Deliverable 2

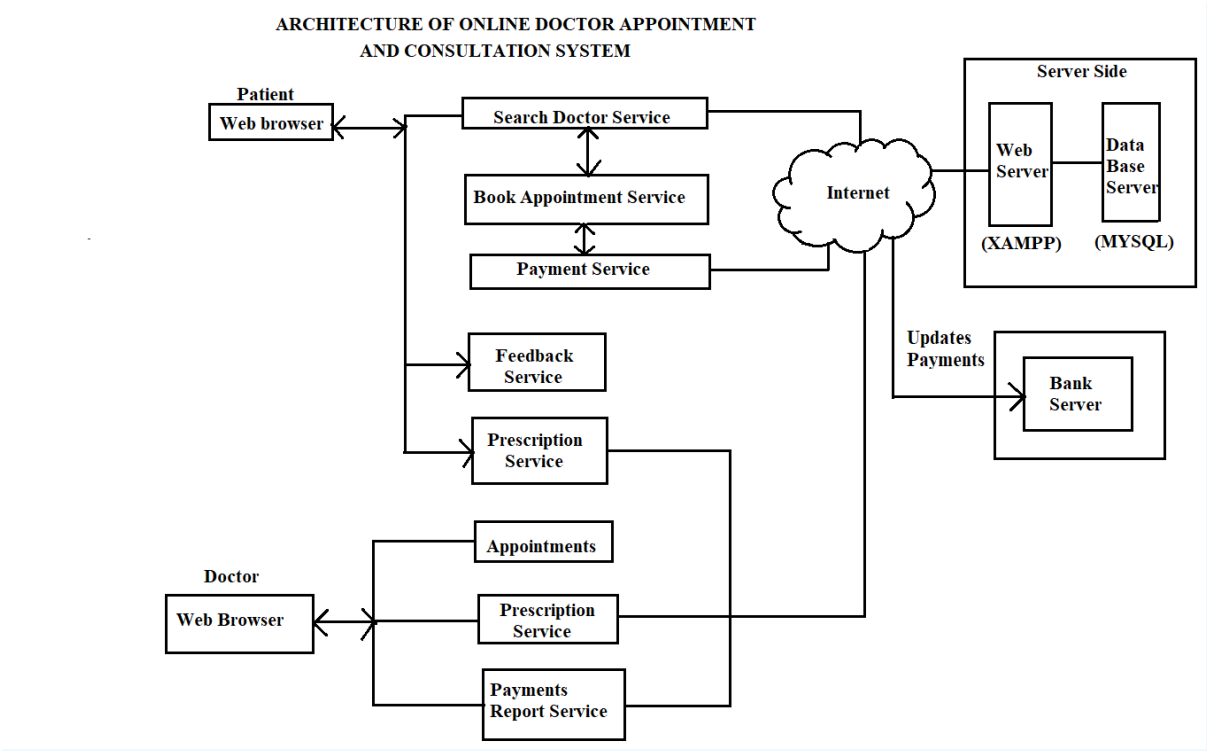
Project Title:

Online Doctor Appointment and Consultation System

Group Name:

SoftwareEngineering5

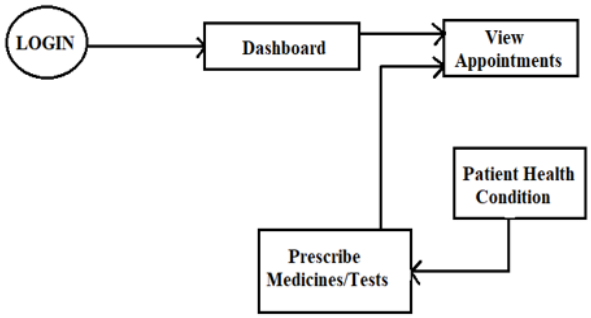
SYSTEM STRUCTURE: -

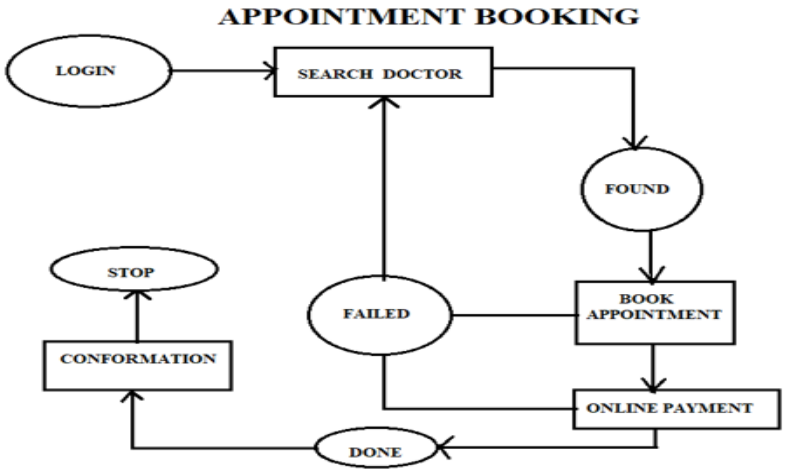


1. Patient sends request from web browse to searching a doctor, this request goes through internet and gives to web server. The server finds the proper webpage and sends back to browser.
2. Now the patient enters doctor details to search in the web page hit a button, this request goes to the web server, and which interacts with the database server for doctors' details. The web server gets the results from database server and returns to client.
3. After selected a doctor, the user clicks on another button to book an appointment, this request is again sent to server and server sends a payment form to client for payment.

