Final Report

1. Summary of our project

The topic name we have given for our project is - ‘Want a Doctor?’. We aim to make an application that will let the users book an appointment online according to their convenience. With changing times, the medical facilities have improved, people have become more conscious about their choices, therefore, people don’t visit hospitals just for emergencies or when something is wrong. They visit hospitals for monthly/annual check-ups too! It’s fair to say, there is a high demand of healthcare services- online application being one of them.

One of the motivations for us to choose this topic was our own past experiences of having to stand in long queues to wait for our turn to book an appointment. Booking through phone calls is even more time-consuming as the person who picks up our call would transfer it to another department, which would further transfer it to another department, and this would go on till someone connected us to our respective department. In this whole process of connecting from one-department-to-another, there would be instances where our call would disconnect, and we would have to start over and do all the things once again. Sometimes, we would not get an appointment with the doctor of our own choice, and even if we did, we would have to bend our tight schedule for that particular time period. Booking an appointment should not be that complicated. We aim to overcome these issues by building an application that allows users to create an account online, book an appointment with the doctors of their own choice, cancel it if something comes up, search for a doctor at a particular location, look up a doctor’s profile, let the users look at their booking status. The status will be sent to the user email when an appointment is booked or cancelled. Basically we build an ideal appointment application with basic functionalities for a user to book a doctor appointment online.

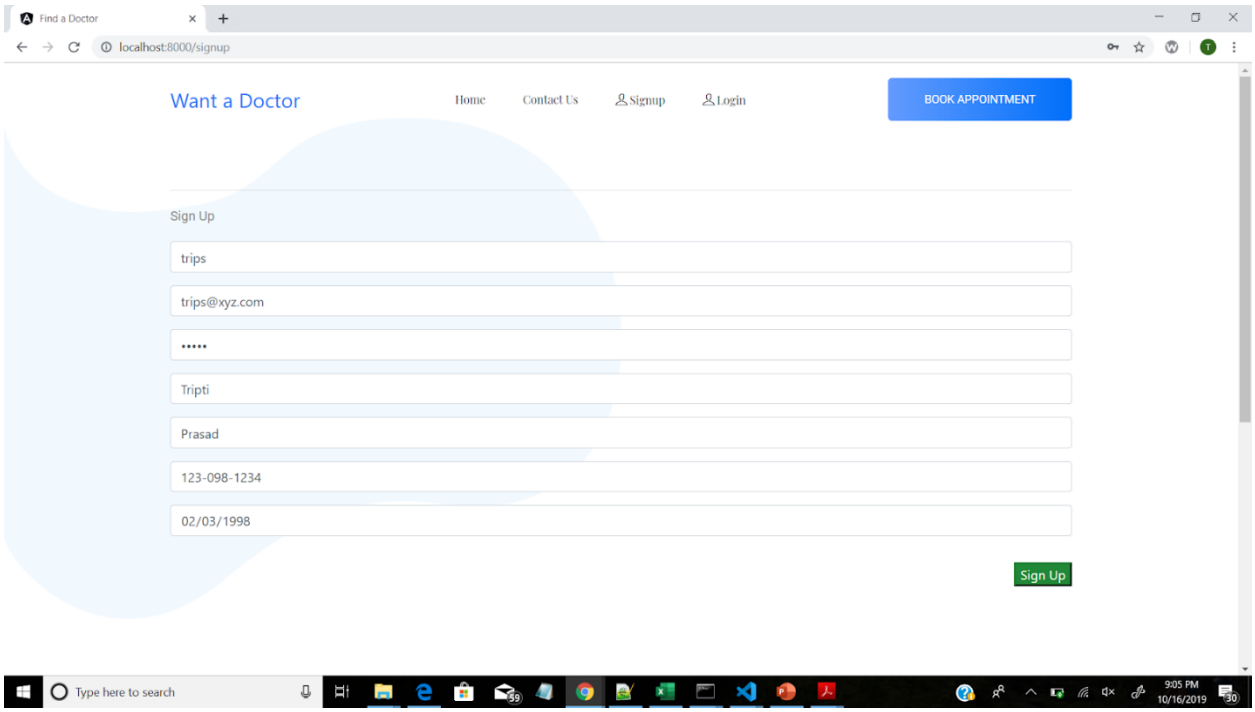
We did programming in Angular Js, Node JS, Loopback framework, prostgreSQL, HTML, Bootstrap.

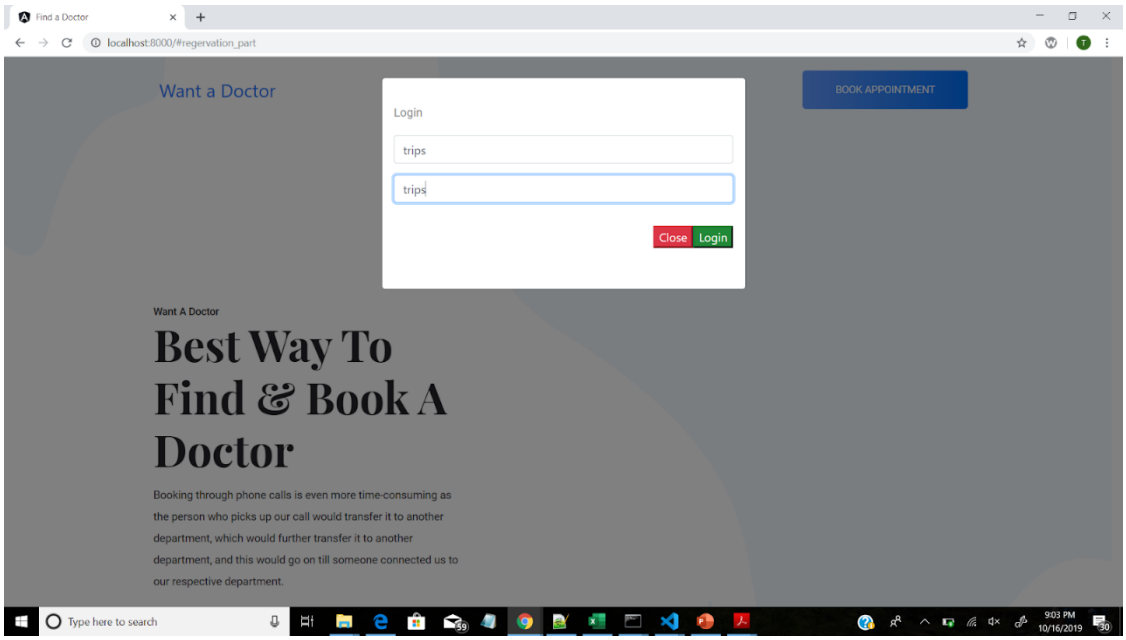
2. Layered Development Schedule

To achieve this, we have divided our project into 5 layers, with each layer defining the functionalities as per our requirements. Those requirements start from ‘adding minimum functionalities’ and ends at ‘adding extra features’.

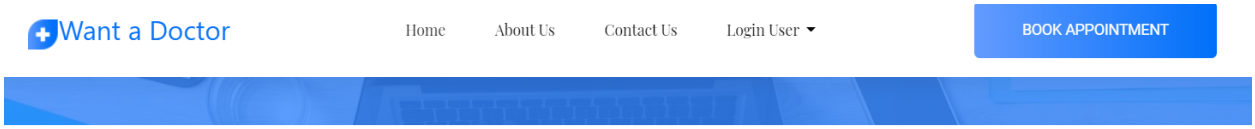
We broke our project into 5 layers described as below. We have mentioned the approximate week numbers of when we would like to deliver each layer.

