

Part 2

Team:

1. Prithvi Nath Manikonda
2. Priyanka Pashte
3. Shreya Joshi

Title: Health Tracker

Project Summary:

A health monitoring web application aimed at building a system that collects data from sensors and can keep track of various health parameters such as heart rate, calories burnt, sleep analysis, etc. A database of the Patients and Doctors profile would be maintained. The Patients can view their health parameters on the web page and report the same to a Doctor if they want an advice. Also, they can add an advising Doctor and schedule an appointment. The Doctor would be able to select a Patient, view their report and provide feedback.

a) Project Requirements:

1. Functional Requirements:

ID	Description	Priority
FR-01	The system must verify that the username is unique and password meets a set of guidelines at the time of registration.	High
FR-02	The system should be able to differentiate between a Patient and Doctor Login.	High
FR-03	If inactivity on the webpage persists for over an hour, then the Patient/Doctor is automatically logged out.	Medium
FR-04	Logging in would involve authentication via two factor authentication through E-mail.	High

2. User Requirements:

ID	Description	Priority
UR-01	Patient and Doctor can register and create their respective profile.	High
UR-02	Patient and Doctor can login and view their respective homepage.	High
UR-03	Patient and Doctor can edit their profile.	Medium
UR-04	Patient can view his health parameters for the past 3 days.	High
UR-05	Patient can view various Doctor's profiles.	Medium
UR-06 (a)	Patient can select an advising Doctor from the registered ones to monitor their profile.	Medium
UR-06 (b)	Patient can deselect the advising Doctor.	Medium

UR-07	Patient can request for health advice from his advising Doctor.	High
UR-08 (a)	Patient can Book an appointment with a specific Doctor.	High
UR-08 (b)	Patient can Reschedule a booked appointment with a specific Doctor.	High
UR-08 (c)	Patient can Cancel a booked appointment with a specific Doctor.	High
UR-09	Doctor can view the profile of his Advising patients.	High
UR-10	Doctor can view the health parameters of his advising patients.	High
UR-11	Doctor can send feedback/emergency notification to his patients.	High
UR-12	Doctor can cancel his appointment with a Patient.	High
UR-13	Patients and Doctors can Delete their respective profiles.	Low
UR-14	Patient and Doctor can logout.	High
UR-15	Patient and Doctor are allowed to reset the password.	High

3. Non-Functional Requirements:

ID	Description	Priority
NFR-01	Every webpage that is accessed must be loaded within 5 seconds.	High

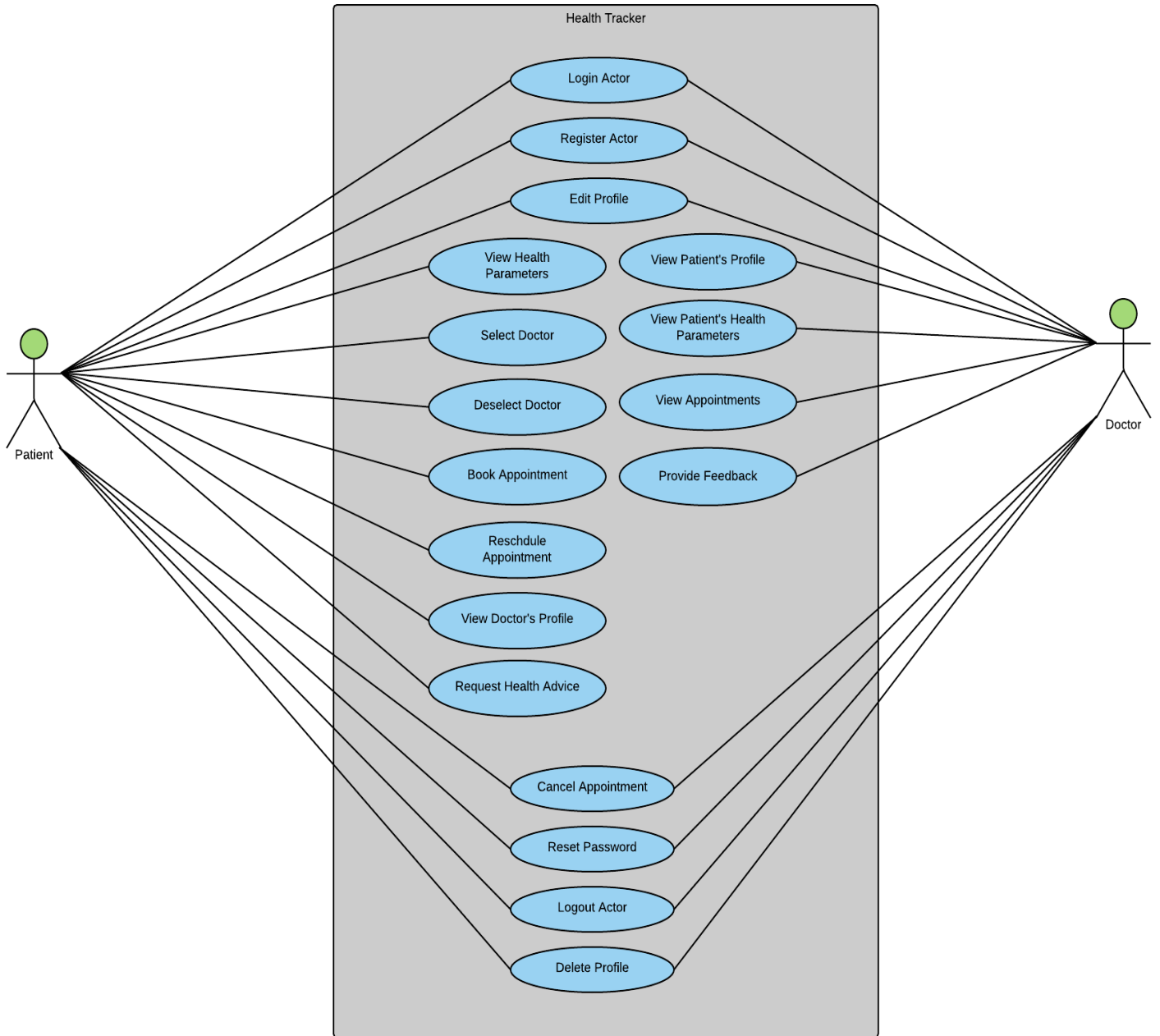
4. Business Requirements:

ID	Description	Priority
NFR-04	While executing a search, the system must be able to display 10 search results per page.	Low

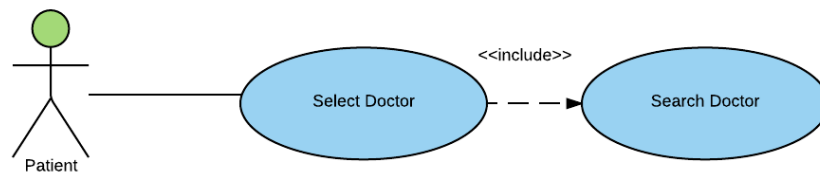
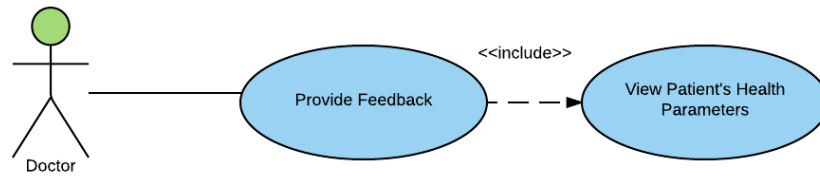
b) Use Cases:

1. Actors: Patient, Doctor

2. Use Case Overview:



3. Sub-diagrams:



4. Use Case Documents:

Use Case ID:	UC-01
Use Case Name:	Register Actor
Description:	A new actor should be able to register and create a profile.

