020 12:23:00 PM
  [-] Jan 17 2020 12:25:01 PM
```

Bookings:

```
- Booking
  - Dec 20, 2019 9:54:18 PM
    appointmentDate: "27-12-2019"
    bookingDate: "Dec 20, 2019"
    bookingTime: "21:54:18"
    currentOnlineUser: "9090399438"
    patientAddress: "Keonjhar, 768900, Kemsapada, near town police stat"
    patientAge: "52"
    patientGender: "Male"
    patientName: "Nuranjan"
    patientPhoneNo: "563214578"
    registeredDoctorName: "Sagar Pradhar"
    slot: "Noon"
  + Jan 17, 2020 10:20:17 PM
  + Jan 17, 2020 12:30:35 PM
```

Web Platform:

The screenshot shows the MySQL Workbench interface for the 'demo' database. The top menu bar includes tabs for Structure, SQL, Search, Query, Import, Operations, Privileges, Routines, Events, Triggers, and Designer. A 'Filters' section allows searching for specific words. Below this is a table listing four tables: bookdoc, doctor, ratingapp, and user. The table details include Rows, Type, Collation, Size, and Overhead. A 'Create table' dialog is open, showing a 'Name:' field and a 'Number of columns:' field set to 4. A 'Go' button is present at the bottom of the dialog. At the bottom left, there is a 'Console' tab.

The screenshot shows the MySQL Workbench interface for the 'user' table within the 'demo' database. The top menu bar includes tabs for Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, and Triggers. The 'Table structure' tab is selected, displaying the table's columns: u_id, u_email, u_mob, u_pass, u_img, u_fullname, u_age, u_gender, and u_address. Each column has its name, type, collation, attributes, nullability, default value, comments, and extra information like AUTO_INCREMENT. Action buttons for Change, Drop, Primary, Unique, Index, and Fulltext are provided for each column. Below the table structure, there is a 'Print' button and options to Propose table structure, Move columns, and Normalize. A 'Add' button allows adding new columns. The 'Indexes' section shows one index named 'PRIMARY' with type BTREE, unique Yes, packed No, column u_id, cardinality 5, and collation A. A note says 'No partitioning defined!'. At the bottom left, there is a 'Console' tab.

Server: 127.0.0.1 » Database: demo > Table: doctor

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Operations](#) [Triggers](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	drregid	int(11)	utf8_bin		No	None		AUTO_INCREMENT	Change Drop More
2	dremail	varchar(234)	utf8_bin		No	None			Change Drop More
3	drpassword	varchar(12)	utf8_bin		No	None			Change Drop More
4	drmob	bigint(20)			No	None			Change Drop More
5	drimg	text	utf8_bin		No	None			Change Drop More
6	drfullname	varchar(234)	utf8_bin		No	None			Change Drop More
7	drsp	varchar(234)	utf8_bin		No	None			Change Drop More
8	drgender	varchar(20)	utf8_bin		No	None			Change Drop More
9	abtldr	text	utf8_bin		No	None			Change Drop More
10	drday	varchar(20)	utf8_bin		No	None			Change Drop More
11	drfee	int(11)			No	None			Change Drop More
12	location	text	utf8_bin		No	None			Change Drop More

[Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) column(s) after location [Go](#)

[Indexes](#) [Edit](#) [Drop](#) PRIMARY BTREE Yes No drregid 3 A No

Create an index on columns [Go](#)

[Partitions](#) [Console](#)

Server: 127.0.0.1 » Database: demo > Table bookdoc

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Operations](#) [Triggers](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	bookid	int(11)	utf8_bin		No	None		AUTO_INCREMENT	Change Drop More
2	usermob	bigint(20)			No	None			Change Drop More
3	uname	varchar(234)	utf8_bin		No	None			Change Drop More
4	usage	tinyint(4)			No	None			Change Drop More
5	ugender	varchar(20)	utf8_bin		No	None			Change Drop More
6	uphone	bigint(20)			No	None			Change Drop More
7	uaddress	text	utf8_bin		No	None			Change Drop More
8	ucity	varchar(234)	utf8_bin		No	None			Change Drop More
9	bookdate	varchar(15)	utf8_bin		No	None			Change Drop More
10	bookslot	varchar(15)	utf8_bin		No	None			Change Drop More
11	drid	int(11)			No	None			Change Drop More
12	date	date			No	current_timestamp()			Change Drop More
13	time	time			No	current_timestamp()			Change Drop More

[Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) column(s) after time [Go](#)

[Indexes](#) [Edit](#) [Drop](#) PRIMARY BTREE Yes No bookid 6 A No

Create an index on columns [Go](#)

Server: 127.0.0.1 » Database: demo > Table: ratingapp

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	ratevalue	double			No	None			Change Drop More
3	doc_id	varchar(11)	utf8_bin		No	None			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext

Print Propose table structure Move columns Normalize

Add 1 column(s) after doc_id Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	8	A	No	

Create an index on 1 columns Go

Partitions

No partitioning defined!