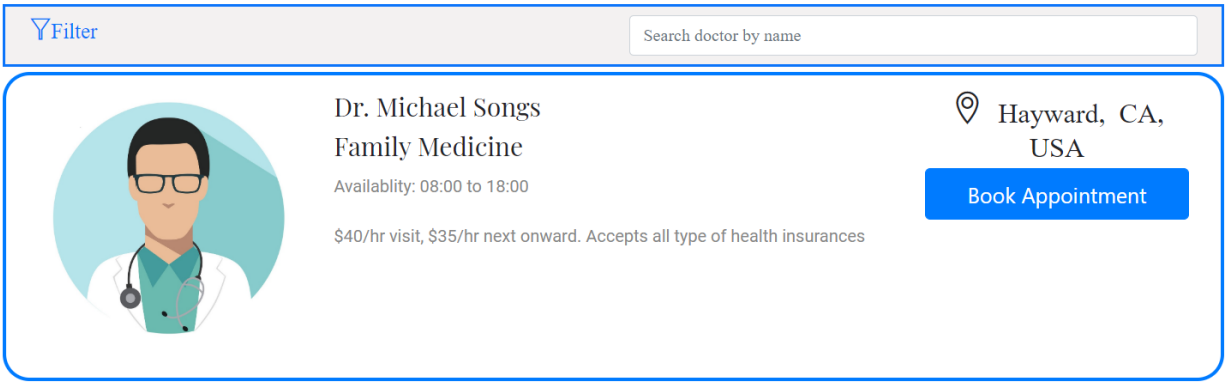
- Functional minimum:
  - login/register to the website



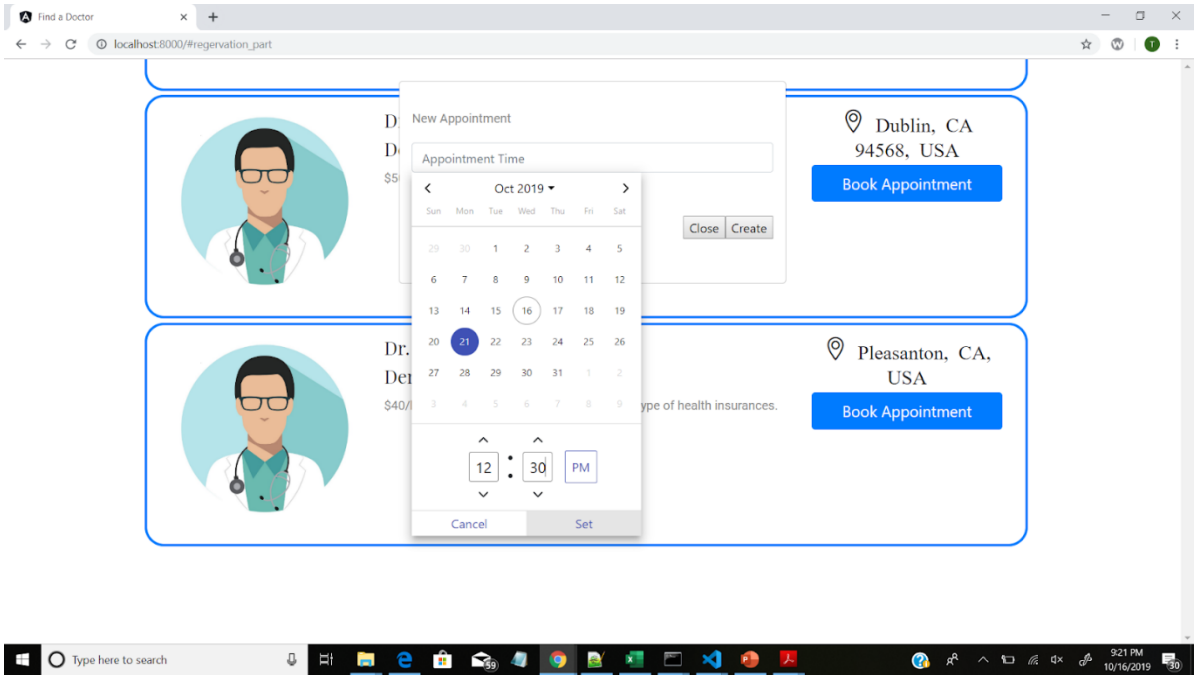


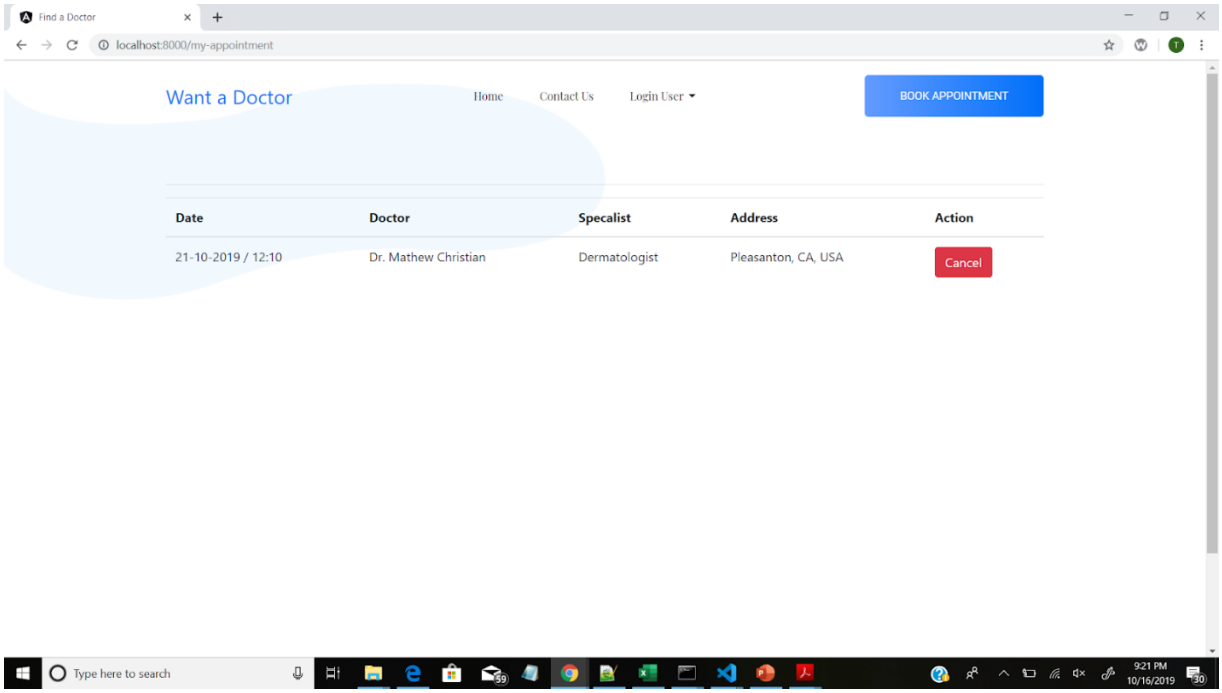
- search for a doctor in particular location





- look at doctor's profile
- online appointment booking/cancelling





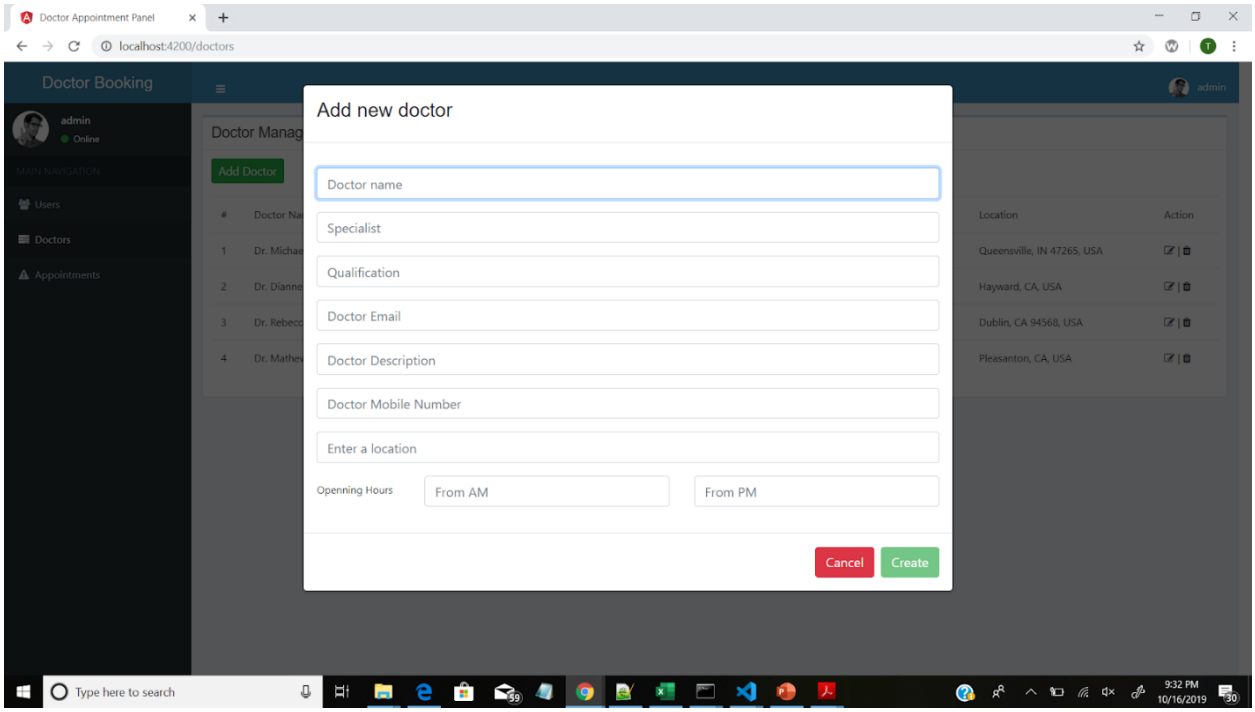
Due date: Week 7

- Your low target:
  - achieve the min functionalities
  - make it user friendly
  - quick response time
  - navigation should be easy

