

## **CPS731 Team 14**

**Topic:** Online Healthcare Patient Booking Management System

**Date:** Dec 4, 2021

**Professor:** Dr Sadaf Mustafiz

### **Group Members**

Sam Oliveros

500748415

[sam.c.oliveros@ryerson.ca](mailto:sam.c.oliveros@ryerson.ca)

Deep Oza

500830262

[deep.oza@ryerson.ca](mailto:deep.oza@ryerson.ca)

Thomas del Mundo

500812684

[tdel@ryerson.ca](mailto:tdel@ryerson.ca)

Zhongling Wang

500863783

[zhongling.wang@ryerson.ca](mailto:zhongling.wang@ryerson.ca)

## Problem Statement

## Requirements

[Informal Requirements](#)

[User Requirements \(Patients\)](#)

[User Requirements \(Employees\)](#)

## Design Models

[Use Cases Model \[Updated\]](#)

[Interaction Model: Communication Diagram](#)

[Domain Model](#)

[Design Class Model](#)

[State Machine Model](#)

[Test Cases](#)

## Design and implementation discussion

## Project Plan Review

# Problem Statement

During the Covid crisis, healthcare facilities are struggling to keep up with the patients and also accommodate new patients with new treatments each day. While the backlog of surgeries and treatments were increasing day by day, we as a development team were approached by Toronto General Hospital.

Our team was asked to develop a patient management system where each patient can meet the doctor and book ahead of time as long as a time slot permits. This way the system can manage patients ahead of time and cut down the wait timing for other patients.

## Requirements

### Informal Requirements

In our system, we need to have the latest information for our users. For the system to run smoothly, we need to use a database to track patients and their information; specifically: name, address, email ID etc.

The database should also keep track of all our facilities. If the healthcare provider has multiple locations, it should be able to display all the locations of the facility, and for each location the patient must be able to view the different services associated with each location. To access different locations/buildings, the system would have to list the address and contact information for the patient. Each patient has different needs and a particular doctor for their situation, so each location must address the different doctors or medical practitioners working there. This way, the patient can get the appropriate treatment that they need to recover. Lastly, the patient should be given available time slots and the appointment table should be updated with its appointment objects which must have a unique key.

## User Requirements (Patients)

When a company uses this system, patients can use this website/app to schedule future appointments. Patients can register as a new account in the system, providing their general, government, and contact information to their profile data, and confirming all information is true. All personal information must be protected and private to the patient. Patients who have accounts can access the appointments section, where they can schedule a new appointment, or to view and edit their current appointment. When scheduling a new appointment, patients can choose the location and doctor of their choice, provide a brief description of their situation, and choose an available date and time for their appointment. Patients can only see the available time slots, and not those already taken. An email confirmation will be sent after a successful scheduled appointment, which also gives an access code to view or change their appointment on the web application. If patients want to change or cancel an appointment, they can use their provided email address and this access code. Patients can view the details of their appointment, and the date and time can be changed, however, some time slots could possibly be or not be available anymore. Appointments can be cancelled, and patients will be given a confirmation email for any changes or cancellations.

## User Requirements (Employees)

Both medical and administrative employees of the healthcare facility must have an account which allows them to modify patient information manually and view their own schedules. Employee accounts are made by upper level system managers. The account should hold the employees' contact information, place of employment, and services they provide. When searching a patient, employees can view the patient's current appointment and information. When searching for a doctor, it will display their current appointments and availability. Employees can view their daily and weekly schedule of appointments at any point in time in a list format. Employees will also have control of modifying or deleting an existing appointment, as long as the patient provides consent. Any changes will follow up with a confirmation email to the associated patient.

As a development team, we think this project can be best accomplished by the incremental software life-cycle. Since our project is a web application that needs to be able to run on many different devices to best serve our customers, we need to be flexible in our development as hardware changes. With the incremental life-cycle, we can develop the system in increments and evaluate each increment before proceeding to the next one to ensure compatibility and stability on the platforms our customers use.