Partition table

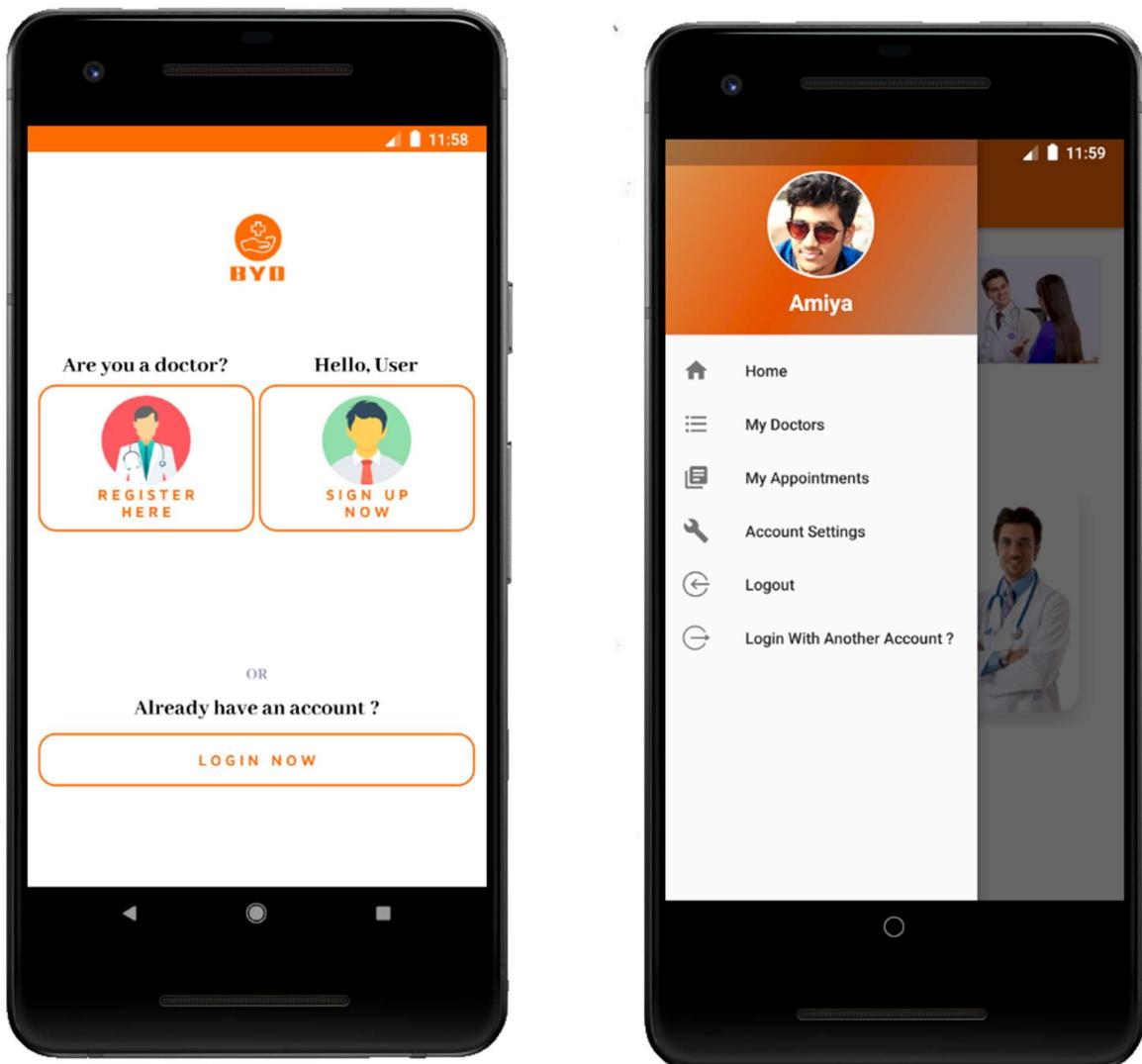
Information

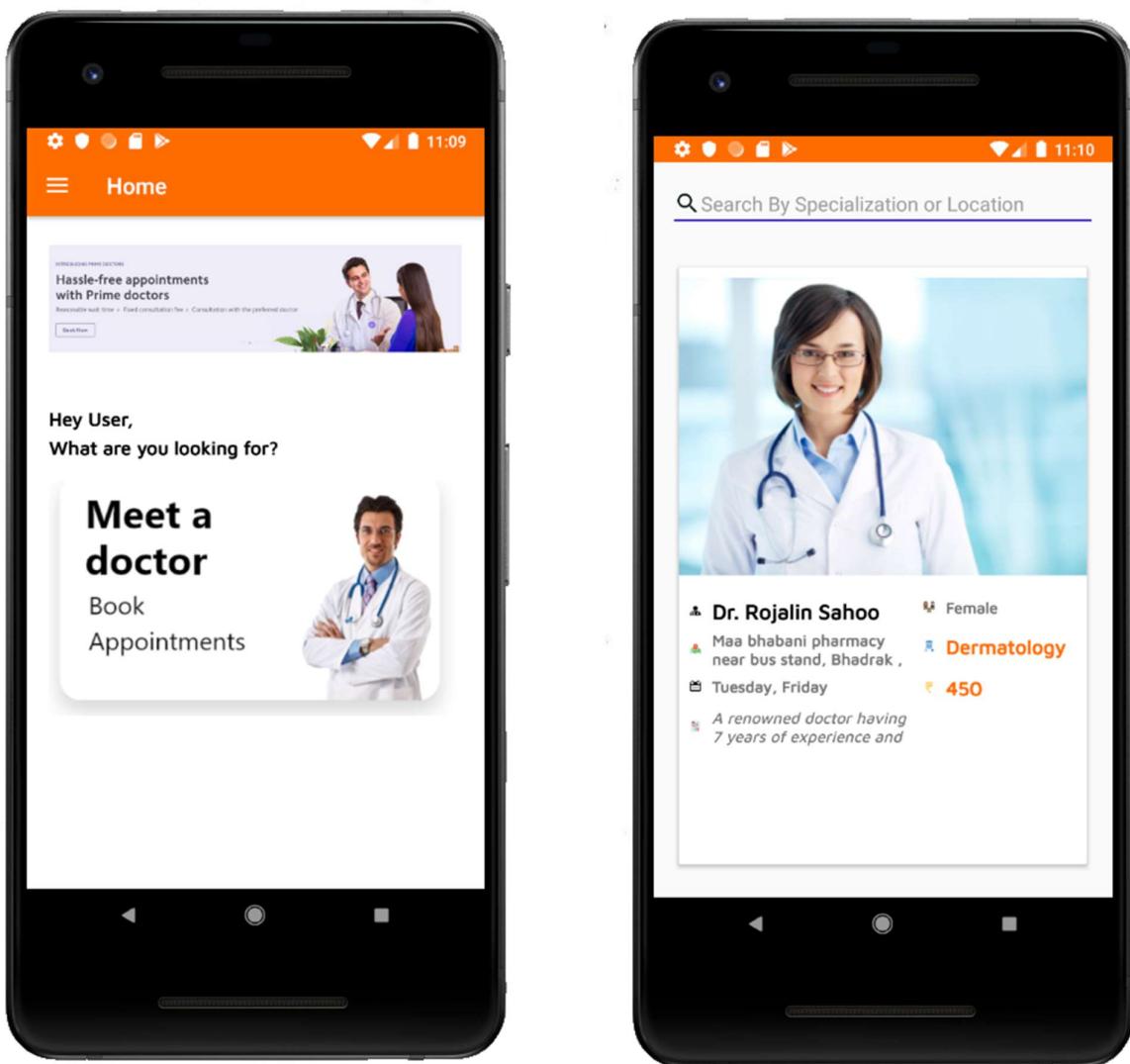
Space usage Row statistics

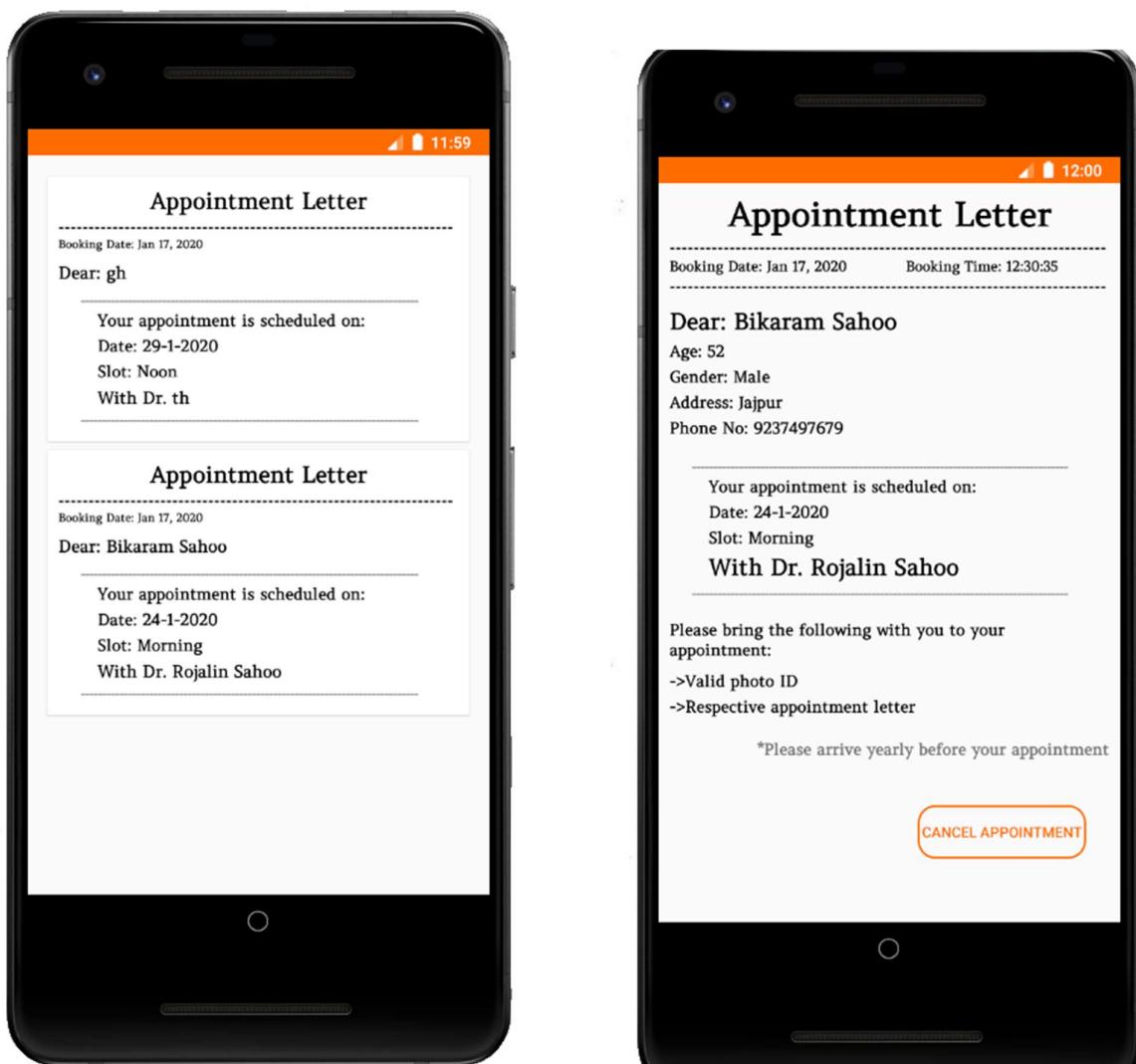
Console 16 KiB Format dynamic

Chapter 12: Output Screens

Android Platform:







Web Platform:

The screenshot shows the homepage of the BookUrDoctor website. At the top, there is a black header bar with the logo 'BYD' (BookUrDoctor) featuring a hand holding a stethoscope icon. Below the logo, the text '•BookUrDoctor.' is displayed in orange. The header also includes navigation links for 'HOME', 'DOCTORS', 'SEARCH', and 'Login'. Below the header is a search bar with fields for 'Location' and 'Search Doctors', along with a magnifying glass icon. The main content area features a large image of a doctor in a white coat and stethoscope talking to a patient. Above the image, the text 'INTRODUCING PRIME DOCTORS' is written in small capital letters. Below the image, the headline 'Hassle-free appointments with Prime doctors' is displayed in bold. Underneath the headline, a subtext reads 'Reasonable wait time • Fixed consultation fee • Consultation with the preferred doctor'. A 'Book Now' button is located at the bottom left of this section.