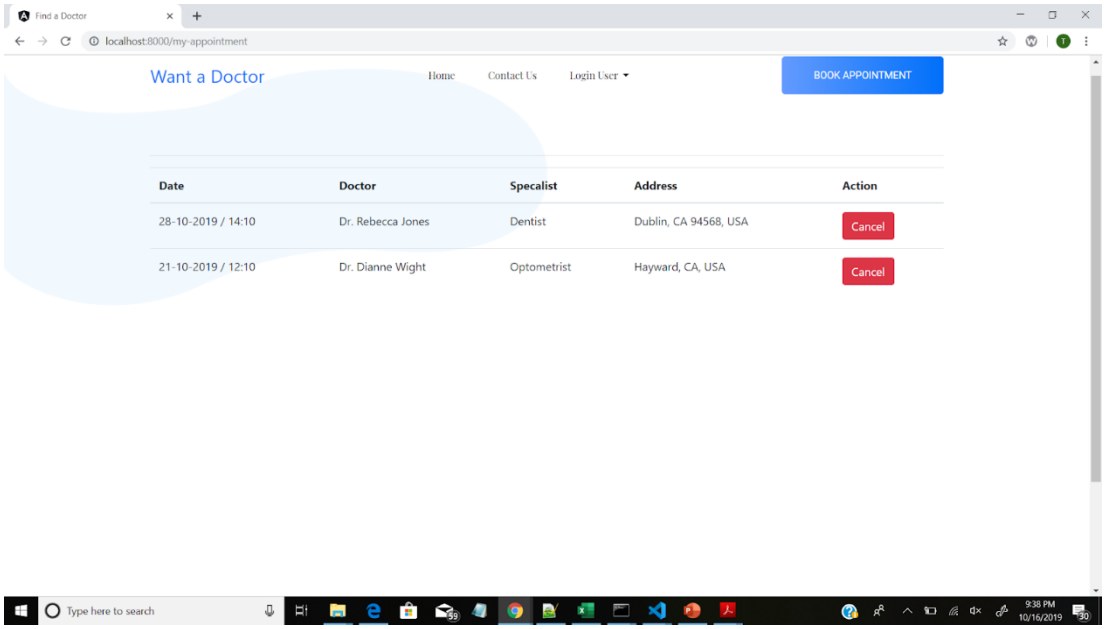
  Due date: Week 8

  The above stated layer 2 goals are non-functional requirements which have been met by the existing application. However, the stated goals will be considered in future layer implementations too.

- Your desirable target:
  - Add/Delete doctor by an admin

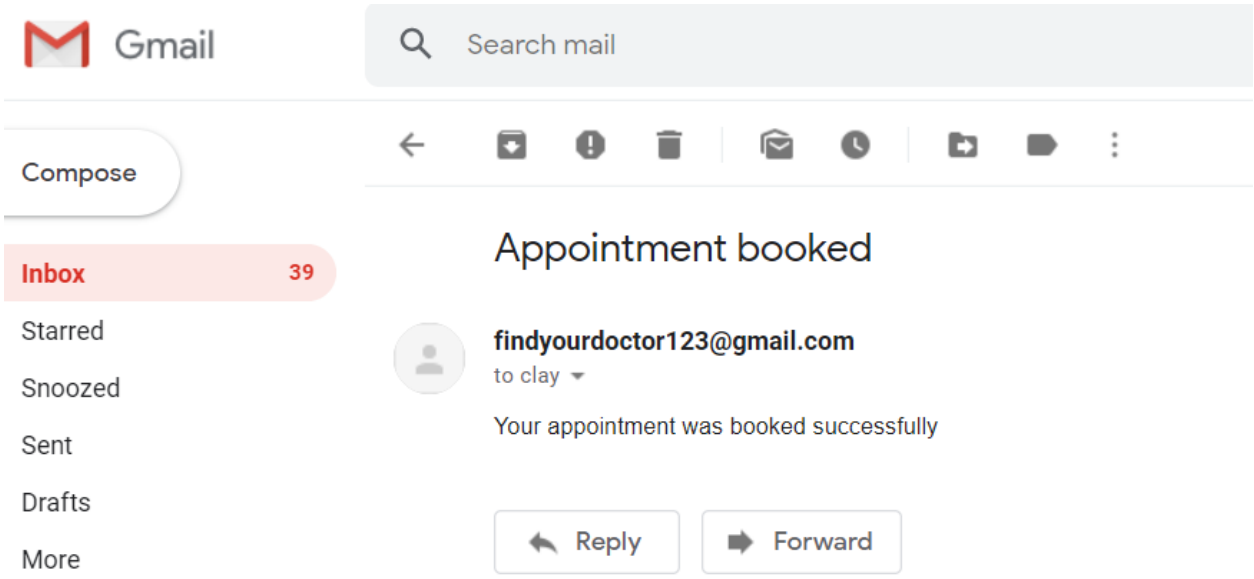


- User's can see their bookings.



Due date: Week 10

- Your high target:
  -email notification to the patients



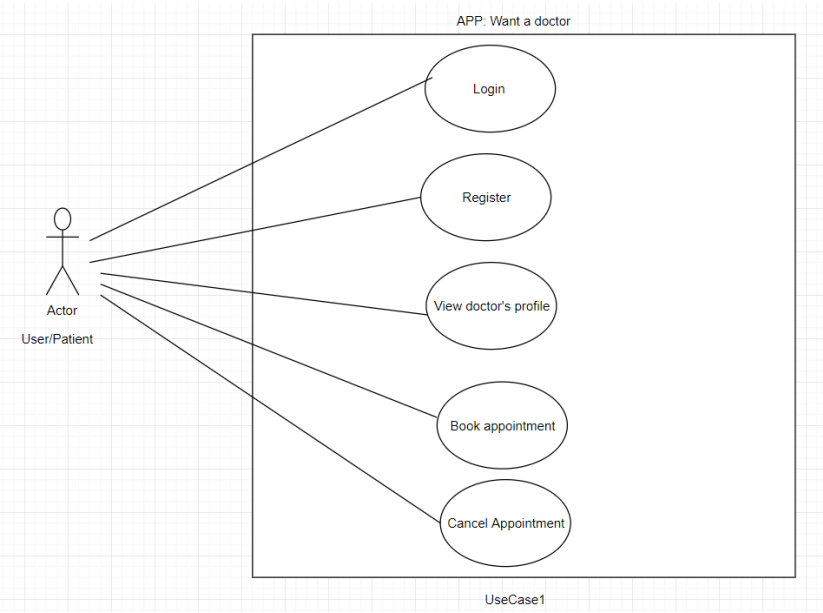
Due date: Week 13

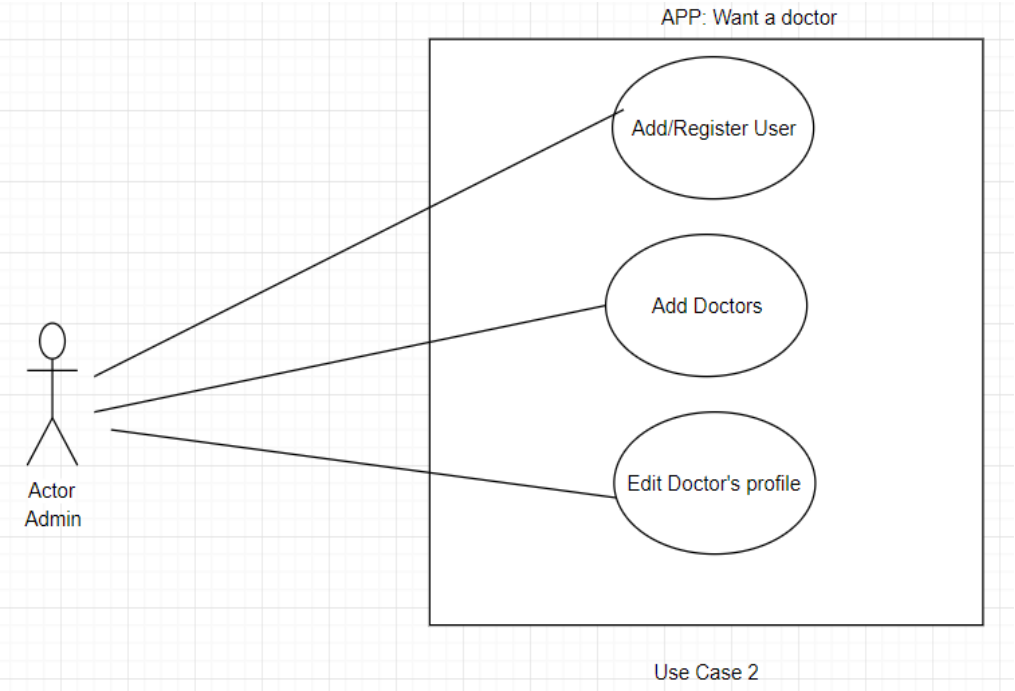
- Your extras:
  - payment module
  - rate your doctor