# Design Models

## Use Cases Model [Updated]

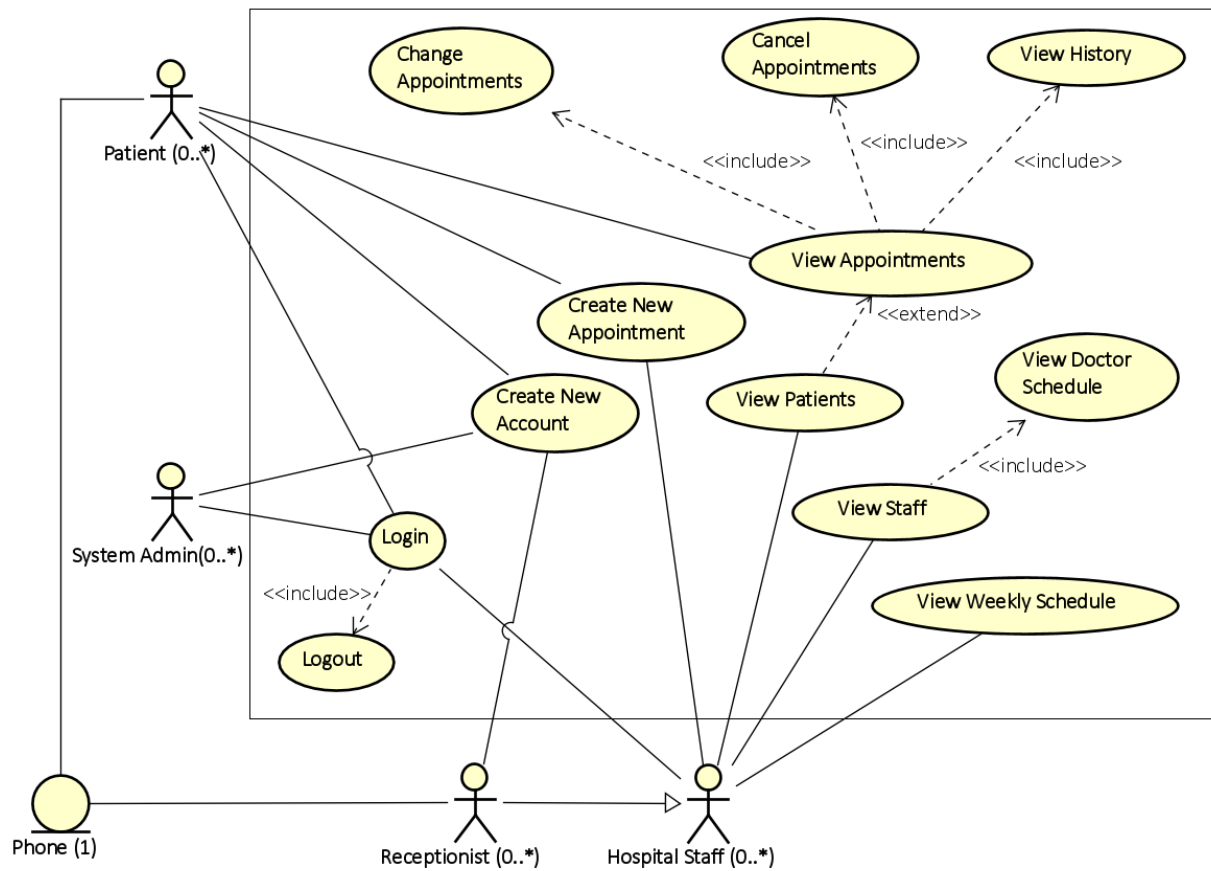


Figure 1: Online Booking Appointment Use Cases Diagram

# Interaction Model: Communication Diagram

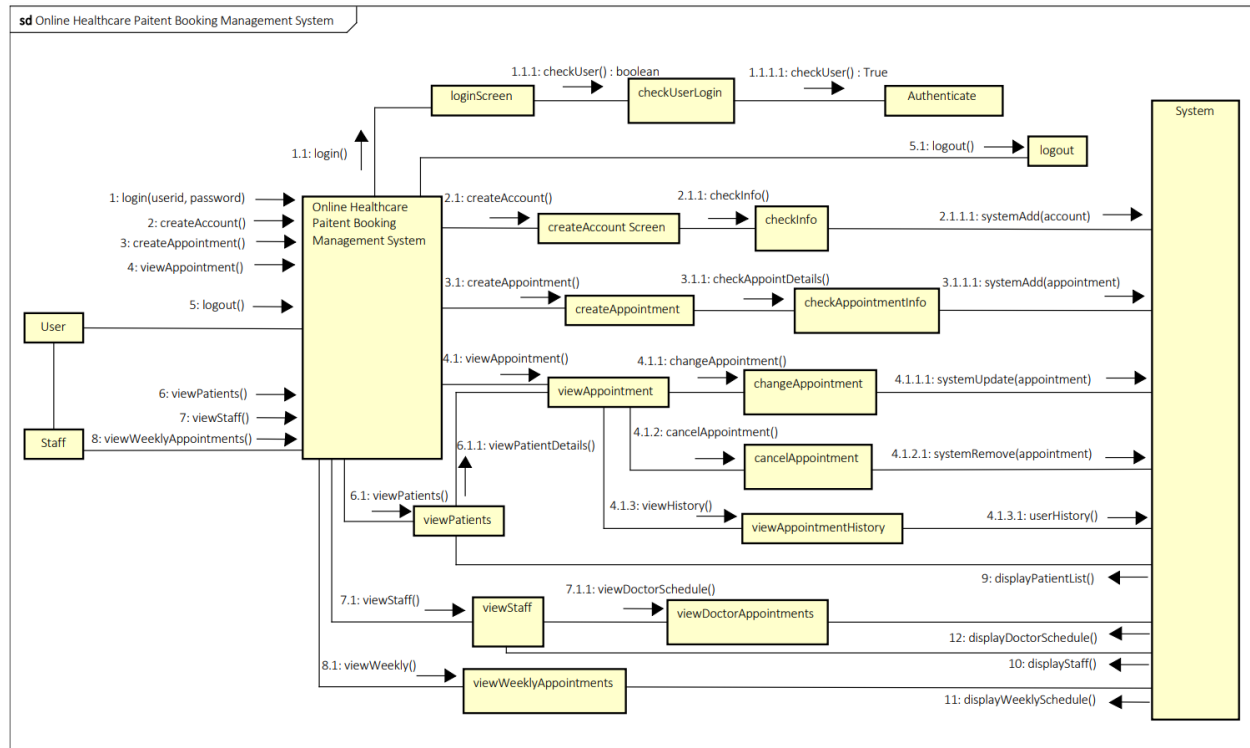


Figure 2: Online Booking Appointment Interaction Mode

# Domain Model

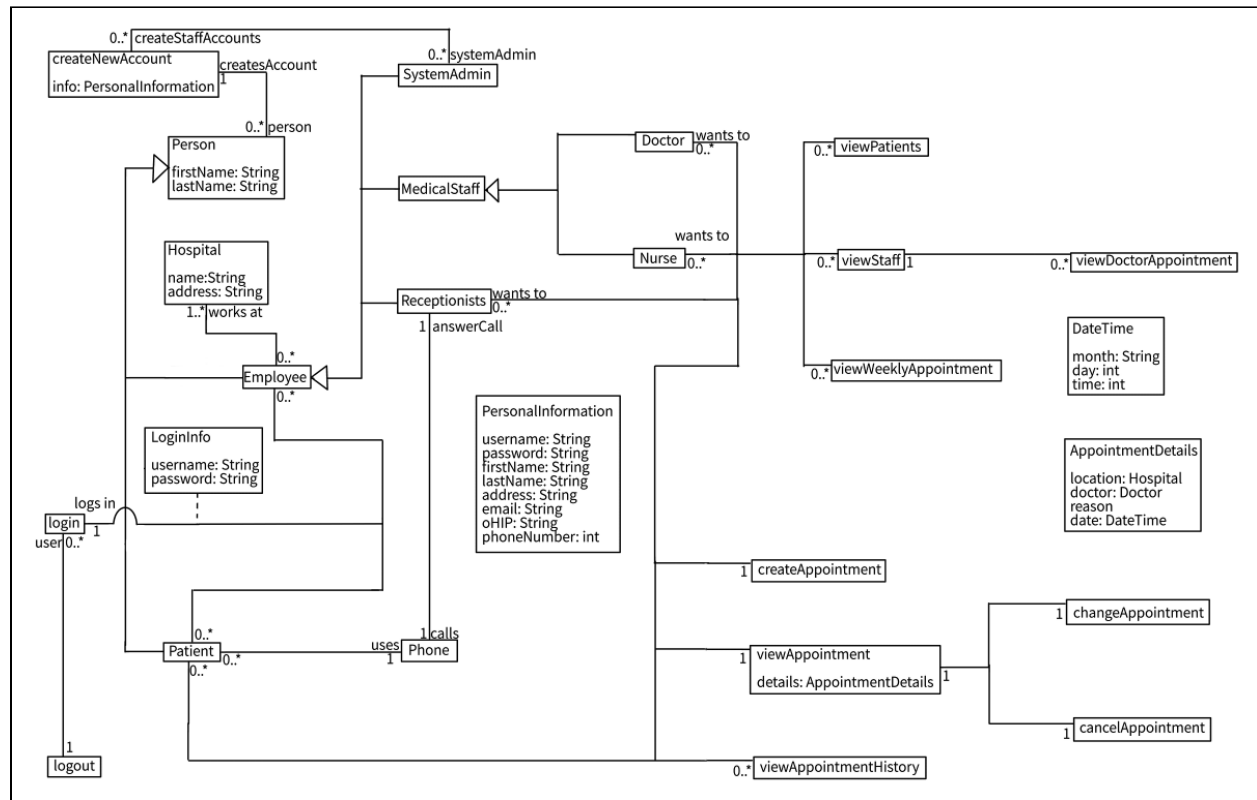


Figure 3: Online Appointment Booking Domain Model

# Design Class Model

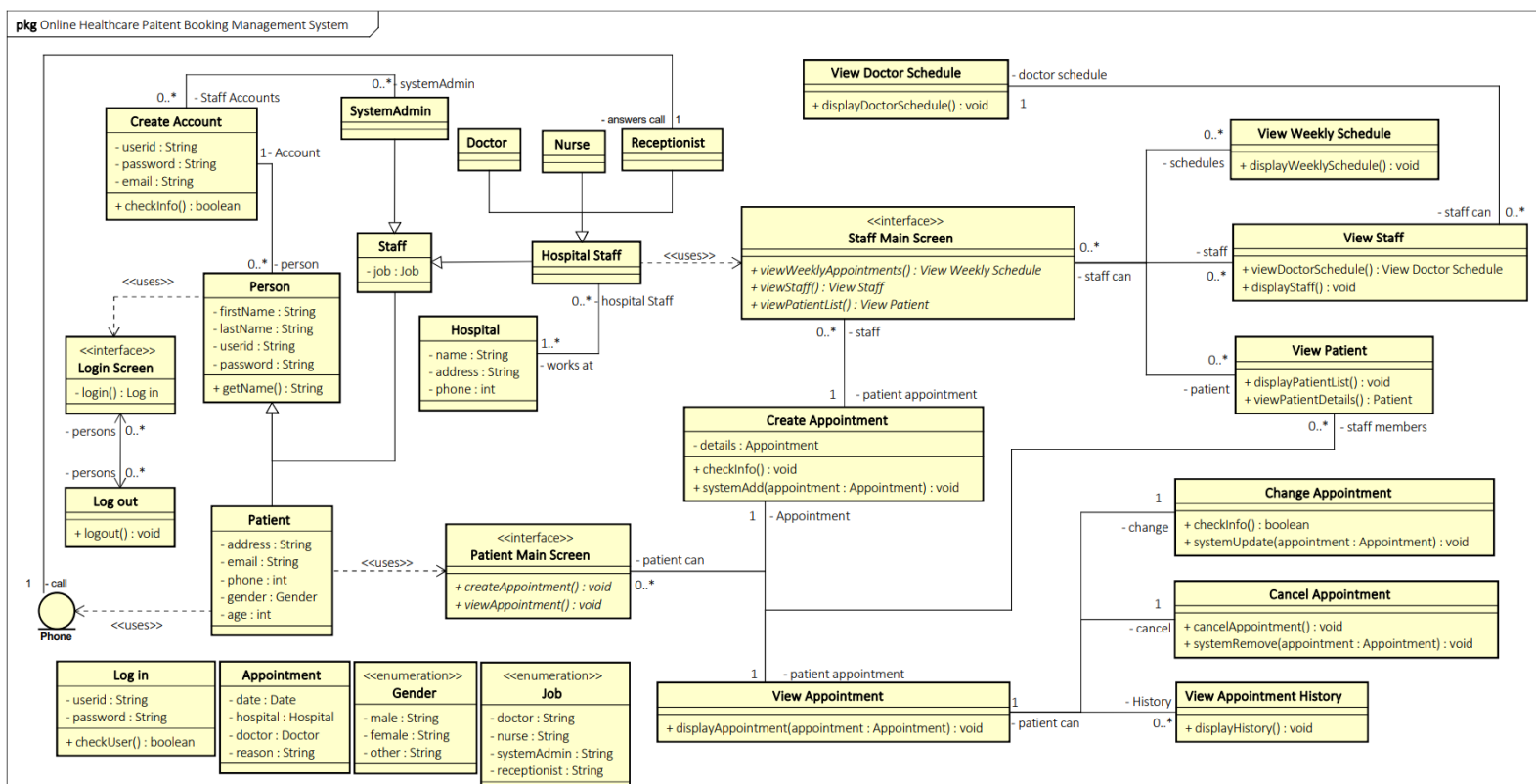


Figure 4: Online Booking Appointment Design Class

## State Machine Model

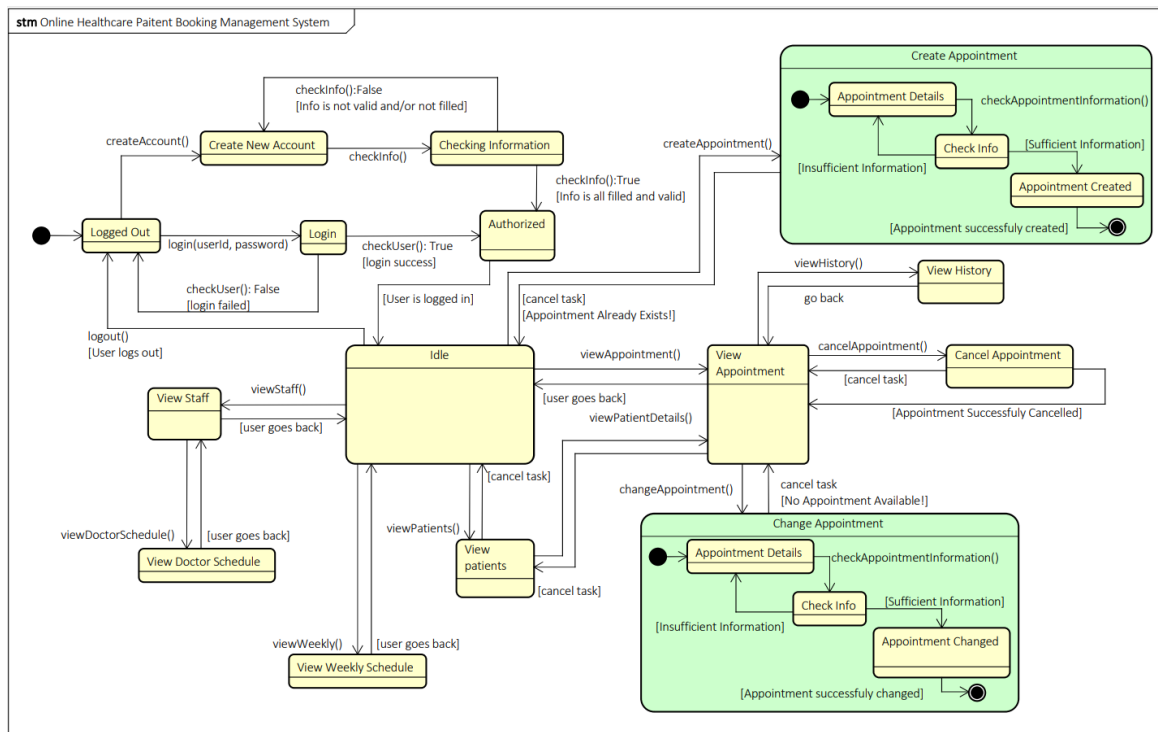


Figure 5: Online Booking Appointment State Machine Model

# Test Cases

## Core Functionality Tests

Test Case		Steps	Input	Expected Output
<b>Login Portal A</b>		<b>Login (Success)</b>	<b>Type</b>	
	1.	Input valid username and password combination	Alphanumeric strings for username and passwords	Successful login - Proceed to User Portal
<b>Login Portal B</b>		<b>Login (Failure)</b>		
	1.	Input invalid username and password combination into respective text fields	Strings for username and password	'Invalid credentials' error message