4. The user after filling the payment form, he clicks on a button so that the request goes to web server where the booking details are stored in database and bank database is also updated.
5. The patient request for a feedback page to server and the server sends feedback page where the patient gives feedback to the doctor already consulted and hits a button which goes to server and server saves the feedback details in database.
6. The patient clicks on the prescription button on the dashboard which sends request to server to show all prescription data. The server sends prescription details in a table to client and the patient select a prescription to download.
7. The doctor on his dashboard clicks on the appointments button which sends request to web servers through internet, the web server get data from database of patients appoints of the doctor and the server send these details as a web page to client.
8. The doctor selects an appointment the appointments web page and post medicine prescription and hits on send button. This data is sent to web browser, and it saves on database.
9. The doctor clicks on view payments button on his dashboard which send request to server which communicates with database and get all payment data of this doctor and sends back to client.
10. Bank is a separate database where the payment details are updated that is data from patient account will be transferred to doctors account.

DOCTOR PRESCRIPTION





REQUIREMENTS SPECIFICATION: -

- Functional Requirements:
    1. A patient can register on the website for booking a doctor's appointment.
    2. The patient is provided to search for doctors on the dashboard.
    3. A patient can select a doctor and book an online appointment which allows paying the doctor's fee online.
    4. A patient can view the appointments history of their bookings.
    5. The patient is provided to give feedback on the doctor's performance.
    6. A patient can send medicine prescriptions to registered medical stores for online delivery.
    7. A patient can edit his profile.
    8. A doctor can create an account on the website with their qualifications, address, experience, certificates, etc.
    9. The administrator activates the doctor's account by accepting or rejecting based on the details provided on the website.
    10. A doctor can add his timings and fee details.
    11. A doctor can view all appointments on his dashboard.
    12. A doctor is allowed to post prescriptions of medicines to patients.
    13. A doctor can view all payment data made by patients.
    14. A doctor can view all feedback given by the patients.
    15. A doctor can be allowed to update his profile at any time.
    16. An administrator can view details of doctors, patients, and medical stores.

- Non- Functional Requirements:-
    1. The services in the project should run at a reasonable speed so that the users can browse the site faster.
    2. The interface design of the project should be clearer and easily understandable so that the users of the site can browse faster without any effort.

3. Users should be allowed to access the site with a strong password so cracking the password should be difficult and provides security.
4. The server should run 24X7 without interruption so that users can use the site at any time.
5. It should develop the application such that it can be used on any operating system on the server.
6. The system should show alert messages for any invalid operations performed by users.
7. The application should use web technologies that can support by all major web browsers.

Interfaces:

a. User Requirements:

1. Firstly, the user must register in the webpage by entering his name, email id, and phone number and create a password.
2. After the registration, user has to login using his/her credentials which navigates him/her to the login page.
3. Admin manages all these accounts.
4. After this Home page is displayed and where the user can see the search option, in which the user can either enter the symptoms or specialization or the doctor's name.
5. A list of doctor's names, their experience, specialization appears on the screen.
6. The patient needs to select the appropriate option needed by them.
7. After selecting the doctor, there is an option to select the offline or online consultation as well as time to book the appointment.
8. And also, the payment page appears, where they can select the mode of payment.
9. After scheduling the appointment, it time for admin to manage the user's request. Admin must map the patient to the doctor and confirm the appointment.
10. After completion of consultation, the user can navigate to the feedback page and rate the doctor accordingly. They can also request the doctor to upload the prescription on the portal or can ask to send it to their mails.

b. Hardware requirements include an Intel i7 processor or higher and RAM 8 GB or more HD 200 GB or more

c. Software prerequisites are Eclipse-based IDE, Java with MySQL, Front-end HTML, CSS, and JavaScript.

d. Server: Tomcat by Apache

e. Platform: Any web browser that is compatible, such as Firefox or Google Chrome.

f. communication Interfaces are HTTP protocol which is used for internet connectivity.

Development Phases:

Phase-I:

In this Phase, we are going to

- Design UML diagrams such as class diagrams and activity diagrams.
- Draw ER-Diagrams for the database and create a database and tables for the project.
- Design the Home Page.
- Design a doctor registration page to register on-site.
- Design a doctor login page for logins.
- Design a Patient registration page to register.
- Design a Patient login page for logins.
- Design Admin login page.
- Design logout code for all users.

Phase-II:

In Phase II, we are going to

- Design and Implementation of Patient's Dashboard.
- Design and implementation of Doctor's Dashboard.
- Create a search page to search for the Doctors.
- Create a View page to view the details of the doctor's search.
- Create an appointment booking page.
- Design the Payment Page.
- View/download medicine prescriptions.
- View the history page of the Patient's previous bookings.
- Creating a Feedback Page.
- View the appointments page from the doctor's point of view to view all patient appointments.
- Implementing all the designs of web pages using HTML, CSS and JS, and logic by JAVA.

Phase-III:

In this phase, we are going to

- Design and implementation of Admin Dashboard.
- Create an admin view for doctors and patients.
- Validation of the doctor registration page from the admin's point of view.
- Rechecking whether all the proposed points are implemented or not.
- Evaluating the website with a sample and real-time data.
- Fix the errors if identified or occurred.
- Finalizing the product.

TEAM MEMBERS CONTRIBUTION

S.NO	NAME	CONTRIBUTION
1	Manvitha Chowdary Bandi	Functional Requirements, Video
2	Meghana Pentyala	Non-Functional Requirements, Member Contribution, Documentation.
3	Anirudh Reddy Gade	Design Phases, video
4	Vagdevi Gudapati	Power Point Presentation, Description diagram
5	Sreeja Reddy Chilligireddy	Non-Functional Requirements, Meeting Minutes
6	Mukunda Priya Rachamalla	Description diagram, Power Point Presentation
7	Sai Vamsi Krishna Kadiyala	Functional Requirements, meeting minutes
8	Vaishnavi Gawni	Design Phases, Video