  Due date: NA

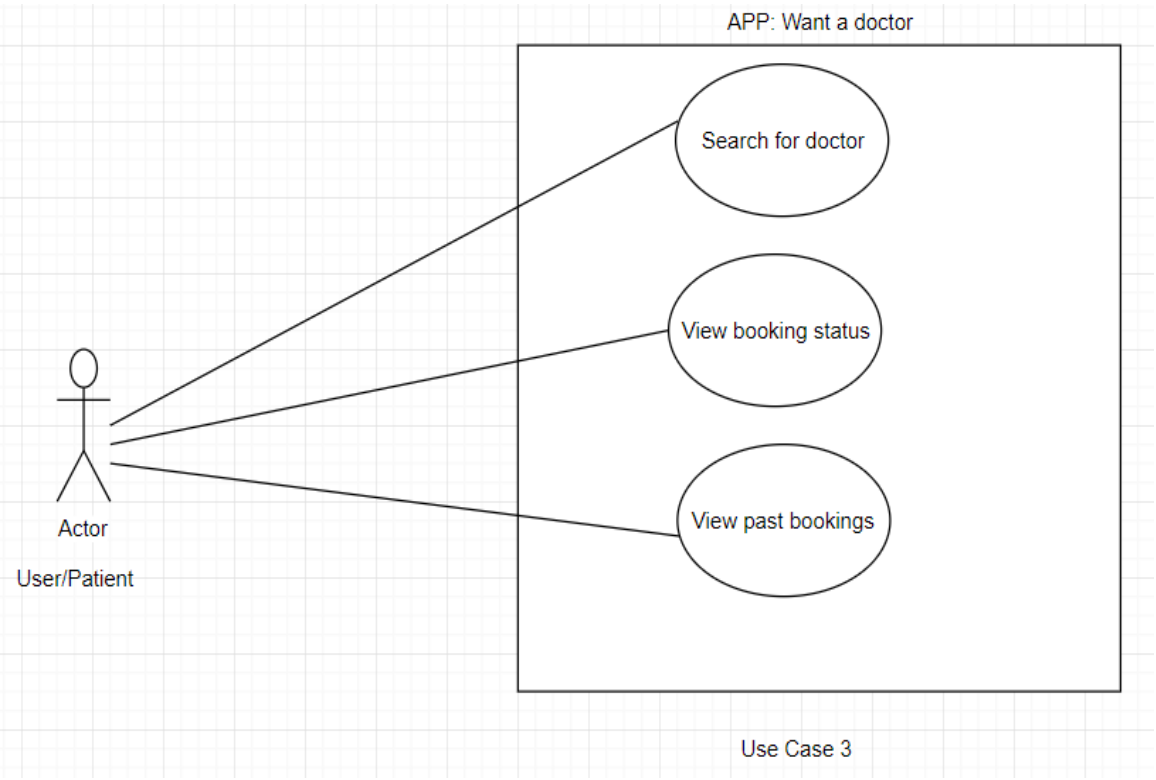
3. **Designs:**

Phase1 designs are shown below. We tried to incorporate the use cases together into one use case.