Actors:	Patient, Doctor		
Pre-conditions:	Actor does not have a registered profile in the system.		
Post- conditions:	The actor is successfully registered with the system.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Open the webpage.	Display web page contents.
	2	Click on “Register” button.	Ask for a unique username, password and account type.
	3	Enter preferred credentials for username and password and select a radio button corresponding to Patient or Doctor.	
	4	Click on “Create Profile” button.	Verify the uniqueness of username and if the password guideline requirements are met and redirect to respective create profile page.
	5	Create a profile by entering the required details.	
	6	Click on “Submit” button.	Register the actor successfully by saving their data in database and display the respective homepage.
Variations:	3. Actor has not provided unique username and appropriate password.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-02
Use Case Name:	Login Actor
Description:	An actor should be able to login and view their respective webpage.

Actors:	Patient, Doctor		
Pre-conditions:	Actor must be registered.		
Post- conditions:	The actor is successfully logged in and can view his webpage.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Upon opening the webpage, click on "Login" button.	Ask for the actor's username and password.
	2	Enter username and password.	
	3	Click on "Submit" button.	Generate a code and send to the actor through email for verification.
	4	Enter verification code.	
	5	Click on "Verify" button.	Redirect to actor's respective homepage.
Variations:	3. Username and password do not match. 5. If verification code is entered incorrectly for 3 times, the actor is redirected to the login page.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-03
Use Case Name:	Edit Profile
Description:	The actor should be able to edit and update his profile details.

Actors:	Patient, Doctor		
Pre-conditions:	Actor is registered and has logged in to his homepage.		
Post- conditions:	The actor will have an updated profile.		
Frequency of Use:	Weekly		
Flow of Events:		Actor Action	System Response
	1	Click on “Edit Profile” button.	Redirect to edit profile page with pre-filled details.
	2	Update the details.	
	3	Click on “Save” button.	Update changes into database.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-04
Use Case Name:	View Health Parameters
Description:	A Patient must be able to view his health parameters.

Actors:	Patient		
Pre-conditions:	Actor must be logged in.		
Post- conditions:	A patient is able to view his health parameters.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on “View Health Parameters” button on the homepage.	Redirect to the Health Parameters page, consisting of three buttons: “Today”, “Yesterday” and “Day Before Yesterday” (Today being selected by default) to view the respective day’s data.
	2	Click on “Today” button.	Display respective day’s health parameters.
Variations:	2. Click on “Yesterday” button. 2. Click on “Day Before Yesterday” button.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-05
Use Case Name:	Select Doctor
Description:	A Patient must be to select the Doctor advising him.

Actors:	Patient		
Pre-conditions:	The patient must be logged in and must not have any advising Doctor.		
Post- conditions:	Patient must be able to see his advising Doctor.		
Frequency of Use:	Weekly		
Flow of Events:		Actor Action	System Response
	1	Click on “Select /Deselect Doctor” button on the homepage.	Display “No Advising Doctors present “.
	2		Display “Add Advising Doctor” button.
	3	Click on the “Add Advising doctor" button.	Provide search parameters with multiple drop down lists to search a Doctor based on Specialization, Location and to choose a sort preference.
	4	Select the preferred Specialization, Location and sort preference.	
	5	Click on “Search” button.	Display a sorted list of top 10 Doctors that matches the specifications.
	6	Select the Radio button corresponding to the Doctor preferred.	
	7	Click on “Select” button.	Add the Doctor selected as the Patient’s Advising Doctor.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-06
Use Case Name:	Deselect Doctor
Description:	A Patient must be able to deselect the Doctor advising him.

Actors:	Patient		
Pre-conditions:	The Patient must be logged in and must have an advising Doctor.		
Post- conditions:	Patient has successfully deselected his advising Doctor.		
Frequency of Use:	Weekly		
Flow of Events:		Actor Action	System Response
	1	Click on “Select /Deselect Doctor” button on the homepage.	Display the current advising Doctor.
	2		Display “Deselect Doctor” button.
	3	Click on the “Deselect Doctor” button.	Ask for confirmation to deselect the Doctor.
	4	Click on “OK” button to confirm the deselection.	Display “No Advising Doctors present”.
	5		Display “Add Advising Doctor” button.
Variations:	4. Click on “Cancel” button. Displays “Select/Deselect” Page.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-07
Use Case Name:	Request Health Advice
Description:	A Patient must be able to request health advice through E-mail from his advising Doctor.

Actors:	Patient		
Pre-conditions:	The Patient must be logged in.		
Post- conditions:	The Patient successfully sends the request to the advising doctor to provide health advice.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on “Request Health Advice” button on the homepage.	Provide the details of the advising doctor and “Compose E-mail” button.
	2	Click on “Compose E-mail” button.	Display a page to compose email.
	3	Compose the email.	
	4	Click on “Send” button.	Send an email to the respective advising Doctor.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-08
Use Case Name:	Book Appointment
Description:	A Patient must be able to book an appointment with a selected Doctor.

Actors:	Patient		
Pre-conditions:	Patient has clicked on the “Appointment Scheduler” button on homepage.		
Post- conditions:	An appointment is successfully booked.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on “Book Appointment” button.	Provide multiple drop down lists to search a Doctor based on Specialization, Location, to choose a sort preference and a textbox to enter a date.
	2	Select the preferred Specialization, Location, sort preference and enter a Date.	
	3	Click on “Search” button.	Display a sorted list of top 10 Doctors that matches the specifications.
	4	Select the Radio button corresponding to the Doctor preferred.	
	5	Click on “Select” button.	Display the selected Doctor’s available slots containing the Day and Time information for two weeks starting from the Date entered.
	6	Select a slot by clicking on the Radio button corresponding to the slot.	
	7	Click on “Book” button.	Display the specific requested slot with all the details and request for confirmation.
	8	Click on “Confirm” button.	Store the appointment details in the database.
	9		E-mail the appointment details to the Patient.
Variations:	7. Click on “Cancel” Button. Redirect to search by parameters page.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-09
Use Case Name:	Reschedule Appointment
Description:	An Actor must be able to reschedule an appointment with a Doctor.