Find Doctor in top specialities



DENTIST

Oral issues?
Schedule a dental Checkup



GYNECOLOGIST/OBSTETRICIAN

Explore for women's health,
pregnancy and infertility treatment



DIETITIAN/NUTRITION

Get guidance on eating right,
weight management and sports
nutrition



PHYSIOTHERAPIST

Pulled a muscle?
Get it treated by physiotherapist





Email

Password

Remember me [Forgot password?](#)

Log In

or

Connect with Facebook

Connect with Gmail

A dark-themed login form with orange borders. It includes fields for email and password, a 'Remember me' checkbox, and links for forgot password and social media logins (Facebook and Gmail).



Satyaraj Bhanu

NAME	Satyaraj Bhanu
SPECIALIZATION	Physiotherapist
GENDER	male
CONTACT NUMBER	9516230847
BRIEF DESCRIPTION	A well established doctor
DAYS	Wednesday
FEES	300
LOCATION	Bhanu Clinic

UPDATE PROFILE **DELETE PROFILE**

A detailed profile page for a user named Satyaraj Bhanu. It features a circular profile picture, the user's name, and a table listing various profile details such as specialization, gender, contact number, brief description, days available, fees, and location. At the bottom, there are green and red buttons for updating or deleting the profile.



RATE HERE

★★★★★

Submit



Copyright © 2020, BookUrDoctor. All rights reserved.


Miss Pari

 Average: 4.38 ★

 Miss Pari

 pari@gmail

 12345

 Gynecology

 female

 Sunday, Friday

 ₹ 500

 Sambalpur

 A renowned doctor

RATING

BOOK NOW

HOME DOCTORS SEARCH LOG IN BookUrDoctor



about US

Your home for health

Smart access with BookUrDoctor app

- Easily add or modify your details

Chapter 13: Conclusion

13.1 LIMITATIONS AND SCOPE FOR FUTURE ENHANCEMENTS:

13.1.1 Limitations of the System:

- System only works in specific platforms and its compatible environments.
- Advanced techniques are not used to check the authorization.
- The system does not work on dynamic conditions.

13.1.2 Future Enhancements:

It is not possible to develop a system that makes all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this system are:

- As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment.
- Because it is based on object-oriented design, any further changes can be easily adaptable.
- Based on the future security issues, security can be improved using emerging technologies.
- Payment gateway can be added
- Prime membership can be added
- GPS tracking can be implemented in order to track particular doctor's location according to which the user book the appointment.

13.1.3 PROJECT SUMMARY

This application software has been computed successfully and was also tested successfully by taking "test cases". It is user friendly, and has required options, which can be utilized by the user to perform the desired operations.

The software is developed using php as front end and mysql as back end in Windows environment .

The android application is developed using xml as front end and firebase as back end in Android environment.

The goals that are achieved by the software are:

- ✓ Optimum utilization of resources.
- ✓ Efficient management of records.
- ✓ Simplification of the operations.
- ✓ Less processing time and getting required information.
- ✓ User friendly.
- ✓ Portable and flexible for further enhancement.

13.2 Work Done

The Book your Doctor System was successfully designed and is tested for accuracy and quality.

During this project we have accomplished all the objectives and this project meets the needs of the organization. The developed project will be used in searching, retrieving and generating information for the concerned requests.

GOALS

- ✓ Reduced time needed previously for paper work
- ✓ Easy retrieval of information
- ✓ User friendly screens to enter the data
- ✓ Portable and flexible for further enhancement
- ✓ Web enabled.
- ✓ Fast finding of information requested

Chapter 14: References/Bibliography

Android Platform:

<https://developers.google.com/android/>
<https://www.android.com/>
<https://www.youtube.com/>
<https://developer.android.com/studio/index.html>
<https://in.udacity.com/courses/android>

Web Platform:

<https://www.w3schools.com/>
<https://www.youtube.com/>
<https://in.udacity.com/courses/web>

THE END