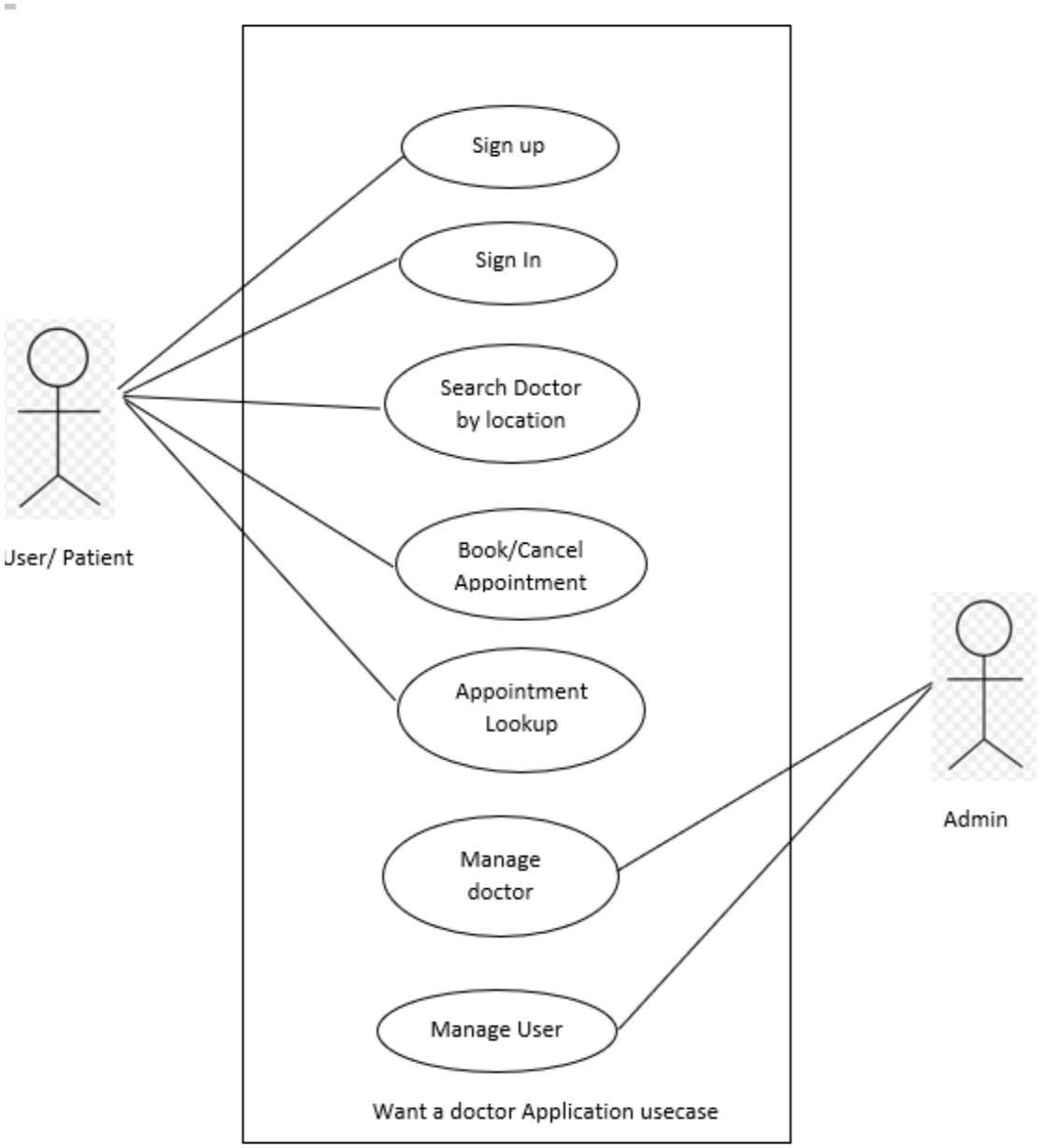




case2

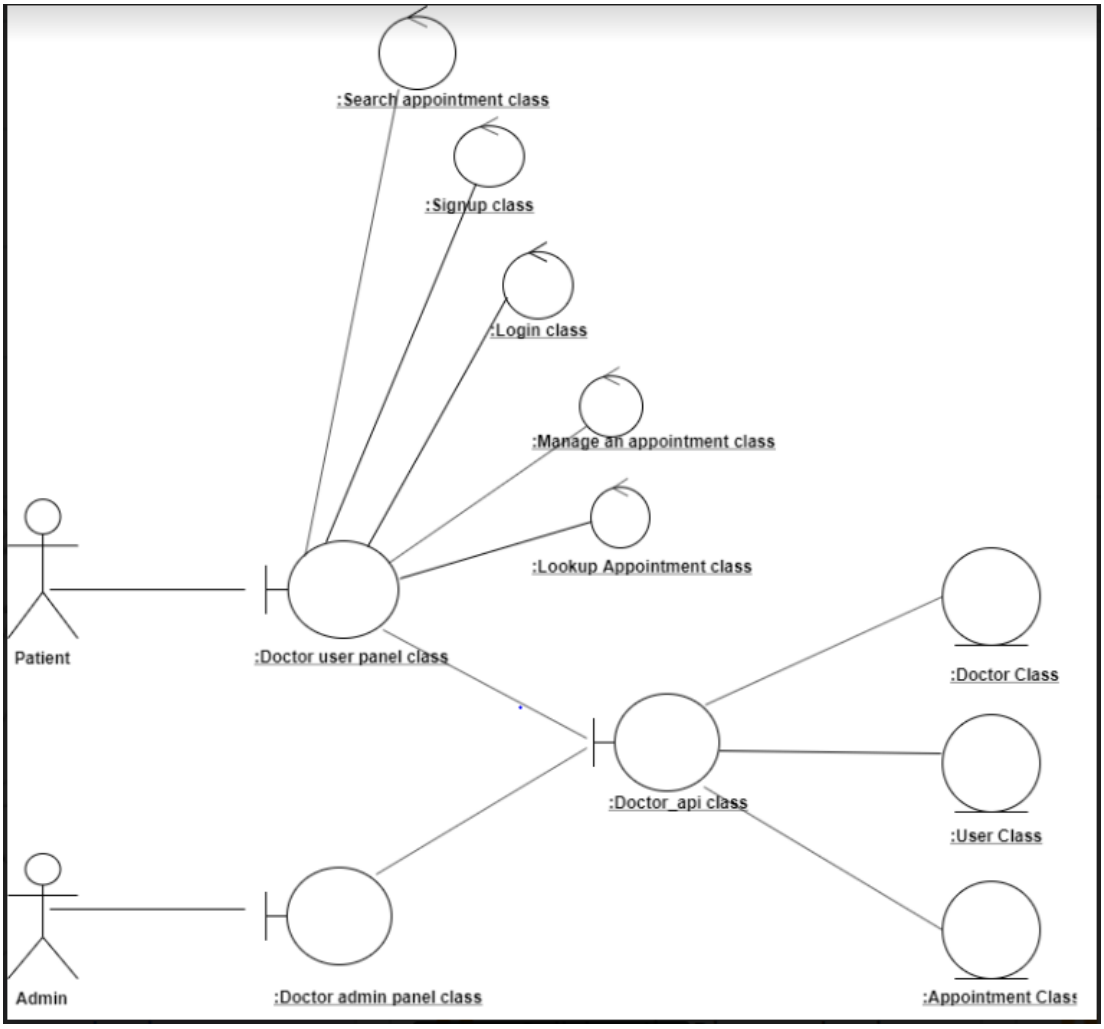


Phase1 usecase3

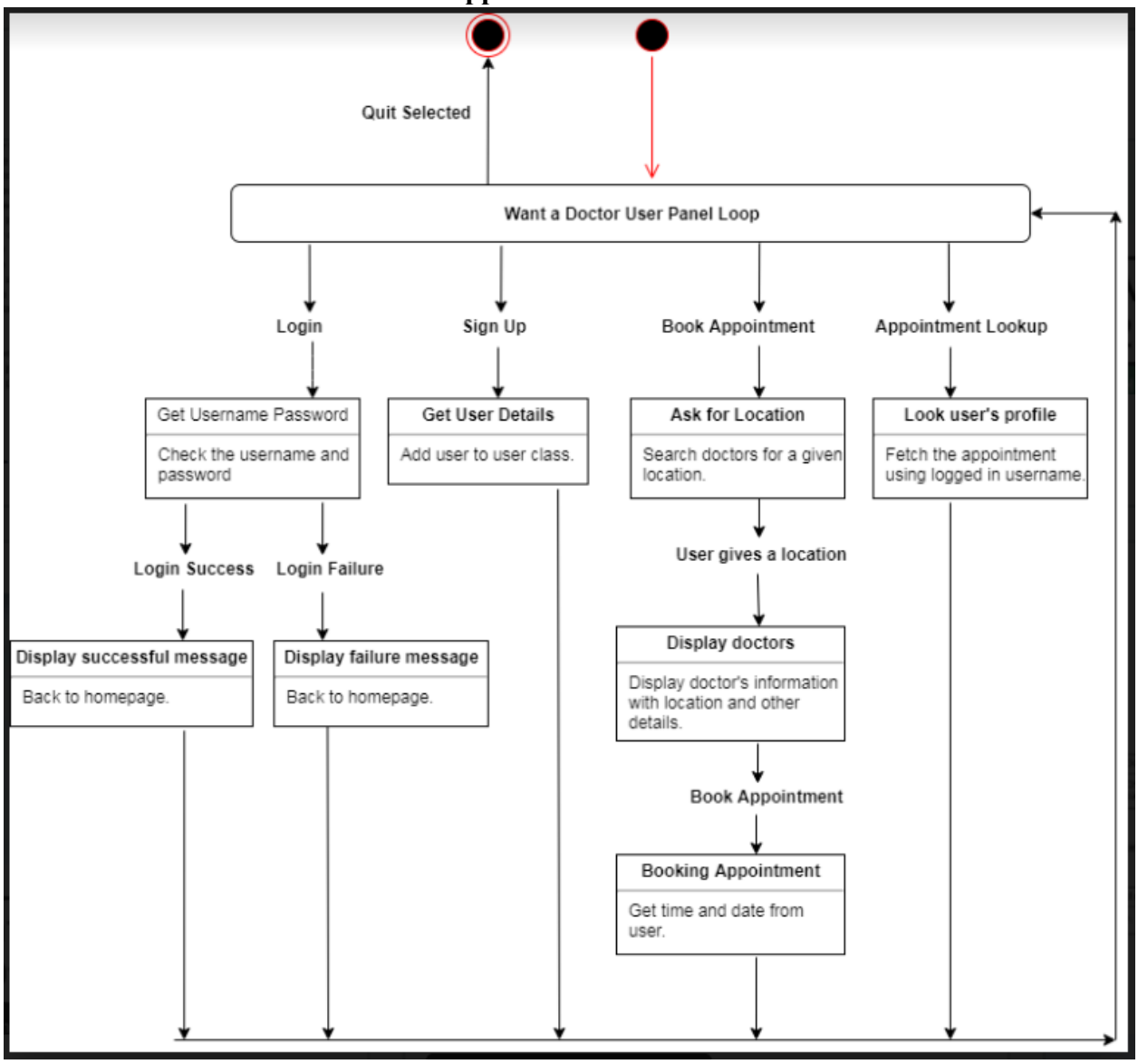


Phase2 Final Use-case

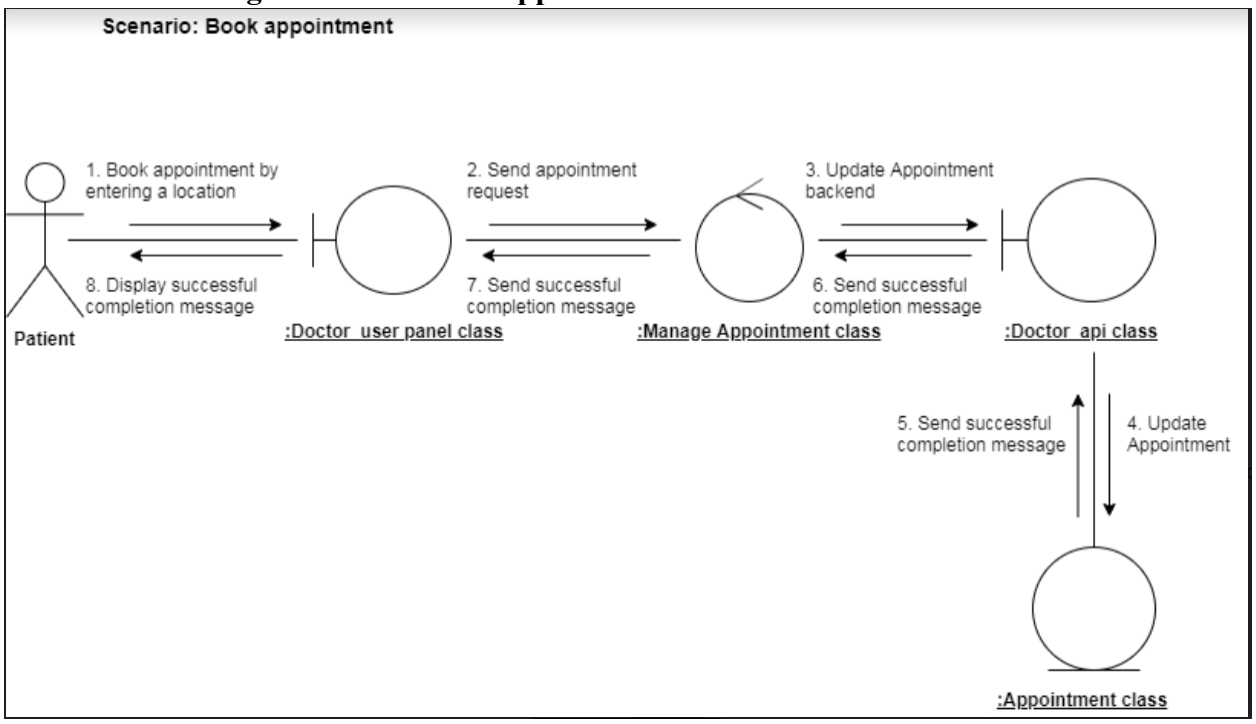
Class diagram for Want a Doctor showing the stereotypes:


:Search appointment class
:Signup class
:Login class
:Manage an appointment class
:Lookup Appointment class
Patient
:Doctor user panel class
:Doctor Class
:Doctor_api class
:User Class
Admin
:Doctor admin panel class
:Appointment Class

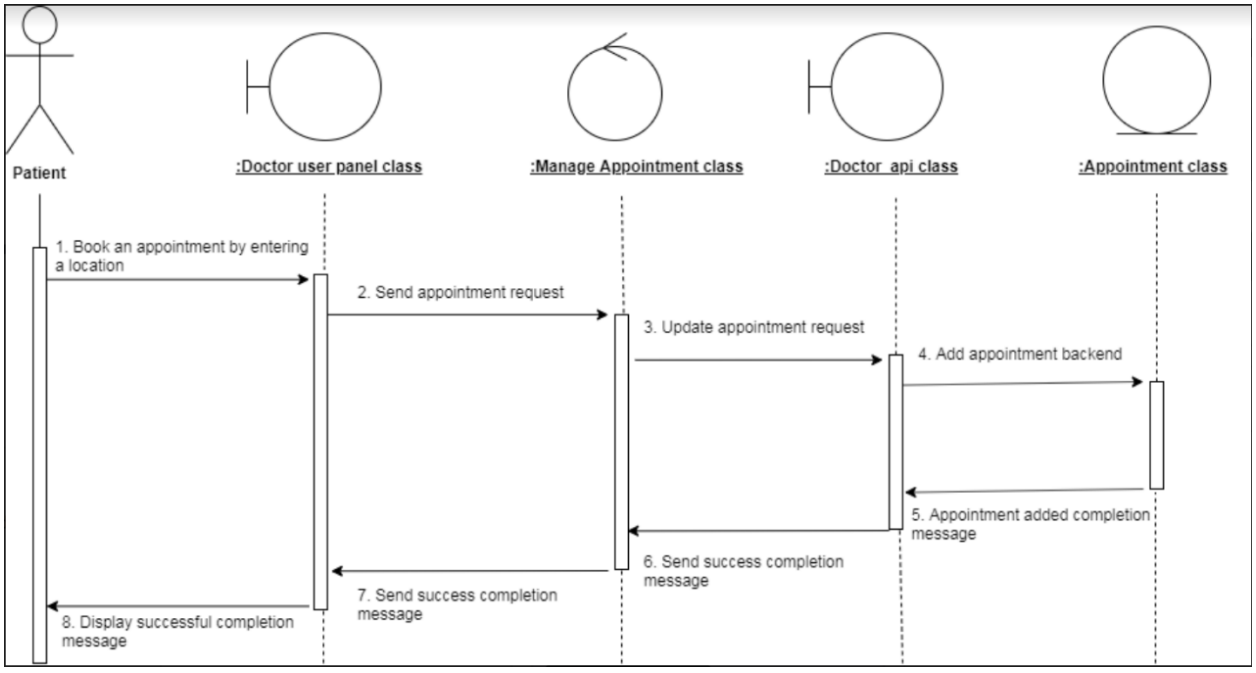
Initial State chart of Want a Doctor App:



Collaboration Diagram for Book an Appointment Scenario:

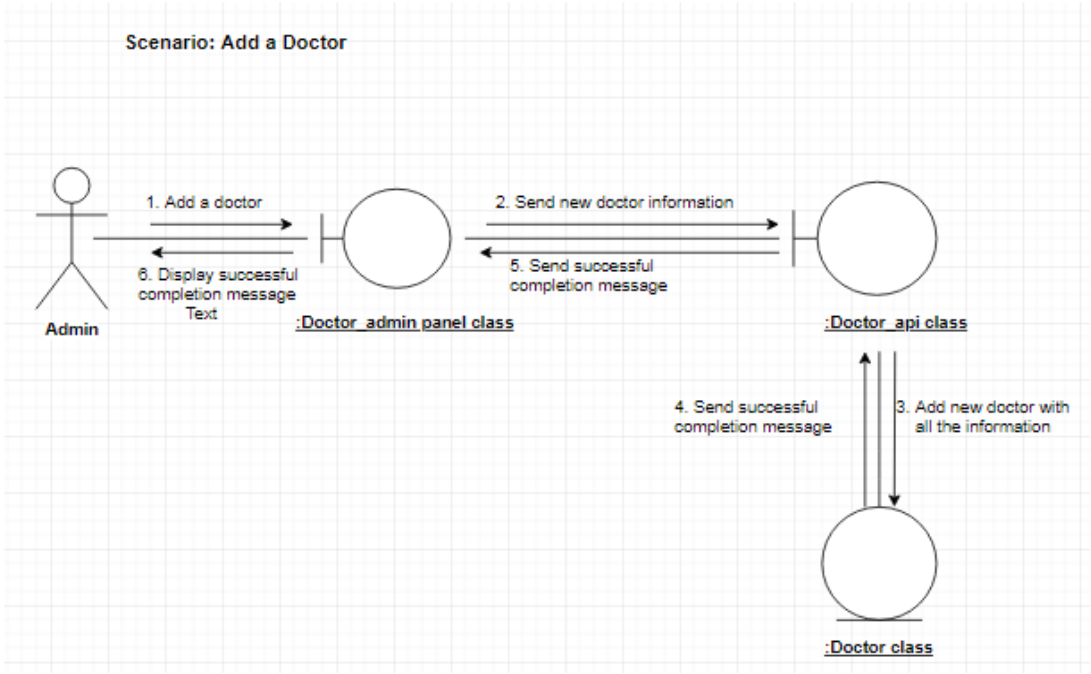


Sequence diagram for the above stated scenario:

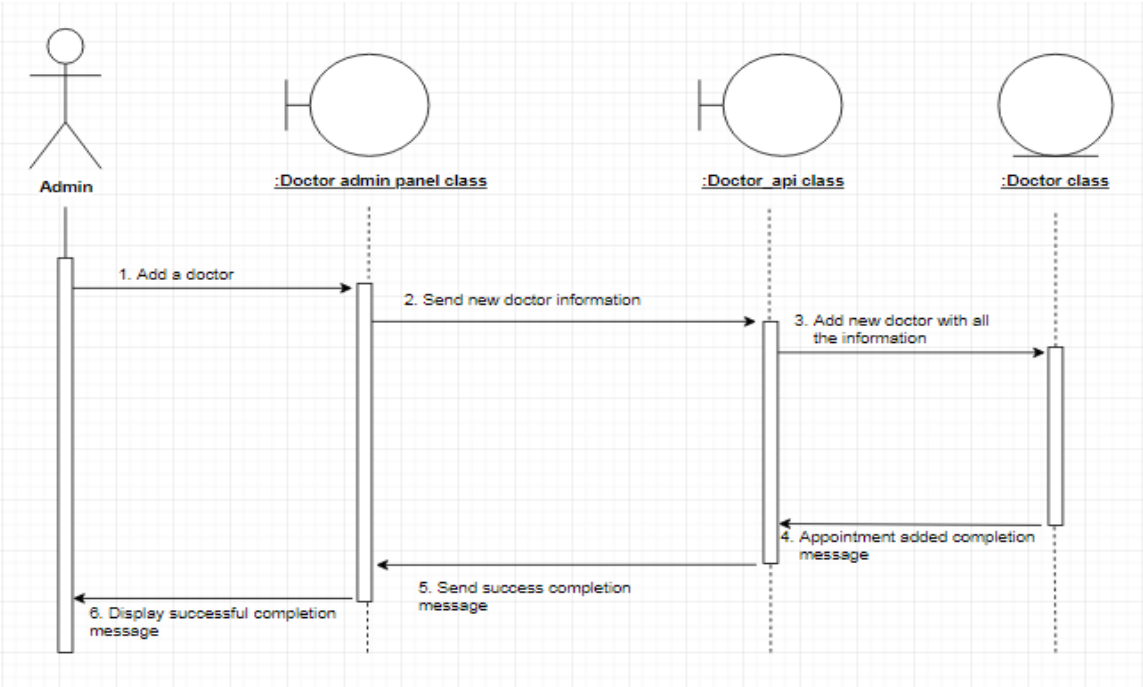


Scenario 1: Add a Doctor

Collaboration diagram

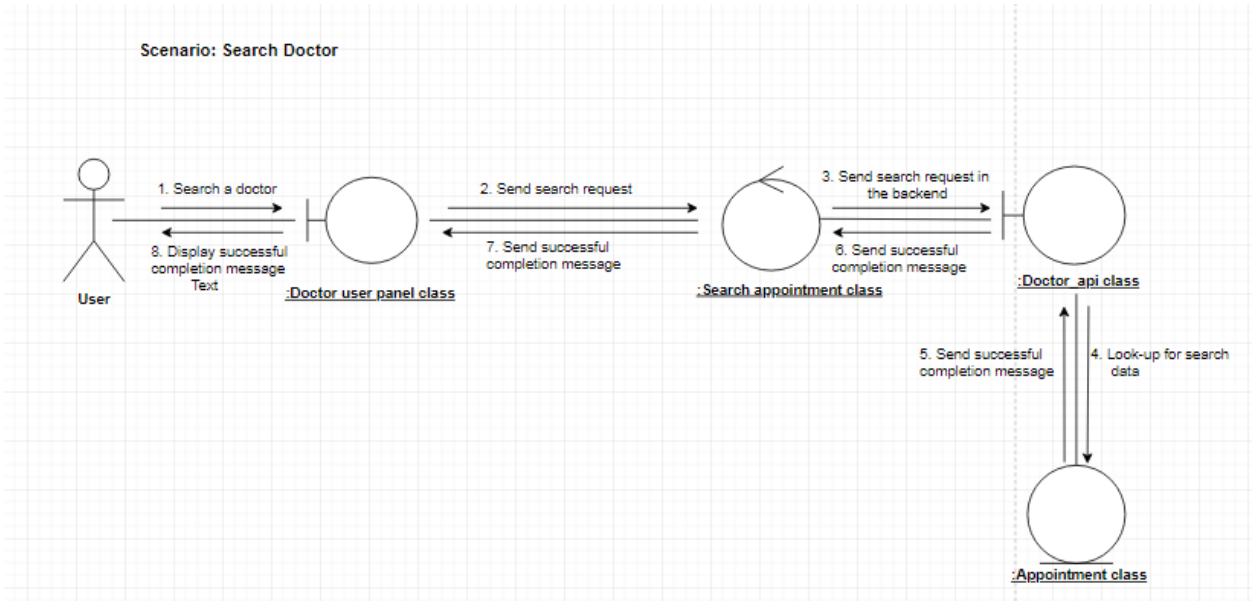


Sequence diagram

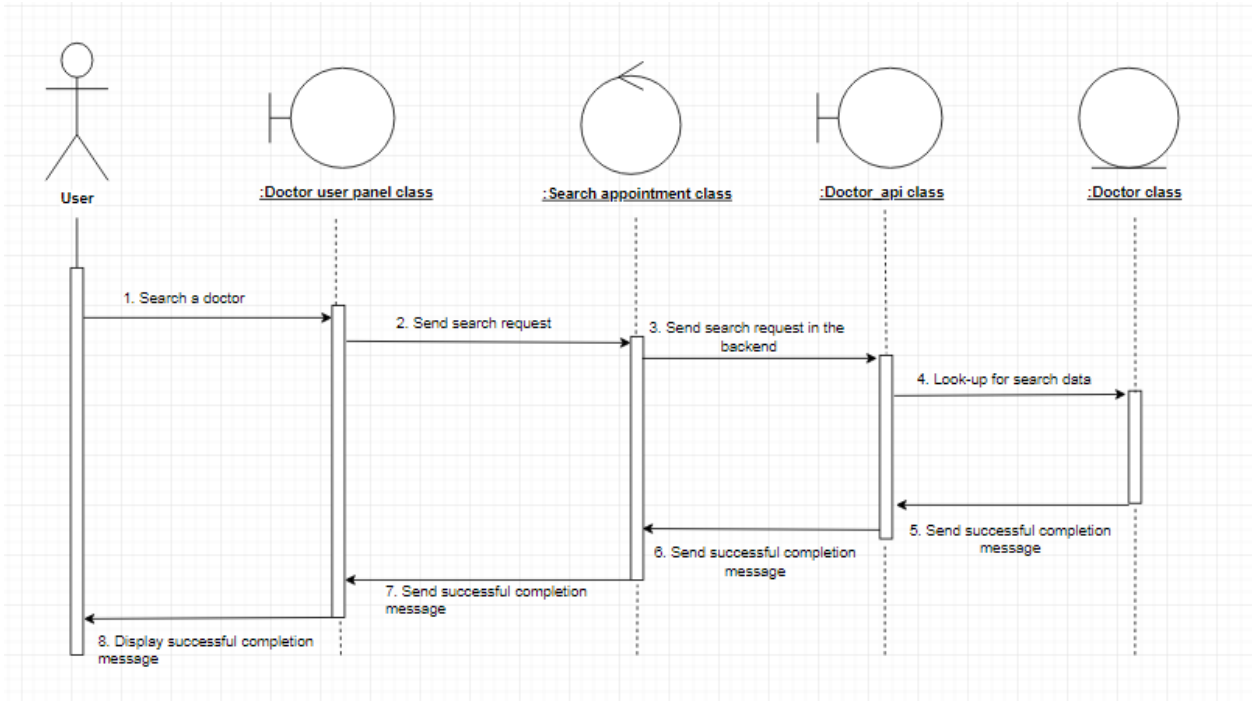


Scenario 2: Search Doctor

Collaboration diagram

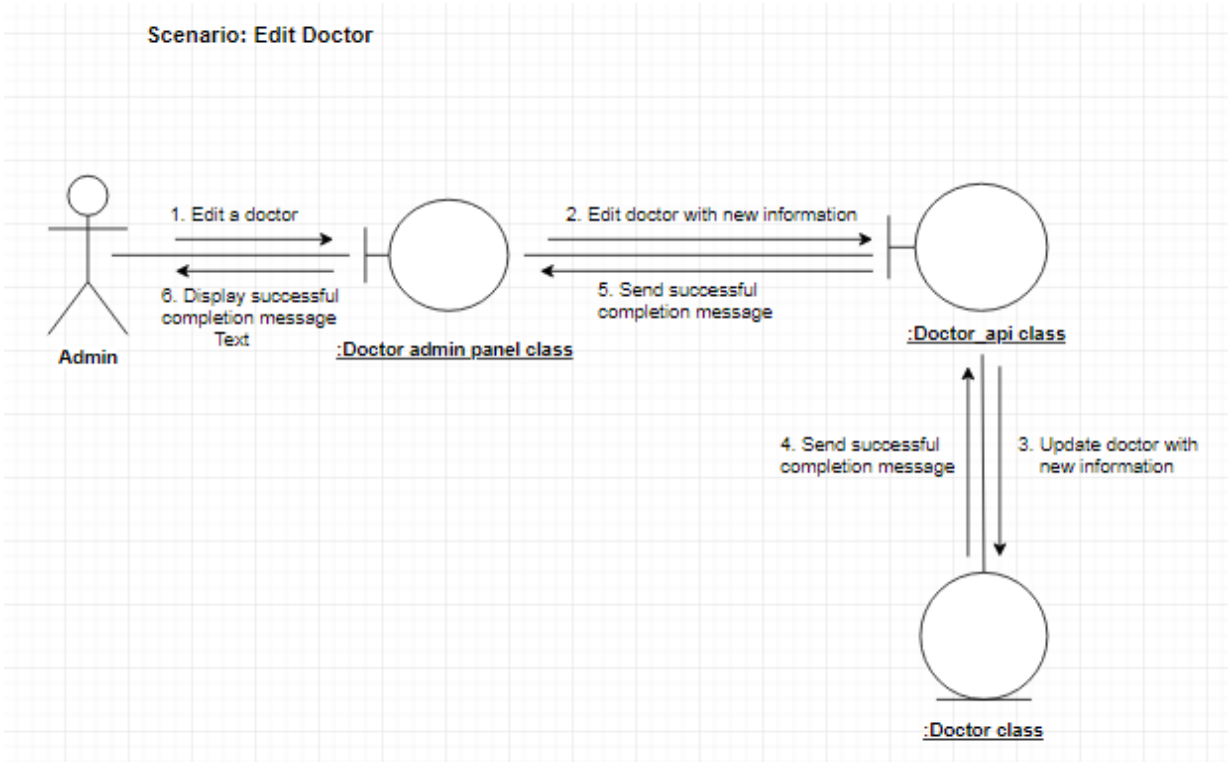


Sequence diagram

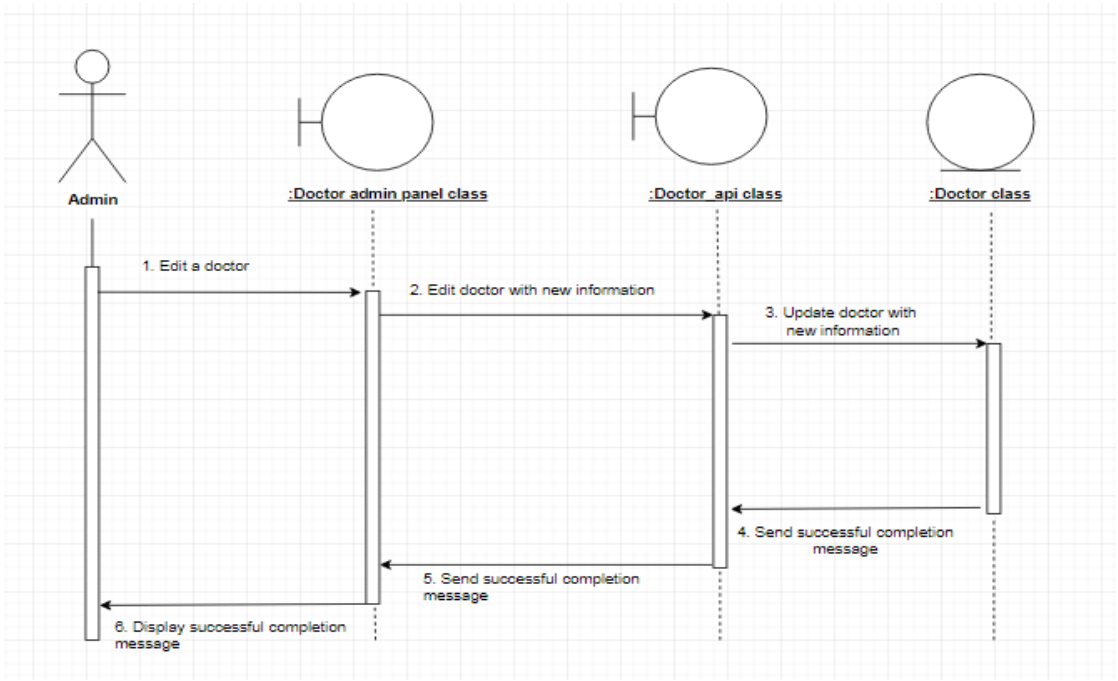


Scenario 3: Edit Doctor

Collaboration Diagram



Sequence Diagram



4. Test Cases

4.1 Login

Users need to be identified and authorized. Users enter their username and password to login the system. After their username and password are verified, home page is displayed. Meanwhile, a session will store username and other key information to trace user's state.

Case#	Input Data	Expected Results
Case 1: Login page	Enter ● correct user Name ● correct password and ● press on login Button	● Displays the welcome information to the user
Case 2: Login page	Enter ● correct User Name ● incorrect Password and ● press on login Button	● Displays error message:" Please check your user name and password and try again."
Case 3: Login page	Enter ● incorrect User Name ● correct Password and ● Press on login Button	● Displays error message :"Please check your user name and password and try again.
Case 4: Login page	● Not enter any username or password ● Press login button.	● Display error message " please input your username and password to retry."
Case 5. Login page	● Click "Signup" button to register as a new patient	● Link to the "register as new patient " page.
Case 6 Login fail page	● Click "try again" hyperlinks	● Links to the login page for users to retry.

4.2 Logout

Users logout, they are not be able to use system unless they login again. Therefore, after user logout, the login page will be re-directed, and the related home page is not available. Also, after

logout, user should not be able to return to the previous web page by clicking “backward ” button on web browsers.

Case#	Input Data	Expected Results
Case 1: Logout menu	● User click the logout menu	● Redirect to the login page ● The menu pages only has “login” and “register “ two menu items
Case 2: Backward menu	● After logout, user tries to return previous web page by clicking “backward” menu button on web browsers.	● Previous web pages should not be accessed, because the session is destroyed in logout.

4.3 Register User Profile (Patient)

On the home page, a new patient can choose ‘Signup’ option from the menu bar. The patient will be able to see a form where he/she will be required to fill in all the relevant information in the given fields. The patient can fill in the fields and press the ‘Submit’ button in the form to submit the information to the database.

Case #	Input Data	Expected Results
CASE 1	● Fill in all the fields in the registration form as required ● Press Submit button	● Display a message confirming that new patient record created successfully
CASE 2	● Leave all the fields empty ● Press Submit button	● Display an error message that user needs to fill in the required information ● Ask the user to enter information in the form again
CASE 3	● Fill in the fields according to an existing patient ● Press Submit button	● Display a message that the record already exists
CASE 4	● Fill in the fields as required ● Leave mandatory field/s empty ● Press Submit button	● Display an error message that required field/s is not filled in ● Ask the user to enter information in the form again

CASE 5	● Fill in a field and exceed the limit of number of characters/digits allowed ● Press Submit button	● Display an error message to show which field is not entered correctly ● Ask the user to enter information in form again
CASE 6	● Fill in a field with characters/digits not allowed for that field ● Press Submit button	● Display an error message to show which field is not entered correctly ● Ask the user to enter information in form again

4.4 Create Doctor Profile (Administrator)

A doctor should have a personal profile activated before being able to have agenda. The admin creates this profile. The administrator must login first, then clicks on the “add a doctor” from the menu, then fill a special form for doctor’s info and once completed a new doctor account is active and can receive appointments.

Case #	**Input Data**	**Expected Results**
CASE 1	● Fill all fields with correct values ● Click on submit button	● A message is displayed informing the administrator that a doctor profile was created successfully.
CASE 2	● Fill all other fields in the form correctly. ● Click on submit button	● An error message displayed, informing the administrator of a duplicate login-ID provided.