Actors:	Patient		
Pre-conditions:	Actor has clicked on the “Appointment Scheduler” button on homepage and has a booked appointment.		
Post- conditions:	An appointment is successfully rescheduled.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on “Reschedule Appointment” button.	Display current appointments.
	2	Select the radio button for the appointment to reschedule.	
	3	Click on “Select” button.	Ask to enter a date for a new appointment.
	4	Enter a date.	Display the doctor’s available slots containing the day and time information for two weeks starting from the Date entered.
	5	Select a slot by clicking on the Radio Button corresponding to the slot.	
	6	Click on “Reschedule Appointment” button.	Display the specific requested slot with all the details and request for confirmation.
	7	Click on “Confirm” Button.	Store the appointment details in the database.
	8		Email the new appointment details to the Patient.
Variations:	7. Click on “Cancel” Button. Redirect to “Appointment Scheduler” page.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-10
Use Case Name:	Cancel Appointment
Description:	An Actor must be able to cancel an appointment.

Actors:	Patient, Doctor		
Pre-conditions:	Actor has clicked on the “Appointment Scheduler” button on homepage and has a booked appointment.		
Post- conditions:	An appointment is successfully cancelled.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on “Cancel Appointment” button.	Display appointments.
	2	Select the checkbox for the appointment(s) to cancel.	
	3	Click on “Cancel” button.	Display all the appointment details and request for confirmation.
	4	Cancel the appointment by clicking on “Confirm” button.	Delete the appointment details from the database.
	5		Appointment cancellation email is sent to the actors.
Variations:	4. Click on “Cancel” button. Redirect to “Appointment Scheduler” page.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-11
Use Case Name:	View Doctor's Profile
Description:	A Patient must be able to view a Doctor's profile.

Actors:	Patient		
Pre-conditions:	Patient has clicked on the "Search Doctor" button on homepage.		
Post- conditions:	A Patient is successfully able to view a Doctor's profile.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Enter a search criteria to search for doctors by providing multiple drop down lists based on Specialization, Location and Name.	
	2	Click on "Search" button	Display the doctor names alphabetically.
	3	Select a radio button corresponding to the Doctor.	
	4	Click on "View Profile" button.	Display the selected Doctor's profile.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-12
Use Case Name:	View Patient's Profile
Description:	A doctor must be able to view a Patient's profile.

Actors:	Doctor		
Pre-conditions:	Actor must be logged in.		
Post- conditions:	A Doctor is able to view a Patient's profile.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on "View Patients" button on the homepage.	Display a list of the Patients.
	2	Select the radio button corresponding to a Patient.	
	3	Click on "View Profile" button.	Display the selected Patient's profile.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-13
Use Case Name:	View Patient's Health Parameters
Description:	A Doctor must be able to view a Patient's health parameters.

Actors:	Doctor		
Pre-conditions:	Doctor must be logged in.		
Post- conditions:	A Doctor is able to successfully view a Patient's health parameters.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on "View Patients" button on the homepage.	Display a list of the Patients.
	2	Select the radio button corresponding to a Patient.	
	3	Click on "View Health Parameters."	Display the selected Patient's health parameters page.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-14
Use Case Name:	Provide Feedback.
Description:	A Doctor must be able to provide feedback to a Patient.

Actors:	Doctor		
Pre-conditions:	Doctor must be logged in.		
Post- conditions:	A Doctor is successfully able to provide feedback to Patient		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on “View Patients.”	Displays list of Patients.
	2	Select the radio button corresponding to a Patient.	
	3	Click on “View Health Parameters” button.	Display the health parameters of the selected patient and provide a button to provide feedback.
	4	Click on “Provide Feedback” button.	An email screen is displayed to enter feedback.
	5	Enter the feedback.	
	6	Click on “Send” button.	An email containing the feedback is sent to the Patient.
Variations:	6. Click on “Cancel” to redirect back to the “View Health Parameters” page.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-15
Use Case Name:	View Appointments
Description:	A Doctor must be able to view his scheduled appointments.

Actors:	Doctor		
Pre-conditions:	Actor must be logged in.		
Post- conditions:	A Doctor is able to view his appointments.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on “View Appointments” button on the homepage.	Display a list of the appointments.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-16
Use Case Name:	Logout Actor
Description:	The actor clicks on Logout and their session is terminated.

Actors:	Patient, Doctor		
Pre-conditions:	The Actor must be logged in.		
Post- conditions:	The Actor is logged out.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	The Actor clicks on the “Logout” button.	The system logs out the actor and invalidates the session.
	2		The Login page is displayed.
Variations:			
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-17
Use Case Name:	Delete Profile
Description:	Actor deletes his profile.

Actors:	Doctor, Patient		
Pre-conditions:	The Patient/Doctor is registered and logged in.		
Post- conditions:	The Patient/Doctor Profile is successfully deleted from the database.		
Frequency of Use:	Low		
Flow of Events:		Actor Action	System Response
	1	Click the “Delete Account” button present on the homepage.	Ask for confirmation to delete the profile.
	2	Provide confirmation by clicking “Confirm” button.	Delete profile details from the database.
Variations:	2. Click on “Cancel” button to cancel the account deletion.		
Notes and Issues:			
Developer Notes:			

Use Case ID:	UC-18
Use Case Name:	Reset Password
Description:	The Actor should be able to reset the password.

Actors:	Patient, Doctor		
Pre-conditions:	No pre-condition		
Post- conditions:	The Password is successfully reset.		
Frequency of Use:	Daily		
Flow of Events:		Actor Action	System Response
	1	Click on the “Reset Password” link on the login page.	The Reset Password page is displayed where the actor is prompted to enter his E-mail ID.
	2	Enter the registered E-mail ID.	
	3	Click “Confirm” button.	Verify if the email is of the registered actor.
	4		Send out an email with the reset link.
	5	Click on the reset link.	Display page to enter the new password twice.
	6	Enter the password twice.	
	7	Click on the “Save” button.	The system displays the confirmation message that the password is reset.
	8		The actor is prompted to login with new credentials.
Variations:	3. If not verified, the actor is prompted to re-enter the E-mail ID. 7. If passwords don’t match, prompt for password again.		
Notes and Issues:			
Developer Notes:			

c) Activity Diagram:

REGISTER ACTOR

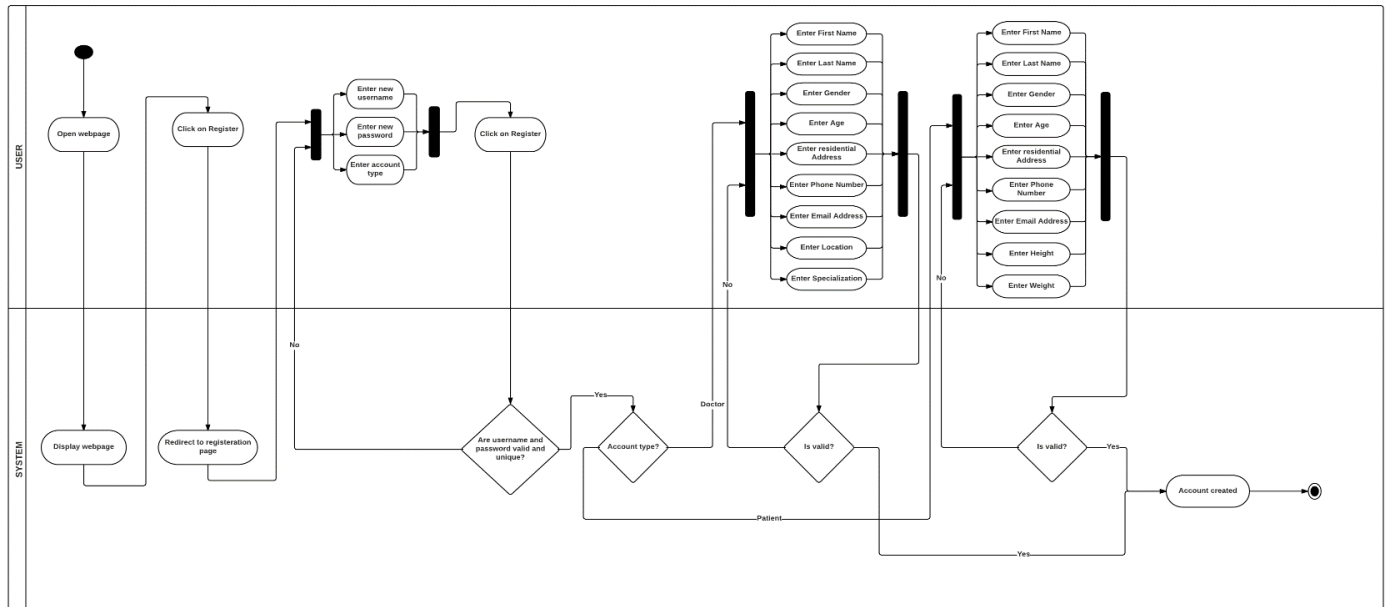
ACTIVITY DIAGRAM

SHREYA JOSHI | MARCH 7, 2016

REQUIREMENT #UR -01

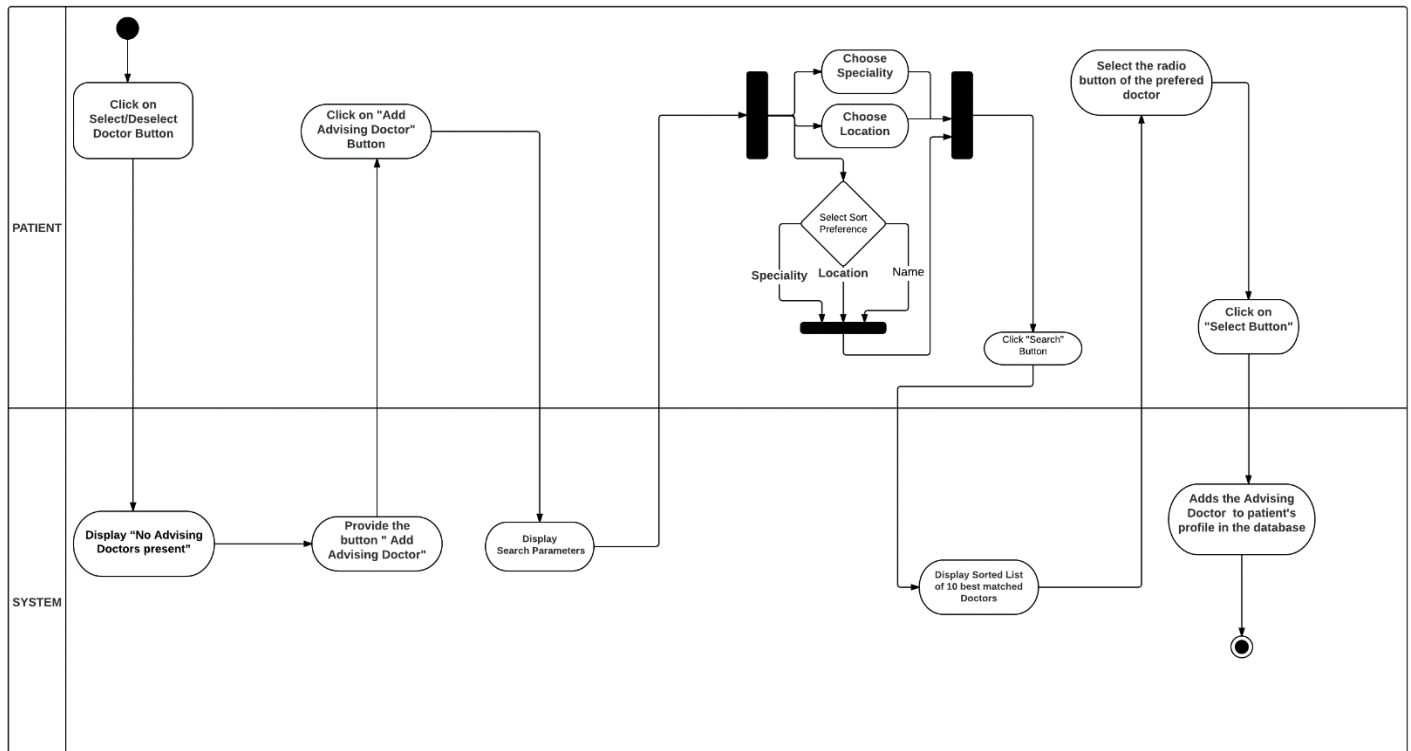
USE CASE #UC-01

A NEW ACTOR SHOULD BE ABLE TO REGISTER AND CREATE A PROFILE



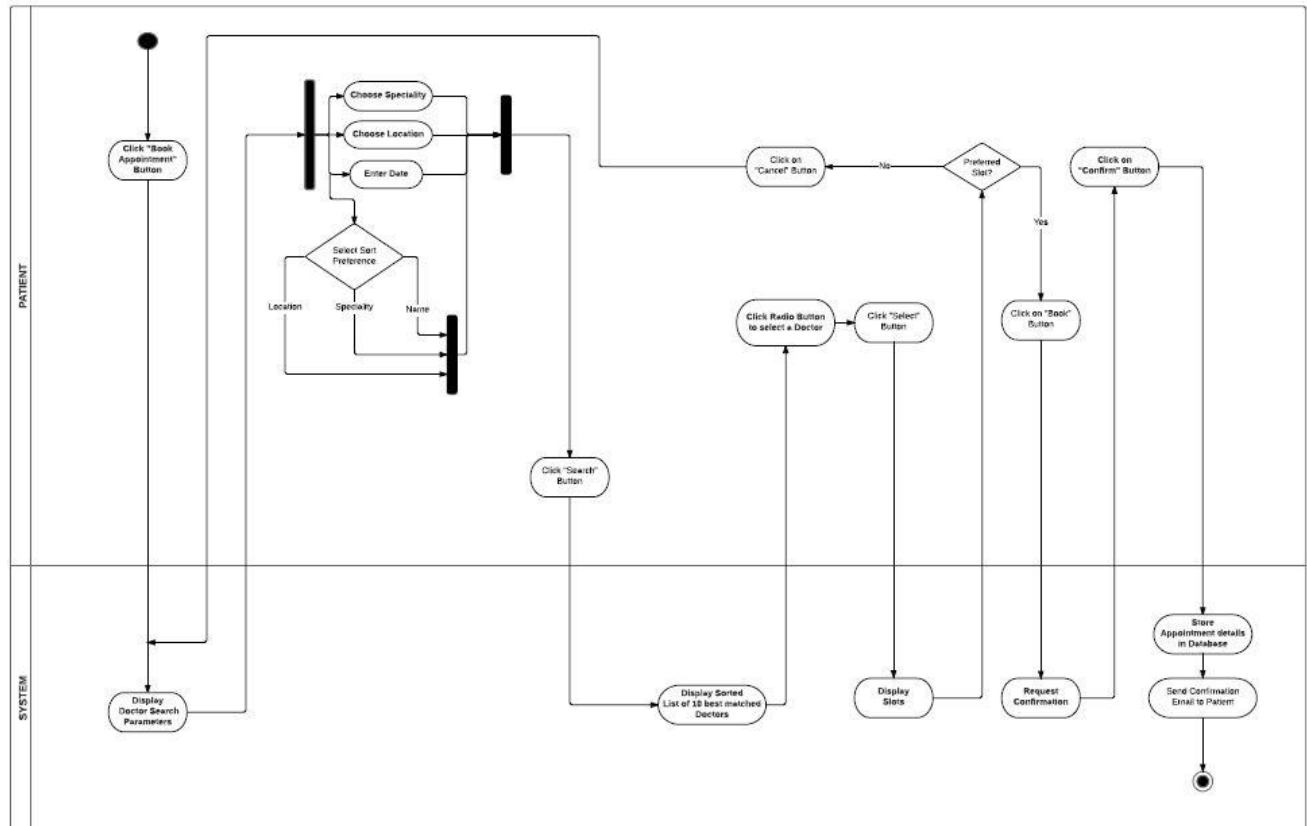
