**Web App Architecture**

**Medical Appointment Booking Platform**

Applied Database Technologies (DSCI - D532)

by **Ameya Dattaram Parab, Trishna Patil**

We are following a MVC architecture for our web app. So the entire application is divided into 3 interconnected parts – the Model, the View and the Controller.

In the Model we store all the business logic and the data of the application. Data consistency and data integrity is managed here. We are using MySQL database server so this will act as the Model.

In the View we display all the data to the user so it is the user interface of the application where the data is presented in a very intuitive and understandable way. We are using the Shiny app which will act as the View.

The Controller receives input from the user, processes it and acts as an intermediary between the Model and the View. In our case the Shiny app will also act as a Controller.

So we have a Shiny app that will display the data of all the doctors and appointments. These would be displayed in the UI components and when the patients click to search or get a list of all the available doctors, the app would query the MySQL database to get the data of all the doctors and upcoming appointments. And then the results will be displayed in the UI component, completing the loop between the Model, View and the Controller. So we have a clear separation of concerns and the application will also be more scalable and maintainable.

1. Where your data is stored: cloud server (e.g. firebase), stand-along (e.g. SQlight)

Our data will be stored in a MySQL database. It is an open-source database and can support large volumes of data so our application can be more scalable. MySQL is also optimized for performance and is stable and reliable. It also provides several security features, including encryption, authentication, and access control.

1. What languages will be used to build back-end: python, R, javascript, java etc

We will be using R to build our back-end. using R for backend development can provide several advantages, including powerful statistical analysis, easy integration with data sources, rapid prototyping, cross-platform compatibility, cost-effectiveness, and easy visualization.

1. How you will be accessing the database: what connections and how secure (e.g. you have admin privileges and users cannot modify stored data etc

We are going to use the MySQL database connector to connect our database where we need to provide a username and password. Users are assigned different levels of privileges, which determine the types of operations they can perform on the database. Admins can perform all the CRUD operations on the database and have the right to modify the data. Users of the application do not have admin rights and can only read the data and book appointments through the application.

1. What will you use to create a front-end layout (HTML, CSS, js). Consider using Bootstrap (templates for design and layout) - [https://getbootstrap.com/Links to an external site.](https://getbootstrap.com/)

We are creating the front-end layout using the R Shiny themes. These are pre-designed stylesheets that we can apply to our shiny app to change it’s appearance. They internally use CSS and Bootstrap. We will also customize the themes by modifying the CSS code.

1. Where your application is deployed: shiny server, firebase, heroku, pythoneverywhere ...

We are going to deploy our application to the shiny server. To deploy a Shiny application to the Shiny Server, we will first prepare our application, make sure all the necessary packages are installed, testing it locally etc. Then we set up the server environment and move our application to the server. And after configuring the Shiny Server we will start it.

1. How you will provide interactivity for your app. Note - users should be able to click, select, view etc

We will have the app title displayed at the top of the screen. On the left of the screen there will be an input box where the user can click and enter text to search for any specific doctors. Just below there will be a tab to switch to view all doctors. We will also have filters to view and change plots and a notification bar where the patients will be able to see their upcoming appointments. We will also display the user information and there will be a clickable logout button at the top right. Once logged in, there will also be a main dashboard where the body will have table of doctor details such as name, address, degree, specialty with a button on right of each row for booking appointment (view/book appointment). After clicking on the button it will display all details of doctors including schoolname, services, phone number and an option to select time for booking appointment.

1. draw a schema with the web app architecture (see an example) - you can use ppt , draw io, and other tools to sketch - 20pts [add a screenshot to your word document]

Diagram

Description automatically generated