CASE 3	● The password field is filled with a value different from that given in the “Confirm Password” field. ● All other fields are filled correctly. ● Click on submit button.	● A pop up error message is displayed informing the administrator that the two passwords do not confirm.
CASE 4	● An email address entered but missing the dot (.) part, or the @ sign. ● All other fields are entered correctly. ● Click on Submit button	● A pop up error message is displayed informing the administrator that the email address is incomplete.
CASE 5	● The phone number field is filled with letters only or a combination of letters and numbers.	● A pop up error message is displayed informing the administrator that the

	● All other fields are filled correctly. ● Submit button is clicked.	phone number is should not be characters
CASE 6	● All form fields are left empty ● Submit button is clicked.	● A pop up error message is displayed listing all the missed fields.
Case 7	● All/Some field are filled with some data ● The “Reset” Button is clicked	● All Fields will become empty.

4.5 Book an Appointment(Patient)

After logging into homepage, the patient/user has the option to schedule an appointment. In order to successfully schedule an appointment, the patient must follow a process: select month and year, select day, select time, and confirm. Once this process had been successfully completed, the patient will receive a confirmation.

Case #	**Input Data**	**Expected Results**
CASE 1	▪ Arrive at Schedule Appointment interface	▪ Backend: Personal information (patient key, user ID, health card number, first name, last name, date of birth, address, telephone number, email address) is stored in session ▪ Interface: Step 1 (select appointment type) is displayed; displays appointment types from the database
CASE 2	▪ Select month from drop down menu (default is January) ▪ Select year from drop down menu (default is 2006) ▪ Click Submit button	▪ Backend: Month and year are stored in session ▪ Interface: Step 2 (select day) is displayed; displays calendar of month and year
CASE 3	▪ Click on a day from the calendar	▪ Backend: Day is stored in session

		▪ Interface: Step 3 (select time) is displayed; displays time slots for day according to estimated duration of appointment type; displays available time slots in green and un-available time slots in red according to database
CASE 4	▪ Click on a green time slot (red time slots cannot be clicked)	▪ Backend: Start time and end time are stored in session ▪ Interface: Step 4 (confirm) is displayed; displays appointment type, date, time with patient health card number, name, date of birth, address, telephone number, email address
CASE 5	▪ Click Confirm button	▪ Backend: Appointment (appointment key, doctor key, patient key, type key, date, start time, end time, memo, status) is inserted into the database ▪ Interface: Confirmation is displayed; displays appointment type, date, time with patient health card number, name, date of birth, address, telephone number, email address

4.6 Edit Doctor Profile (Administrator)

The doctor's information may need changes; the administrator can modify the profile after logging in. He should click the "Edit Doctor Profile" button from the administrator menu. Then choose a specified doctor's name from a list, and then the profile form will appear with all doctor information displayed in its relevant fields.

Case #	Input Data	Expected Results

Case 1	• Nothing changed in the form fields. • Submit button is clicked.	• Backend: Fields related to the doctor chose are re-saved in the Doctor and User tables in the database. • Interface: A new web page viewed with a confirmation message.
Case 2	• All fields are filled correctly. • Submit button is clicked.	• A pop up message is displayed informing the administrator that the edit done successfully.
Case 3	• All/Some Fields in the form are left without modification. • Submit button is clicked.	• A pop up error message is displayed listing all the missed fields.
Case 4	• An email address modified missing the dot (.) part, the @ sign or both • All other fields are modified correctly or left without modification. • Click on Submit button	• A pop up error message is displayed informing the administrator that the email address is incomplete.
Case 5	• The phone number field is modified to contain letters only or a combination of letters and numbers. • All other fields are modified correctly or left unchanged • Submit button is clicked.	• A pop up error message is displayed informing the administrator that the phone number is should not be characters
Case 6	• All/Some fields are modified. • The “Reset” Button is clicked	• All Fields will have their initial value again.