<b>Login Portal C</b>		<b>Create new account (Success)</b>		
	1.	Click "Create new account" link/button	Mouse input	Proceed to Account Registration form
	2. a)	Input new username	Alphanumeric String	Username passes all checks - 'Valid username' message
	2. b)	Input new password	Alphanumeric String with special characters	Password meets length and character requirements - 'Okay password' message
	2. c)	Re-input new password	The same string as 2. b)	String from 2. b) and 2. c) matches - 'Valid password'



				message displayed
	2. c)	Input first name	Alphabet, '-', and "" characters only	'First name' field non-empty - 'FirstName' flag active
	2. d)	Input last name	Alphabet, '-', and "" characters only	'Last name' field non-empty - 'LastName' flag active
	2. e)	Input address	Alphanumeric strings + ' '	'Address' field non-empty - 'Address' flag active
	2. f)	Input contact information		At least one 'Contact information' flag active
	2. f) i.	Input valid Phone Number	Numbers only	'Contact information' flag active
	2. f) ii.	Input valid Email Address	Alphanumeric string	'Contact information' flag active
	3.	Create account	Mouse input	New account created - Information in fields added to database
<b>Login Portal D</b>		<b>Create new account (failure)</b>		
	1 a)	Click "Create new account"	Mouse input	Proceed to Account

		link/button		Registration form
	b)	Input invalid new username		
	i)	Input taken username	Alphanumeric String	'Username has already been used' message
	ii)	Input username with invalid characters	Alphanumeric String	'Username contains invalid characters' message
	c)	Input invalid new password		
	i)	Input password without special characters	Alphanumeric String without special characters	Password need to meet the characters requirements- "Password requires at least 1 special character" message
	ii)	Input password too short	Alphanumeric String with special characters but not meet the length requirement	"Please does not meet length requirements" message
	2 d)	Input invalid first name	Alphanumeric string	"Name contains invalid characters" message
	e)	Input invalid last name	Alphanumeric string	"Name contains invalid characters"

				" message
	f)	Input address with special characters	Alphanumeric string	"Name contains invalid characters" message
	g)	Input invalid contact information		
	i)	Input invalid Phone Number	Alphanumeric string	"Invalid phone number" message
	ii)	Input invalid Email Address	Alphanumeric string but without @	"Please enter a valid email address" message
	2. h)	Click 'Create account' without all the forms filled or with forms incorrectly filled		
	i)	Click 'Create account' without all the forms filled	Mouse input	"Please fill in all the forms" message
	i)	Click 'Create account' with forms incorrectly filled	Mouse input	"Please fill in all the forms" message
	3.	Account creation interrupted		
	a)	Click 'Cancel' button	Mouse input	Return to Login Portal
<b>User Portal A</b>		<b>Create Appointment (Success)</b>		
	1	Click "Create Appointment" button	Mouse input	Proceed to 'Account Appointment' form (2. a))
	2. a)	Select Location and then Next	Mouse input	Proceed to b)

	b)	Input Problem description	Alphanumeric string	Proceed to c)
	c)	Select Doctor	Mouse Input	Proceed to d)
	d)	Select Time and Date	Mouse Input	Proceed to e)
	e)	Confirm Appointment	Mouse Input	Proceed to confirmation screen, send confirmation email, add appointment to database
<b>User Portal B</b>		<b>Create Appointment (Failure)</b>		
	1	'Cancel' button clicked during any page of appointment creation	Mouse input	Return to User Portal
<b>User Portal C</b>		<b>View Appointment (Success)</b>		
	1	Click "View Appointments" button	Mouse input	Proceed to Appointments list
<b>User Portal D</b>		<b>View Appointment (Failure)</b>		
	1	Click "View Appointments" button without having any pending appointments	Mouse input	"No Appointments" message
<b>Appointments A</b>		<b>Change Appointment (Success)</b>		
	1	Click "Change Appointment" button on an appointment	Mouse input	Proceed to 'Create Appointment' form with filled in details (User Portal A)
	2	Confirm Appointment	Mouse input	"This

				appointment has been changed successfully" message, appointment updated on database
<b>Appointments B</b>		<b>Change Appointment (Failure)</b>		
	1	Click "Change Appointment" button on an appointment	Mouse input	
	2.	Click 'Cancel' button	Mouse Input	Return to appointment list
<b>Appointments C</b>		<b>Cancel Appointment</b>		
	1	Click "Cancel Appointment" button	Mouse input	"This appointment has been cancelled successfully" message, appointment removed from database
<b>Appointments D</b>		<b>View History (Success)</b>		
	1	Click "Past Appointment" button	Mouse input	List of all past Appointments and changes made
<b>Appointments E</b>		<b>ViewHistory(Failure)</b>		

	1	Click "Past Appointment" button without any appointments having passed nor changed	Mouse input	"No appointments found" message
<b>Staff A</b>		<b>View Patients (Success)</b>		
	1	Click "View Patients" button	Mouse input	List all patients with an appointment with the doctor
	2	Click their name on the list	Mouse input	List all the patient's appointments with the doctor
<b>Staff B</b>		<b>View Patients (Failure)</b>		
	1	Click "View Patients" button when no patients are scheduled for the doctor	Mouse input	"No patients found" message
<b>Staff C</b>		<b>View Doctor Schedule(Success)</b>		
	1	Click "View Doctor's Schedule" button when the doctor has at least one patient scheduled	Mouse input	List all of the appointments for the corresponding doctor
<b>Staff D</b>		<b>View Doctor Schedule(Failure)</b>		
	1	Click "View Doctor's Schedule" button when no patients are schedule with the doctor	Mouse input	"No appointments found" message
<b>Staff E</b>		<b>View Weekly Schedule</b>		
	1	Click "View Weekly Schedule" button	Mouse input	Shows all the appointments on the Calendar
<b>Staff F</b>		<b>View WeeklySchedule</b>		

	1	Click "View Weekly Schedule" button with no appointments scheduled	Mouse input	Shows the Calendar.
<b>Staff G</b>		<b>View Staff</b>		
		Click 'View Staff' Button	Mouse input	Lists all staff
<b>General A</b>		<b>Logout</b>		
	1	Click "Logout" button	Mouse input	Logout successful
<b>General B</b>		Force exit application		
	1	On any screen: exit application without using the 'Logout' button		All forms closed without saving, user logs out

# Design and implementation discussion

When it comes to the design and implementation, we changed a few aspects of our project. We decided to change our user interface, and implementation of the staff requirements will be delayed for our next feature. The patient requirements have been successfully implemented, which is the main use and functionality of our software. Although, patients can not currently view their previously made or changed appointments due to lack of code knowledge thus unable to display the proper information. We decided to set this implementation in the next update.

## Project Plan Review

In accordance with our project plan, we have been able to successfully develop key aspects of our application. Specifically, the ability for patients to log in and book appointments at their chosen healthcare facilities has been attained. Additionally, the ability to register new patients online and view their appointments have been implemented. A basic user interface has been created to accommodate the current functionality.

Overall progress is as follows:

- Phase 1: Complete
- Phase 2: Currently underway
- Phase 3: Pending

As outlined in the project plan, it was expected that phase 2 would take the longest amount of time as it is where most development of our application takes place. Unfortunately, the first cycle of phase 2 was unable to include all of the features listed in the plan but as stated earlier, the most important functions have been implemented. Accordingly, these features have been delayed to following cycles and future estimations in regards to time and difficulty have been adjusted accordingly. These delays have been anticipated and will not incur any further changes to the project plan as a whole.

Of the Cycle One features, the following features have been implemented:

- The ability for a user to register
- The ability for a user to login to the system once registered
- The ability for a user to create an appointment at the listed facilities
- The ability for a user to view the details of their next appointment
- The ability for a user to change or delete an existing appointment
- The ability for a user to logoff
- Basic UI to facilitate the above features

The following features have been delayed:

- Separate subsystem for staff and related functionality
  - Staff user subtype
  - List of patients
  - List of all appointments
  - Doctor's schedule
- The ability for a user to view all their past appointments