REQUIREMENT ID : UR-06(a) USE CASE ID: UC-05

Description: A Patient must be able to select the Doctor advising him.



REQUIREMENT ID : UR-08(a) USE CASE ID: UC-08

Description: A Patient must be able to book an appointment with a selected Doctor.

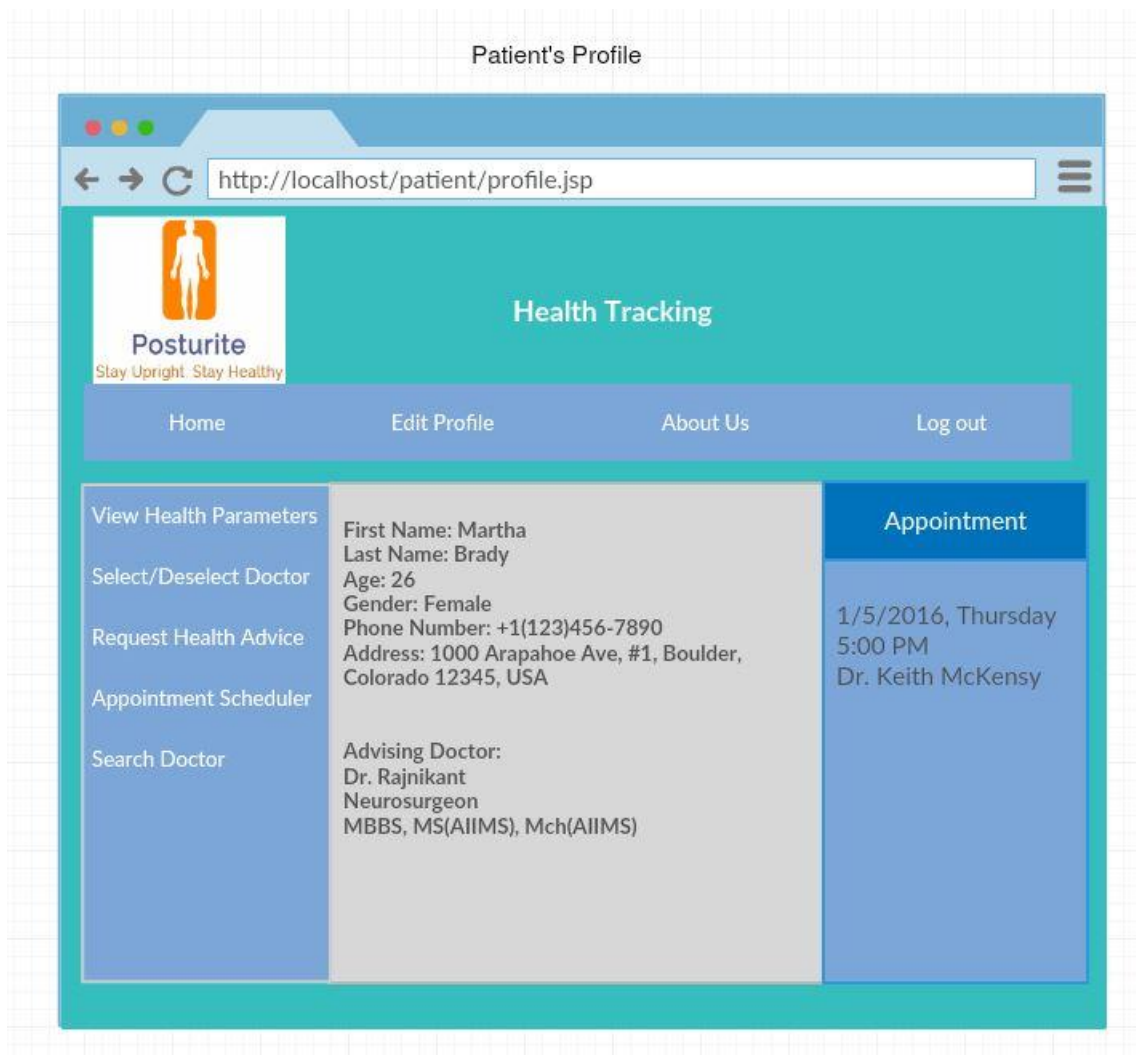


d) Data Storage:

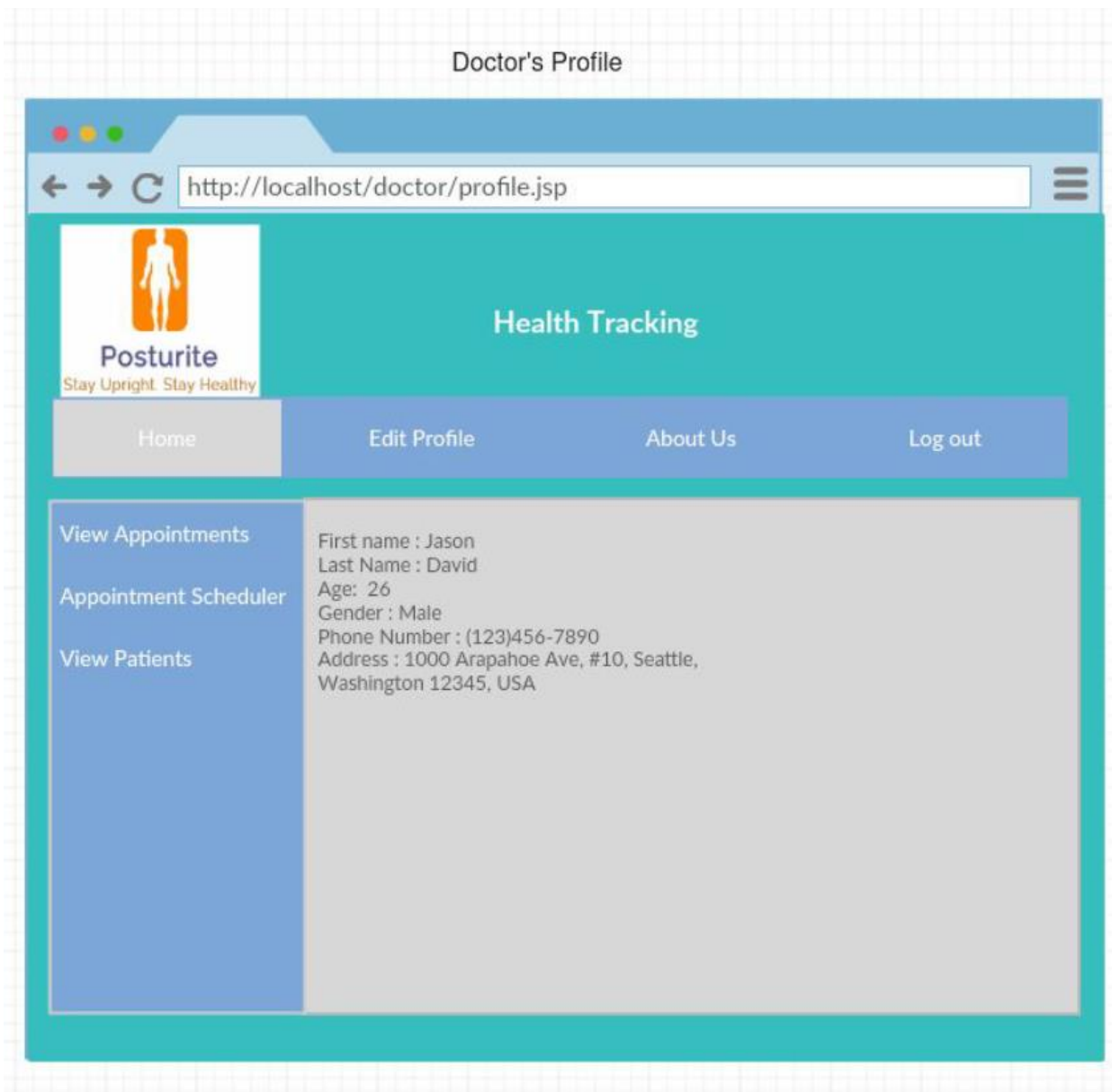
We are using MYSQL as the database and to connect with Java we are using JDBC connector driver from MySQL to persist and retrieve the data from the database. The interaction with database is done by DBHandler that makes queries and retrieves the data. The DBHandler class has various methods like validateUser, savePatientProfile, saveDoctorProfile, getPatientProfile, getDoctorProfile, validateUsername, setAppointment, cancelAppointment, getHealthParameters, resetPassword, deleteProfile. These methods perform various functions to update, delete, and add data into various tables. The DBHandler interacts with these tables- Patient, Doctor, Appointments, and Health Parameters. Each user - Doctor or Patient has a unique ID associated by his account. This ID is used as the primary key to retrieve/persist the raw data to be processed by the methods mentioned above in the DBHandler class.

e) UI Mockups:

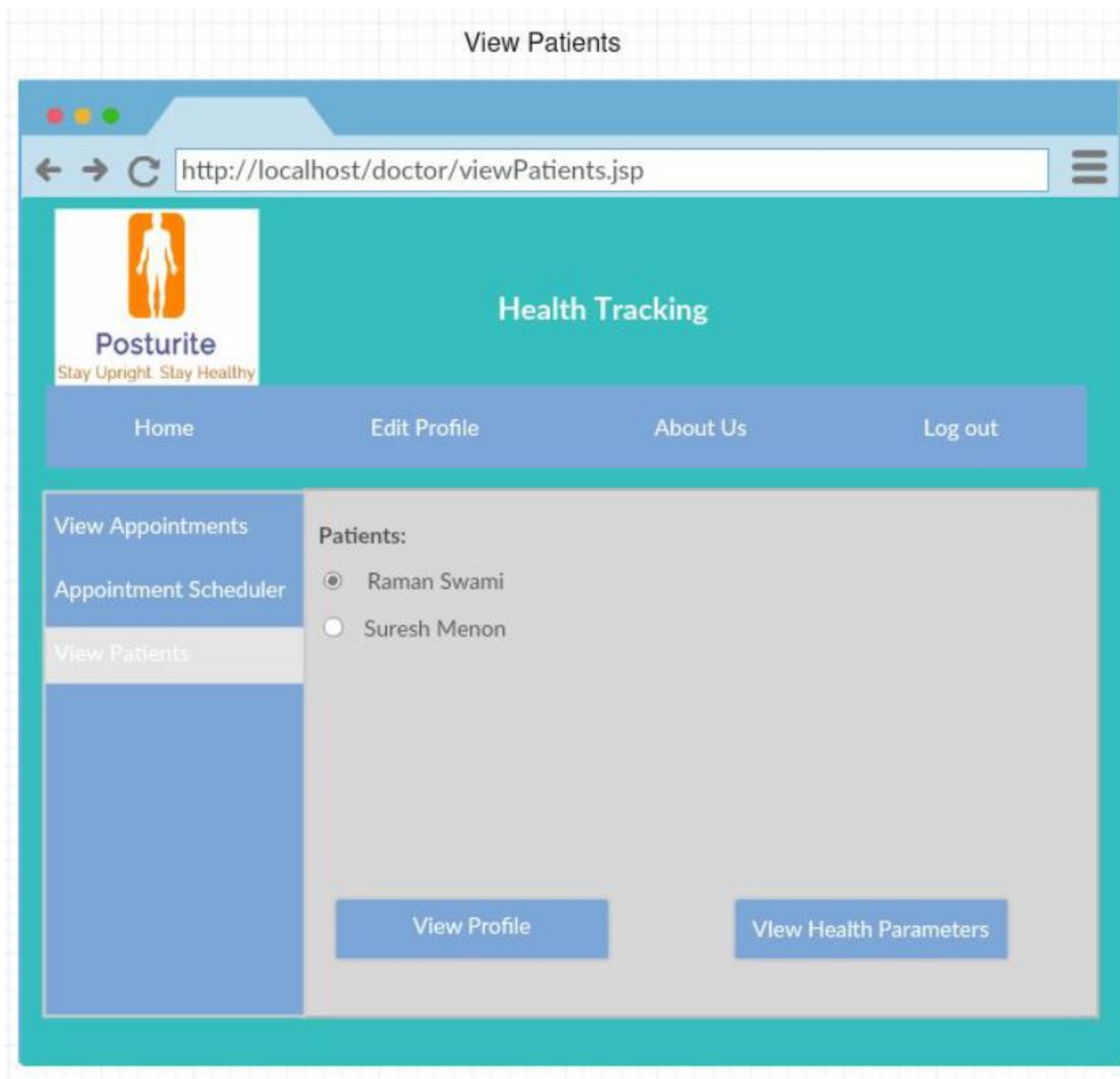
1. Patient's HomePage.



2. Doctor's Homepage



3. Doctor's View Patients Page



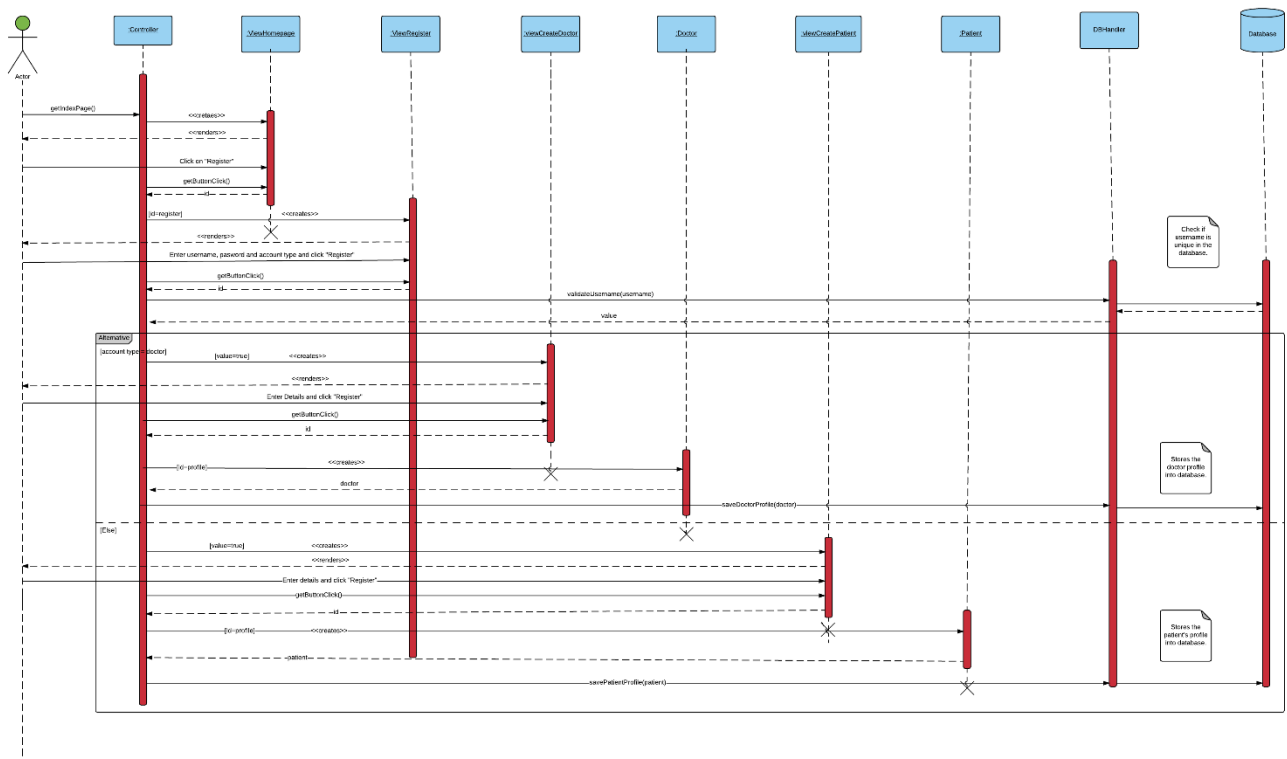
f) User Interactions:

REGISTER ACTOR

SEQUENCE DIAGRAM

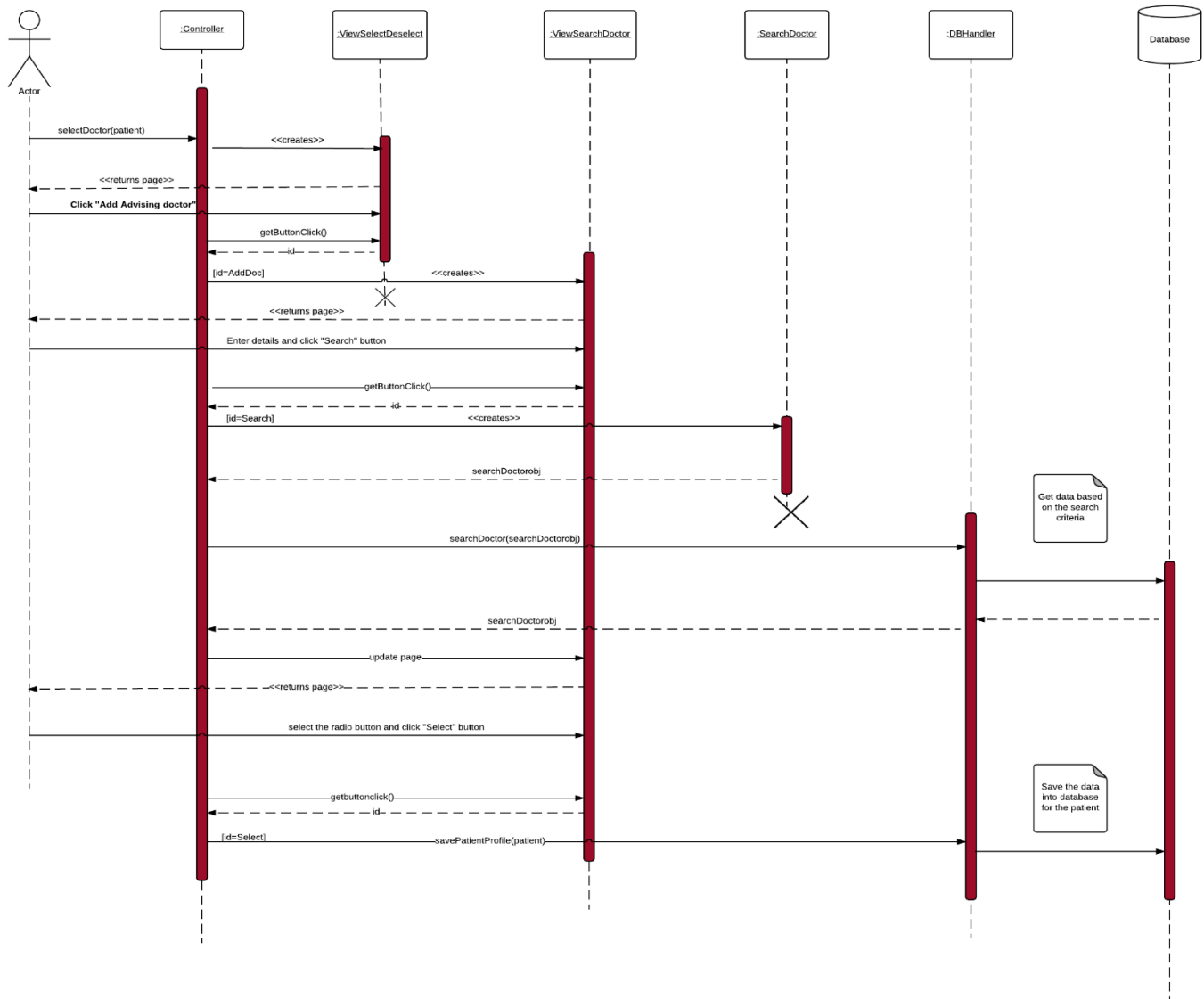
Shreya | March 9, 2016

REQUIREMENT ID : UR-01 USE CASE ID: UC-01
Description: Patient and Doctor can register and create their profile



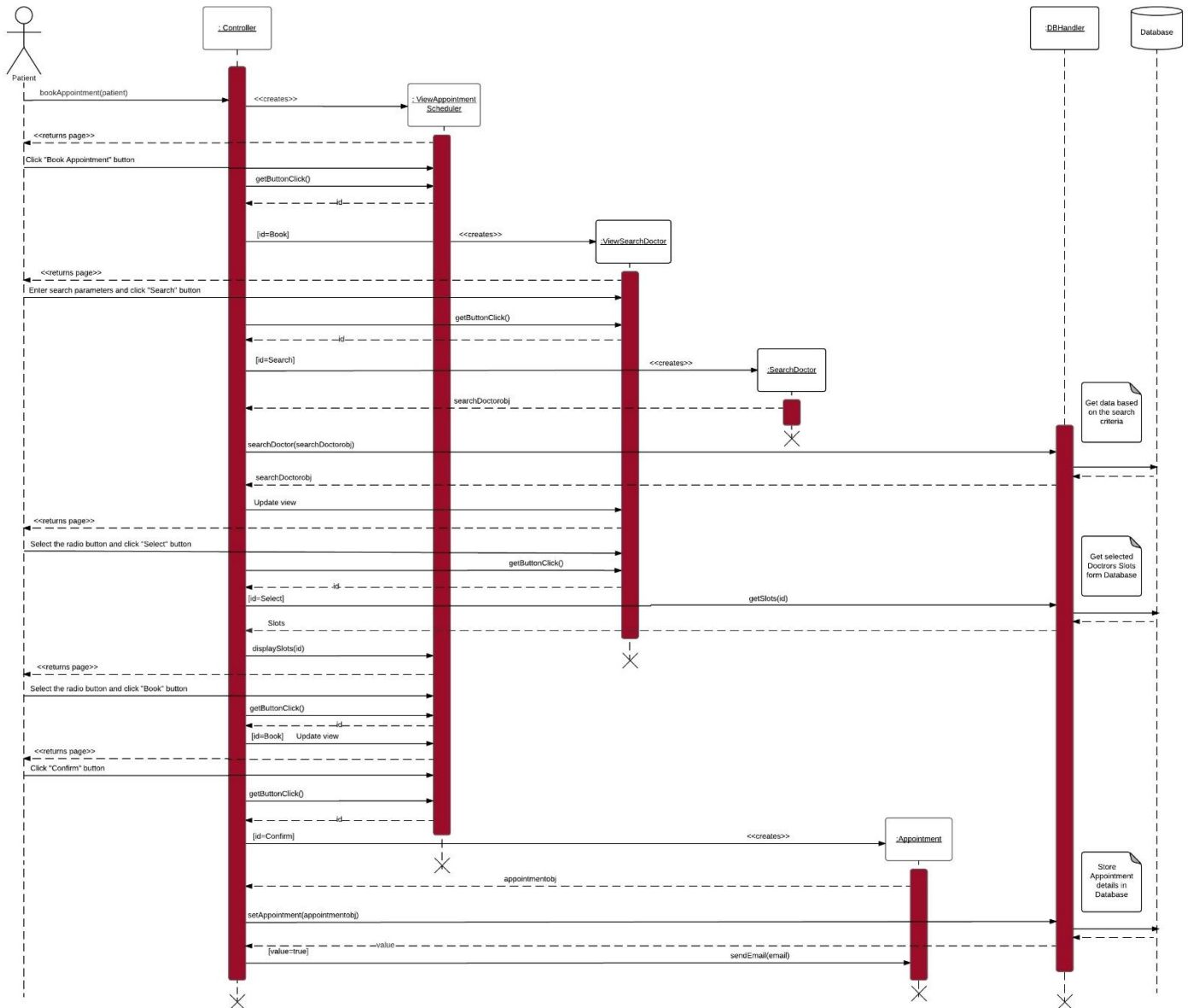
REQUIREMENT ID : UR-06(a) USE CASE ID: UC-05

Description: A Patient must be able to select the Doctor advising him.



REQUIREMENT ID : UR-08(a) USE CASE ID: UC-08

Description: A Patient must be able to book an appointment with a selected Doctor.



g) Class Diagram:

