

Software Requirements Analysis and Design

T27 - Brilliant Dentistry CRM Web App

Industry Partner	Brilliant Dentistry
Primary Instructor	Anjana Shah
Team Member	Truong Thi Bui
Team Member	Phuong Hoang
Team Member	Feriel Maamer
Team Member	Robert Kaczur
Team Member	Mahmoud Farghali

1. Introduction

The Introduction section provides an overview of the system using software requirements analysis and design for the scope of the system.

1.1 Purpose

The purpose of this document is to present a detailed description of the Brilliant Dentistry CRM System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the client, Dr. Shiva for approval.

1.2 Scope

Brilliant Dentistry is using an open-source software called Open Dental for data handling and daily operations at the clinic. Brilliant Dentistry is seeking to develop a patient portal to serve as a customer facing interface.

The goal of this project is to create a website where patients can easily access clinic information, contact the clinic, register, book appointment as well as retrieve information about their past visits.

1.2.1 In Scope

- Create a react frontend with responsive design that will have core functionality for the primary stakeholder, employees, and customers.
- Design and deploy a MySQL database using industry best practices on a cloud provider.

- Create a restful flask API to host backend services and allow the frontend to access core business processes, the flask backend would also provide access to the database and authentication.
- Execute different types of testing (unit test, integration test, function test, user acceptance test) to ensure the functions work as per requirements.

1.2.2 Out of Scope

- Marketing and dental specialized content shall be provided by the clinic
- The clinic shall be responsible for maintaining and ensure that the current website domain (<http://www.brilliantdentistry.ca/>) is accessible 24/7. Failure to maintain the website domain that results in service corruption will be out-of-scope for development.
- Clinic staff is responsible for maintain data integrity between Open Dental and Brilliant Dentistry CRM until integration with Open Dental is implemented which will might be discussed in later phase of the project. The proposal for integration between Brilliant Dentistry CRM and Open Dental shall be a separate project which requires extensive effort analysis, design and study the existing Open Dental Software.

2.0 System Overview

The System Overview section introduces the system context and design.

2.1 Project Perspective

The goal of this project is to replace the existing Brilliant Dentistry Website by designing a new CRM web-based system called Brilliant Dentistry CRM.

2.2 System Context

Following are issues with the existing Brilliant Dentistry website:

- No https
- Entire sections are duplicated
- References another unrelated clinic in another country
- Low effort
- Inconsistent styling
- Broken layout
- Random icons
- Booking form doesn't work

The new Brilliant Dentistry CRM system will not only ensure that there are no such issues as the existing website but will also provide design flexibility for future extension, improvement in brand recognition and site experience, increase user traffic and conversion rate.

2.3 General Constraints

- A user familiar with basic browser navigation skills should be able to understand all functionality provided by the system.
- The system should work on most home desktop and laptop computers which support JavaScript and HTML5.
- The system shall be implemented using React front-end, Flask API and MySQL Database.
- Limited access to Open Dental data which is the current software used to manage the clinic which contains confidential information and records about patients.
- No access to confidential medical records of patients due to the Personal Health Information Protection Act which protects the information of clients and mandates that patients consent to have their information disclosed.

- Due to the fact we have no budget, we need to be careful about the services to try and procure so that we don't go over our budget of zero.

2.4 Assumptions and Dependencies

2.4.1 Assumptions

This project makes the following assumptions.

- Development team will have permissions to inspect current database tables without accessing any data.
- Development team will have access to existing documentation that describes business processes.
- Development team have the skills to build a website that is better than the current Brilliant Dentistry website.
- Front desk person will make sure that the schedule on the website is aligned with the schedule in Open Dental and maintain an update-to-date time slot availability
- The clinic will provide the content (text, images, media) to be displayed on the website.
- The website will be provided to the clinic for free of charge and our budget will be approximately zero.
- The new website will be replacing the existing website and hosting on the same domain.
- The clinic is willing to increase the amount they pay for hosting by a small amount due to the nature of creating a web application which tends to cost more than static sites.

2.4.2 Dependencies

The following are the internal and external dependencies that will have to be acknowledged and addressed.

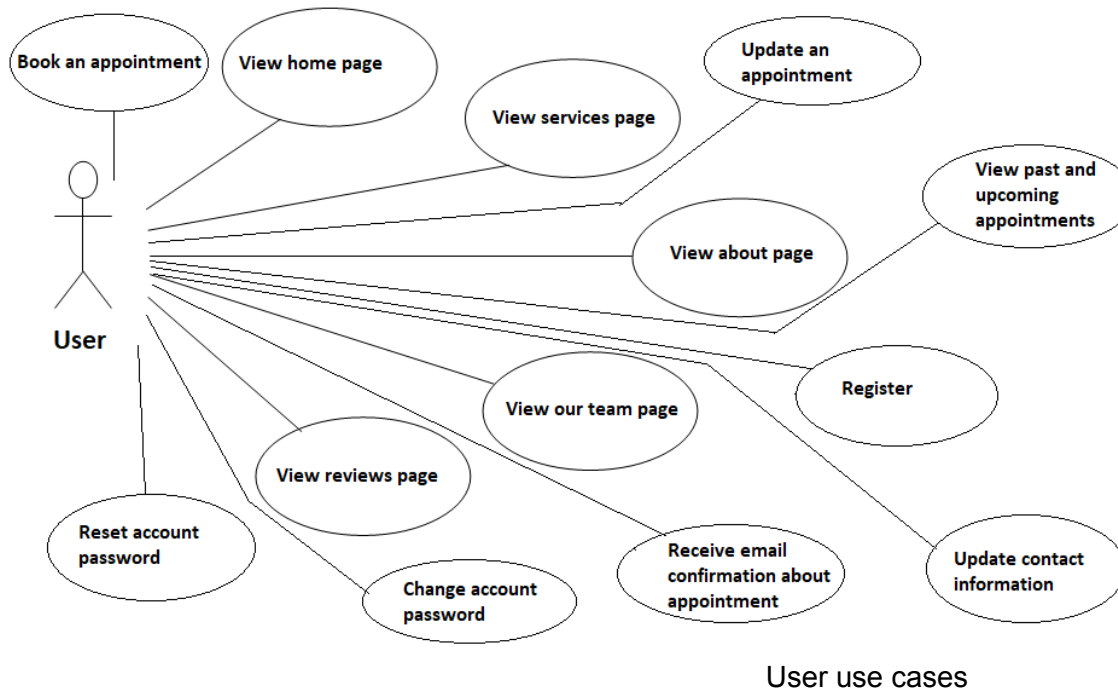
- Access to the brilliant dentistry domain and hosting services.
- Access to mock patient data that represents actual patient data.
- Access to existing Open Dental installation endpoints.

- Business will provide us with information about the services they provide and other relevant content to post on the new web application.

3.0 Functional Requirements Specifications

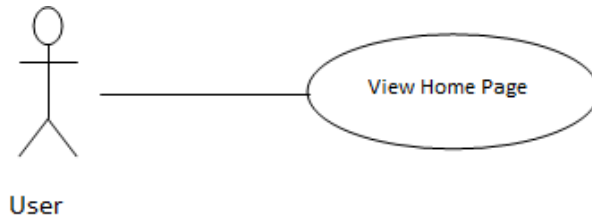
This section describes specific features of the software project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.

3.0.1 User Use Cases



3.0.1.1 Use case: View Home Page

Diagram:



Brief Description

The User accesses the Website, and it loads the Home page automatically which will display a small introduction about the dental clinic as well as descriptive pictures and text fields.

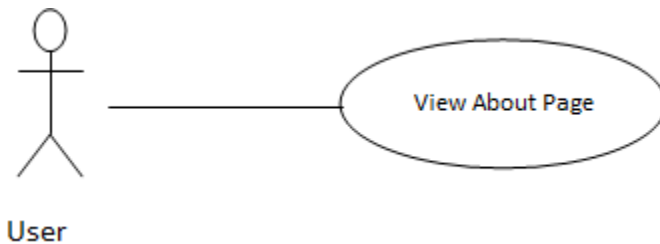
Initial Step-By-Step Description

1. The user accesses the webpage link from an external source

Xref: Section 3.1.1.1, View Home Page

3.0.1.2 Use case: View About Page

Diagram:



Brief Description

The User accesses the About page of the website which contains information about the clinic (for example location) and pictures of the different areas of the clinic. It also has a contact us field that displays the phone number and email address of the clinic.

Initial Step-By-Step Description

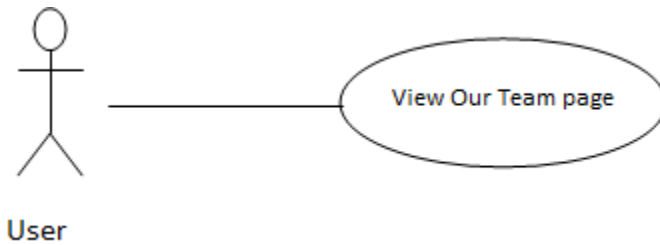
Before this use case can be initiated, the user has already accessed the Website.

1. The user accesses the main page of the website
2. The user clicks on the About tab on the menu
3. The user is redirected to the About page

Xref: Section 3.1.1.2, View About Page

3.0.1.3 Use case: View Our Team Page

Diagram:



Brief Description

The User accesses the Our Team page of the website that displays pictures of the staff members and dentists at the clinic followed by a small introduction of each.

Initial Step-By-Step Description

Before this use case can be initiated, the user has already accessed the Website.

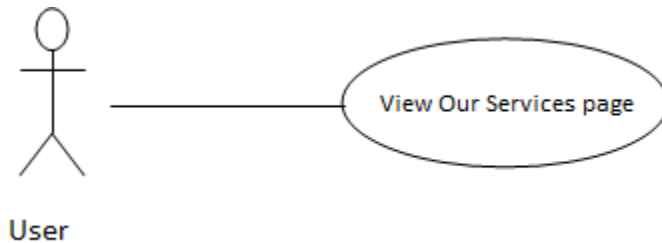
1. The user accesses the main page of the website
2. The user clicks on the Our Team tab on the menu

3. The user is redirected to the Our Team page

Xref: Section 3.1.1.3, View Our Team Page

3.0.1.4 Use case: View Services Page

Diagram:



Brief Description

The User accesses the Services page of the website where information about the different services available at the clinic is provided. The description of the services will allow the patient to understand better the treatment they will be given.

Initial Step-By-Step Description

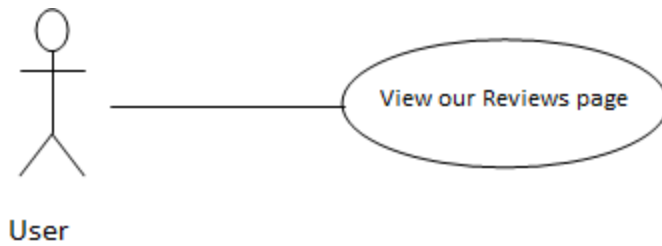
Before this use case can be initiated, the user has already accessed the Website.

1. The user accesses the main page of the website
2. The user clicks on the Services tab on the menu
3. The user is redirected to the About page

Xref: Section 3.1.1.4, View Services Page

3.0.1.5 Use case: View Reviews Page

Diagram:

**Brief Description**

The User accesses the Reviews page of the website where they can see other patients' reviews and opinions about the clinic and their personal experience with the staff members/dentists.

Initial Step-By-Step Description

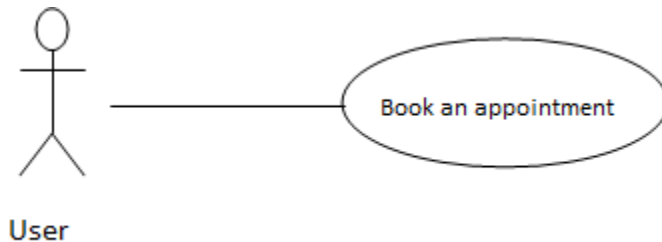
Before this use case can be initiated, the user has already accessed the Website.

1. The user accesses the main page of the website
2. The user clicks on the Reviews tab on the menu
3. The user is redirected to the Reviews page

Xref: Section 3.1.1.5, View Reviews Page

3.0.1.6 Use case: Book an appointment

Diagram:

**Brief Description**

The User should be able to book an appointment online with the dentist. The booking page should display a calendar with the available dates and time slots.

Initial Step-By-Step Description

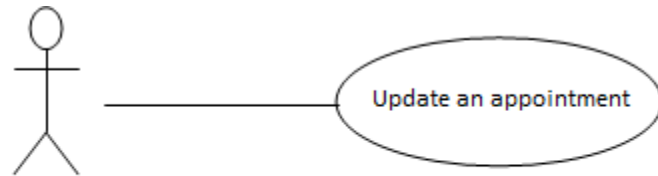
Before this use case can be initiated, the user has already accessed the Website.

1. The user accesses the main page of the website
2. The user clicks on the Booking button on the navigation
3. The user can select from the time and date slots available
4. The user can save the appointment choice

Xref: Section 3.1.1.6, Book an appointment

3.0.1.7 Use case: Update an appointment

Diagram:



Registered User

Brief Description

The User should be able to update an existing appointment online.

Initial Step-By-Step Description

Before this use case can be initiated, the user has already logged in to the Website.

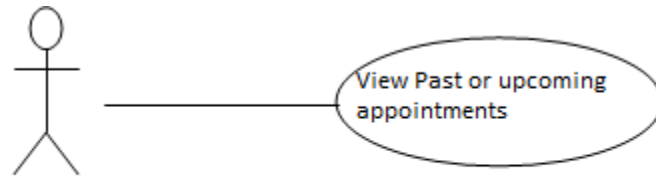
1. The user accesses the main page of the website
2. The user clicks on their profile from the user menu on navigation bar
3. The user clicks on View Booking

4. The user is able to select the appointment to be modified and click on update
5. The user is able to make the changes and save them on the website

Xref: Section 3.1.1.7, Update an appointment

3.0.1.8 Use case: View past or upcoming appointments

Diagram:



Registered User

Brief Description

The User should be able to view past or upcoming appointments online. The page should display a list of different appointments filtered by date.

Initial Step-By-Step Description

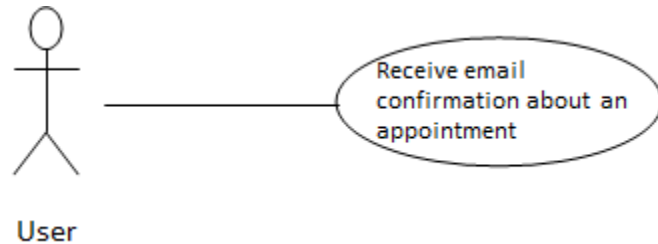
Before this use case can be initiated, the user has already logged in to the Website.

1. The user accesses the main page of the website
2. The user clicks on their profile from the user menu on navigation bar
3. The user clicks on View Booking
4. The user can view past or upcoming appointments in the list and filter based on the date

Xref: Section 3.1.1.8, View past or upcoming appointments

3.0.1.9 Use case: Receive email confirmation about appointment

Diagram:

**Brief Description**

The User should be able to receive an email confirmation after they have booked an appointment on the website.

Initial Step-By-Step Description

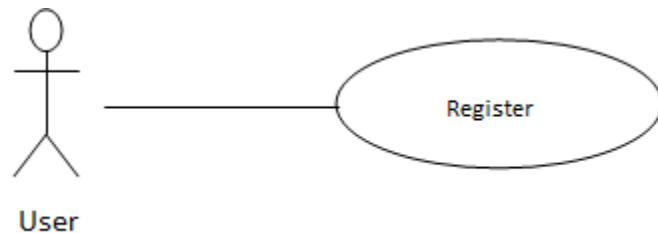
Before this use case can be initiated, the user has already logged in to the Website and booked the appointment.

1. The user receives an email automatically after saving the new appointment booking

Xref: Section 3.1.1.9, Receive email confirmation about appointment

3.0.1.10 Use case: Register

Diagram:

**Brief Description**

The User should be able to register as a patient on the website and the page should display a form where they can enter their personal information.

Initial Step-By-Step Description

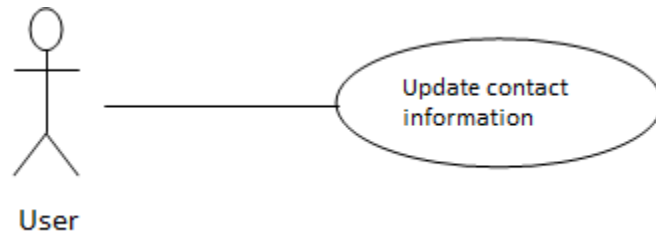
Before this use case can be initiated, the user has already accessed the Website.

1. The user accesses the main page of the website
2. The user clicks on the Register option on the menu
3. The user can enter their personal information and create an account

Xref: Section 3.1.1.10, Register

3.0.1.11 Use case: Update contact information

Diagram:

**Brief Description**

The User should be able to update their contact information on the website.

Initial Step-By-Step Description

Before this use case can be initiated, the user has already logged in to the Website.

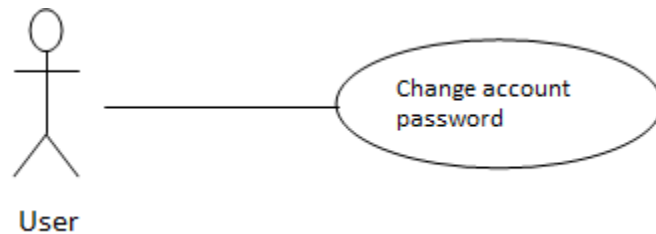
1. The user accesses the main page of the website
2. The user clicks on their profile from the user menu on navigation bar
3. The user clicks on update personal information

4. The user can make the changes to the contact information and save

Xref: Section 3.1.1.11, Update contact information

3.0.1.12 Use case: Change account password

Diagram:



Brief Description

The User should be able to change their account password on the website.

Initial Step-By-Step Description

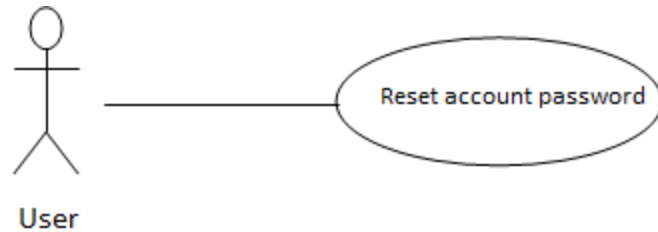
Before this use case can be initiated, the user has already logged in to the Website.

1. The user accesses the main page of the website
2. The user clicks on their profile from the user menu on the navigation bar
3. The user clicks on the change password option
4. The user can change the password and save

Xref: Section 3.1.1.12, Change account password

3.0.1.13 Use case: Reset account password

Diagram:

**Brief Description**

The User should be able to reset their account password on the website in case they forget it.

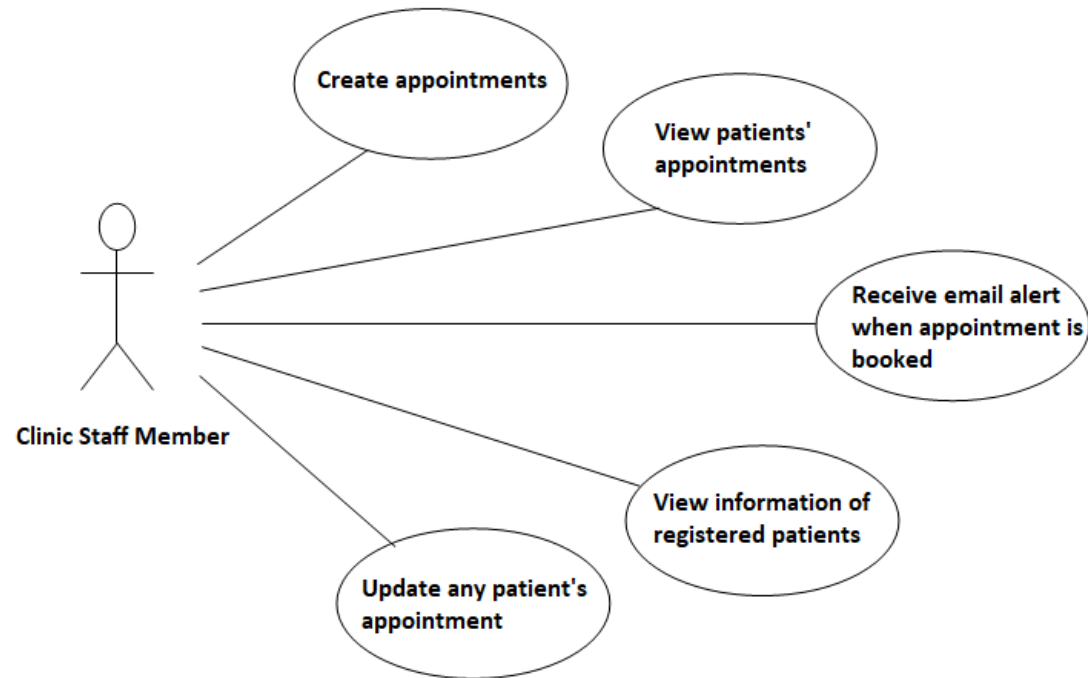
Initial Step-By-Step Description

Before this use case can be initiated, the user has already accessed the Website.

1. The user accesses the main page of the website
2. The user clicks on the login option on the menu
3. The user clicks on forgot password
4. The user should receive an automatic email with a link to reset their password

Xref: Section 3.1.1.13, Reset account password

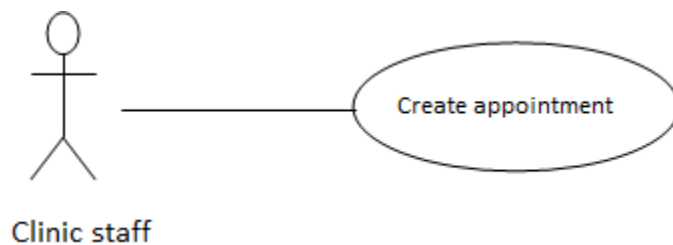
3.0.2 Clinic Staff Use Cases



Clinic Staff use cases

3.0.2.1 Use case: Create appointment

Diagram:



Brief Description

The staff member should be able to create an appointment for a patient from the website.

Initial Step-By-Step Description

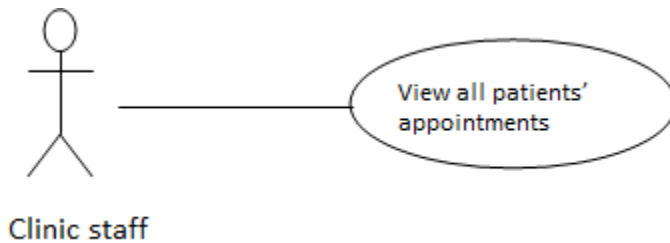
Before this use case can be initiated, the staff member has accessed the Website.

1. The staff member accesses the main page of the website
2. The staff member clicks on the Booking button on the navigation
3. The staff member can enter the patient details for the appointment
4. The staff member can select from the time and date slots available
5. The staff member can save the appointment choice

Xref: Section 3.1.2.1, Create appointment

3.0.2.2 Use case: View all patients' appointments

Diagram:

**Brief Description**

The staff member should be able to view all patients' appointments from the website.

Initial Step-By-Step Description

Before this use case can be initiated, the staff member has already logged in to the Website.

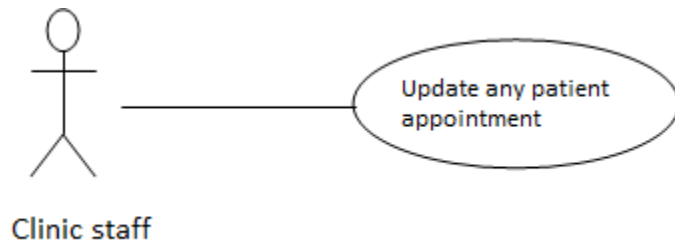
1. The staff member accesses the main page of the website

2. The staff member clicks on the Booking button on the navigation
3. The staff member can view all patients' previous and upcoming appointments and filter by patient name or date.

Xref: Section 3.1.2.2, View all patients' appointments

3.0.2.3 Use case: Update any patient appointment

Diagram:



Brief Description

The staff member should be able to update an appointment for a patient from the website.

Initial Step-By-Step Description

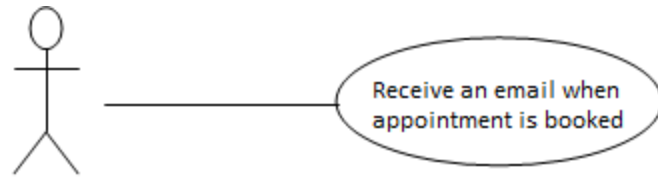
Before this use case can be initiated, the staff member has already logged in to the Website.

1. The staff member accesses the main page of the website
2. The staff member clicks on the Booking button on the navigation
3. The staff member can search for appointments by patient
4. The staff member can select the appointment to make changes to and click on update
5. The staff member can save the appointment changes

Xref: Section 3.1.2.3, Update any patient appointment

3.0.2.4 Use case: Receive an email when an appointment is booked

Diagram:



Clinic staff

Brief Description

The staff member should be alerted to an appointment being booked via email.

Initial Step-By-Step Description

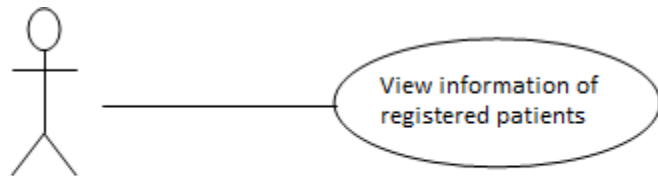
No pre-requirements for this use case.

1. The staff receives an email when an appointment is booked online.

Xref: Section 3.1.2.4, Receive an email when an appointment is booked

3.0.2.5 Use case: View the information of the registered patients

Diagram:



Clinic staff

Brief Description

The staff member should be able to view the information of the registered patients from the website.

Initial Step-By-Step Description

Before this use case can be initiated, the staff member has already logged in to the Website.

1. The staff member accesses the main page of the website
2. The staff member clicks on the Patients tab on the menu
3. The staff member can search by patient and view each patient's information on the website

Xref: Section 3.1.2.5, View the information of the registered patients

3.1 Use Cases

3.1.1 User use case

3.1.1.1 View Home page

Use Case Name	Users must be able to view Home Page
XRef	Section 3.0.1.1, View Home page
Trigger	The user accesses the website
Precondition	The user is accessing the website from an external source
Basic Path	User clicks on the external link that redirects them to the home page of the dental clinic website
Alternative Paths	If the user prefers, they can click on the website's icon or the home tab
Postcondition	The user is redirected to the home page
Exception Paths	The attempt may be abandoned at any time.

3.1.1.2 View About page

Use Case Name	Users must be able to view About Page
XRef	Section 3.0.1.2, View About Page
Trigger	The user clicks on the About tab
Precondition	The user is on the website already
Basic Path	1. The user accesses the main page of the website 2. The user clicks on the About tab on the menu 3. The user is redirected to the About page
Alternative Paths	None
Postcondition	The user is redirected to the About page
Exception Paths	The attempt may be abandoned at any time.

3.1.1.3 View Our Team page

Use Case Name	Users must be able to view Our Team Page
XRef	Section 3.0.1.3, View Our Team Page
Trigger	The user clicks on the Our Team tab
Precondition	The user is on the website already
Basic Path	1. The user accesses the main page of the website 2. The user clicks on the Our Team tab on the menu 3. The user is redirected to the Our Team page

Alternative Paths	None
Postcondition	The user is redirected to the Our Team page
Exception Paths	The attempt may be abandoned at any time.

3.1.1.4 View Services page

Use Case Name	Users must be able to view Services Page
XRef	Section 3.0.1.4, View Services Page
Trigger	The user clicks on the Services tab
Precondition	The user is on the website already
Basic Path	1. The user accesses the main page of the website 2. The user clicks on the Services tab on the menu 3. The user is redirected to the Services page
Alternative Paths	None
Postcondition	The user is redirected to the Services page
Exception Paths	The attempt may be abandoned at any time.

3.1.1.5 View Reviews page

Use Case Name	Users must be able to view Reviews Page
XRef	Section 3.0.1.5, View Reviews Page

Trigger	The user clicks on the Reviews tab
Precondition	The user is on the website already
Basic Path	1. The user accesses the main page of the website 2. The user clicks on the Reviews tab on the menu 3. The user is redirected to the Reviews page
Alternative Paths	None
Postcondition	The user is redirected to the Reviews page
Exception Paths	The attempt may be abandoned at any time.

3.1.1.6 Book an appointment

Use Case Name	Users must be able to book an appointment
XRef	Section 3.0.1.6, Book an appointment
Trigger	The user clicks on the booking button from the navigation bar
Precondition	The user is on the booking page
Basic Path	1. The user accesses the main page of the website 2. The user clicks on the Booking button on the navigation 3. The user is presented with a form to fill in the first name, last name, email address and phone number 4. Once the previous information is saved, the system presents a calendar grid that show the available dates to book an appointment 5. Once the day is selected the system shows the available time slots 6. The user can save the appointment information The appointment information is saved to the database as a new entry

Alternative Paths	If in step 3, either field is blank, the User is instructed to add an entry.
Postcondition	The booking is made and saved on the system
Exception Paths	The attempt may be abandoned at any time.

3.1.1.7 Update appointments

Use Case Name	Registered Users must be able to update their appointment
XRef	Section 3.0.1.7, Update an appointment
Trigger	The user clicks on the “Book or Update an appointment” link
Precondition	The user is on logged in to the website
Basic Path	<ol style="list-style-type: none">1. The user accesses the main page of the website2. If it's an existing patient, the user proceeds to log in with their username and password3. The system redirects them back to the home page4. The user clicks on their profile from the user menu on navigation bar5. The user clicks on View Booking6. The system presents the bookings in a list form for modification.7. The user chooses which appointment to edit updates the information and submits the form.8. The system checks that required fields are not blank.
Alternative Paths	In step 7, if any required field is blank, the Editor is instructed to add an entry.
Postcondition	The database has been updated with the new booking details.
Exception Paths	The attempt may be abandoned at any time.

3.1.1.8 View past or upcoming appointments

Use Case Name	Registered Users must be able to view their past or upcoming appointment
XRef	Section 3.0.1.8, View past or upcoming appointments
Trigger	The user clicks on the “Book or Update an appointment” link
Precondition	The user is logged in to the website
Basic Path	<ol style="list-style-type: none">1. The user accesses the main page of the website2. If it's an existing patient, the user proceeds to log in with their username and password3. The system redirects them back to the home page4. The user clicks on their profile from the user menu on navigation bar5. The user clicks on View Booking6. The system presents the bookings in a list form filtered by date.
Alternative Paths	None
Postcondition	The user is redirected to a list of the existing appointments ordered by date
Exception Paths	The attempt may be abandoned at any time.

3.1.1.9 Receive email confirmation about appointment

Use Case Name	Users must be able to receive email confirmation about their appointment
XRef	Section 3.0.1.9, Receive email confirmation about appointment
Trigger	The user books or updates an appointment
Precondition	The user books or updates an appointment

Basic Path	This use case will automatically send the user an email with a confirmation and details about the appointment
Alternative Paths	None
Postcondition	The user receives an email confirmation
Exception Paths	None

3.1.1.10 Registration

Use Case Name	Users must be able to register
XRef	Section 3.0.1.10, Register
Trigger	The user clicks on the Register option on the menu
Precondition	The user is on the website already and is not a registered user
Basic Path	<ol style="list-style-type: none">1. The user accesses the main page of the website2. The user clicks on the Register option on the menu3. The system presents a blank grid to enter the user information: first name, last name, email address, date of birth, address...4. The user enters the information and submits the form.5. The system checks that none of the fields are blank and meets the requirements and updates the database.6. The user is redirected to the home page
Alternative Paths	In step 5, if any required field is blank, the user is instructed to add an entry. No validation for correctness is made.

Postcondition	The user's account is created, and the information is saved on the database
Exception Paths	The attempt may be abandoned at any time.

3.1.1.11 Update contact information

Use Case Name	Users must be able to update their contact information
XRef	Section 3.0.1.11, Update contact information
Trigger	The user clicks on update personal information from their user profile page
Precondition	The user is logged in already
Basic Path	<ol style="list-style-type: none">1. The user accesses the main page of the website2. The user clicks on Profile from the user menu on the navigation bar3. The system shows all user's personal information retrieved from the database4. The user clicks on update personal information5. The page displays a grid form where the user is able to delete any entry previously saved on the system6. The user can make the changes to the contact information and save7. The changes are updated on the database
Alternative Paths	If in step 6, either field does not meet the requirements or is empty, the user is instructed to add/change the entry.
Postcondition	The user can see and update their contact information
Exception Paths	The attempt may be abandoned at any time.

3.1.1.12 Change account password

Use Case Name	Users must be able to change their account password
XRef	Section 3.0.1.12, Change account password
Trigger	The user clicks on the Account tab then
Precondition	The user is logged in already
Basic Path	<ol style="list-style-type: none">1. The user accesses the main page of the website2. The user clicks on the user menu on the navigation bar3. The user clicks on Change Password from the menu4. The system displays a form where the user must enter their old password, new password, and confirmation of the new password.5. The user saves the new password and if it meets all requirements, it redirects the user to the login page6. The new password is updated on the database
Alternative Paths	In step 4, if the password entered does not meet the requirements the user is asked to enter a different password
Postcondition	The password is changed on the database
Exception Paths	The attempt may be abandoned at any time.

3.1.1.13 Reset account password

Use Case Name	Users must be able to reset their account password
XRef	Section 3.0.1.13, Reset account password

Trigger	The user accesses the website
Precondition	The user is on the website already
Basic Path	The user accesses the main page of the website The user clicks on the login option on the menu The user clicks on forgot password The user should receive an automatic email with a temporary password and a link to reset their password The link takes the user to the change password page
Alternative Paths	None
Postcondition	The user can see the content of the Services page
Exception Paths	The attempt may be abandoned at any time.

3.1.2 Clinic Staff use cases

3.1.2.1 Create appointment

Use Case Name	Clinic staff must be able to create appointment
XRef	Section 3.0.2.1, Create appointment
Trigger	The Clinic staff clicks on the booking button from the home page
Precondition	The clinic staff is logged in on the website already
Basic Path	1. The staff member accesses the main page of the website 2. The staff member clicks on the Booking button on the navigation

	<ol style="list-style-type: none"> 3. The staff member is presented with a form to fill in the patient's first name, last name, email address and phone number 4. Once the previous information is saved, the system presents a calendar grid that show the available dates to book an appointment 5. Once the day is selected the system shows the available time slots 6. The user can save the appointment information 7. The appointment information is saved to the database as a new entry
Alternative Paths	If in step 3, either field is blank, the staff member is instructed to add an entry.
Postcondition	The new appointment for the patient is created and saved in the database
Exception Paths	The attempt may be abandoned at any time.

3.1.2.2 View all patients' appointments

Use Case Name	Clinic staff must be able to view all patients' appointment
XRef	Section 3.0.2.2, View all patients' appointments
Trigger	The staff member clicks on the booking button from the home page
Precondition	The staff member is logged in on the website
Basic Path	<ol style="list-style-type: none"> 1. The staff member accesses the main page of the website 2. The staff member clicks on the Booking button on the navigation 3. The system displays all patients' previous and upcoming appointments in the form of a grid with the option to filter by patient name or date.
Alternative Paths	None
Postcondition	The staff member is redirected to the patients' appointments page

Exception Paths	The attempt may be abandoned at any time.
------------------------	---

3.1.2.3 Update any patient appointment

Use Case Name	Clinic staff must be able to update any patient appointment
XRef	Section 3.0.2.3, Update any patient appointment
Trigger	The staff member clicks on the update button next to the appointment that needs to be modified
Precondition	The staff member is logged in to the website already
Basic Path	<ol style="list-style-type: none">1. The staff member accesses the main page of the website2. The staff member clicks on the Booking button on the navigation3. The staff member can search for appointments by patient4. The staff member can select the appointment to make changes to and click on update5. The system displays a calendar with the date slots available6. After confirming the date, the system displays the time slots available on that date7. The staff member can save the appointment changes
Alternative Paths	None
Postcondition	The updated appointment is saved to the database
Exception Paths	The attempt may be abandoned at any time.

3.1.2.4 Receive email confirmation when an appointment is booked

Use Case Name	Clinic staff must be alerted to an appointment being booked via email
XRef	Section 3.0.2.4, Receive email confirmation when an appointment is booked

Trigger	The user books or updates an appointment
Precondition	The user books or updates an appointment
Basic Path	This use case will automatically send the staff member an email with a confirmation and details about the appointment
Alternative Paths	None
Postcondition	The staff member receives an email with details about the appointment made by the user
Exception Paths	None

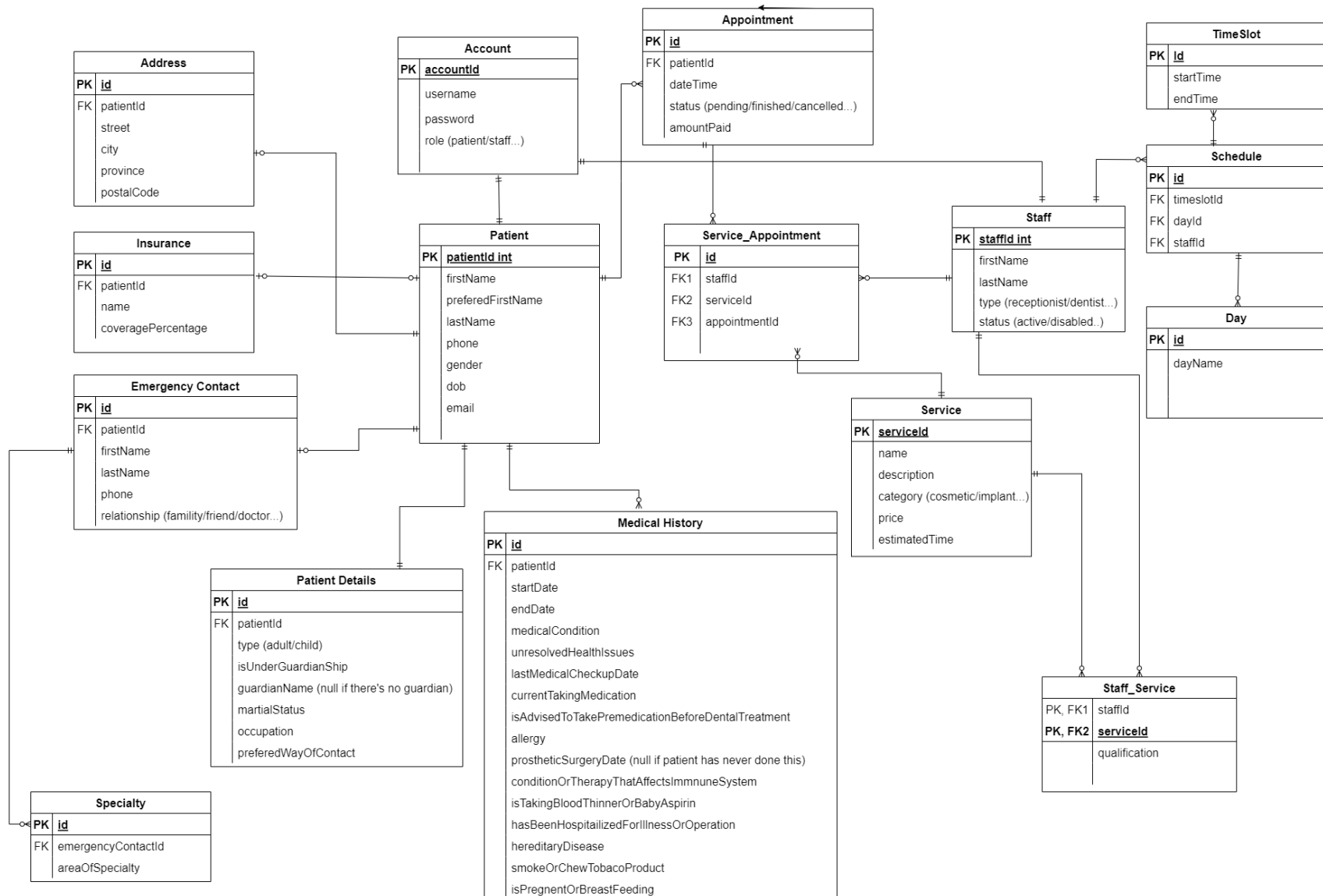
3.1.2.5 View information of registered patients

Use Case Name	Clinic staff must be able to view the information of the registered patients
XRef	Section 3.0.2.5, View information of registered patients
Trigger	The staff member access the Patient's page
Precondition	The staff member is logged in to the website
Basic Path	<ol style="list-style-type: none">1. The staff member accesses the main page of the website2. The staff member clicks on patients tab from the navigation bar3. The system shows a patients list ordered by alphabetical order4. Once the staff member clicks on a specific patient, the system redirects them to that specific patient's page where it displays their personal information retrieved from the database
Alternative Paths	None

Postcondition	The staff member is redirected to a specific patient's page that display personal information
Exception Paths	The attempt may be abandoned at any time.

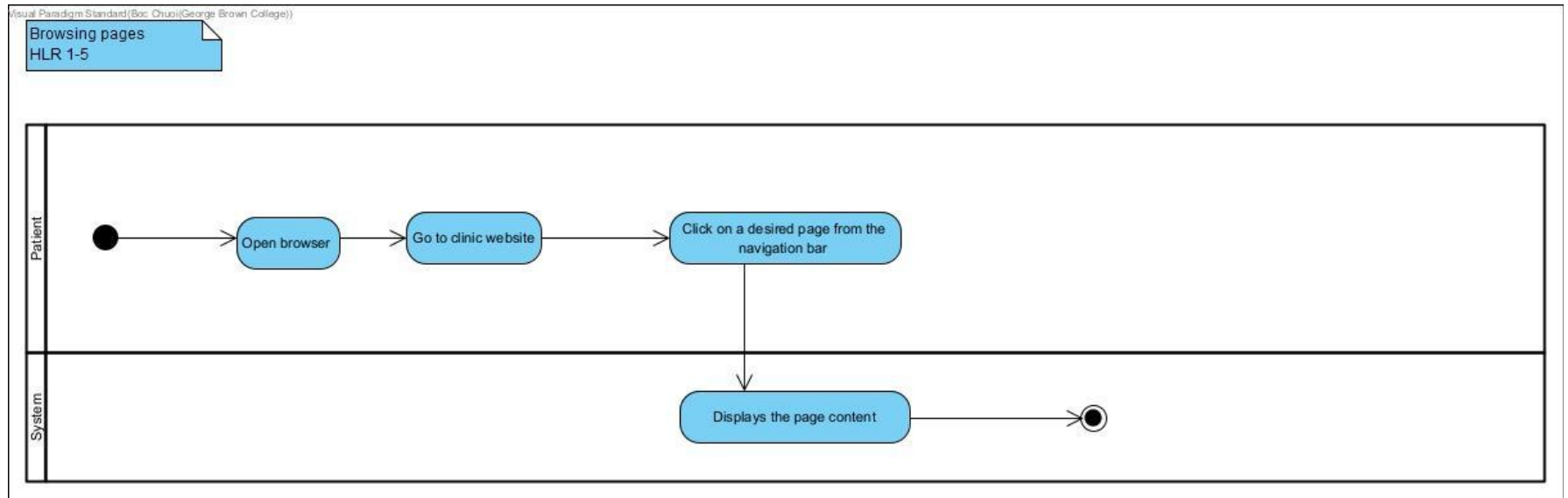
3.2 Data Modelling and Analysis

3.2.1 Normalized Data Model Diagram (see next page)

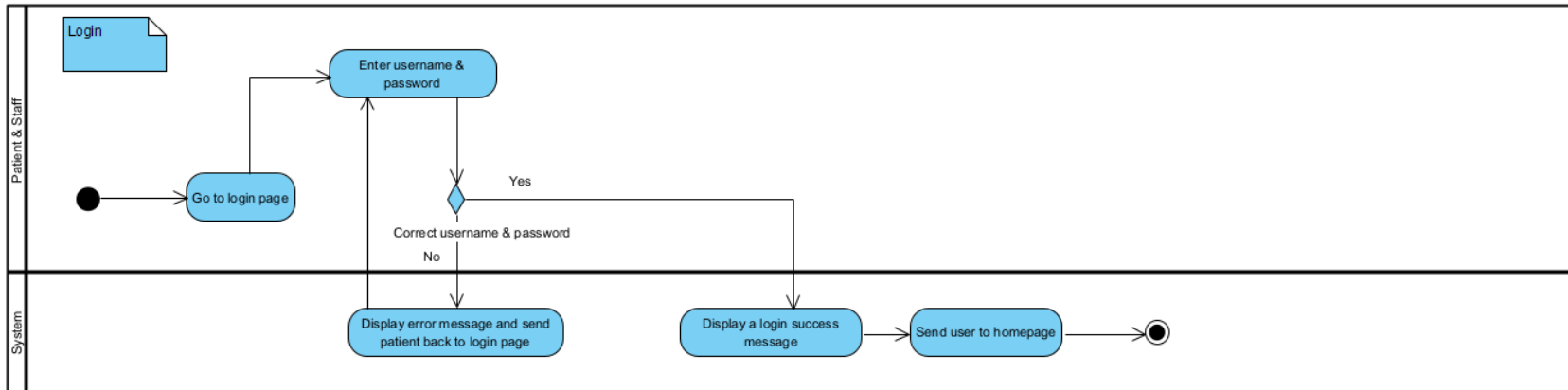


3.2.2 Activity Diagrams

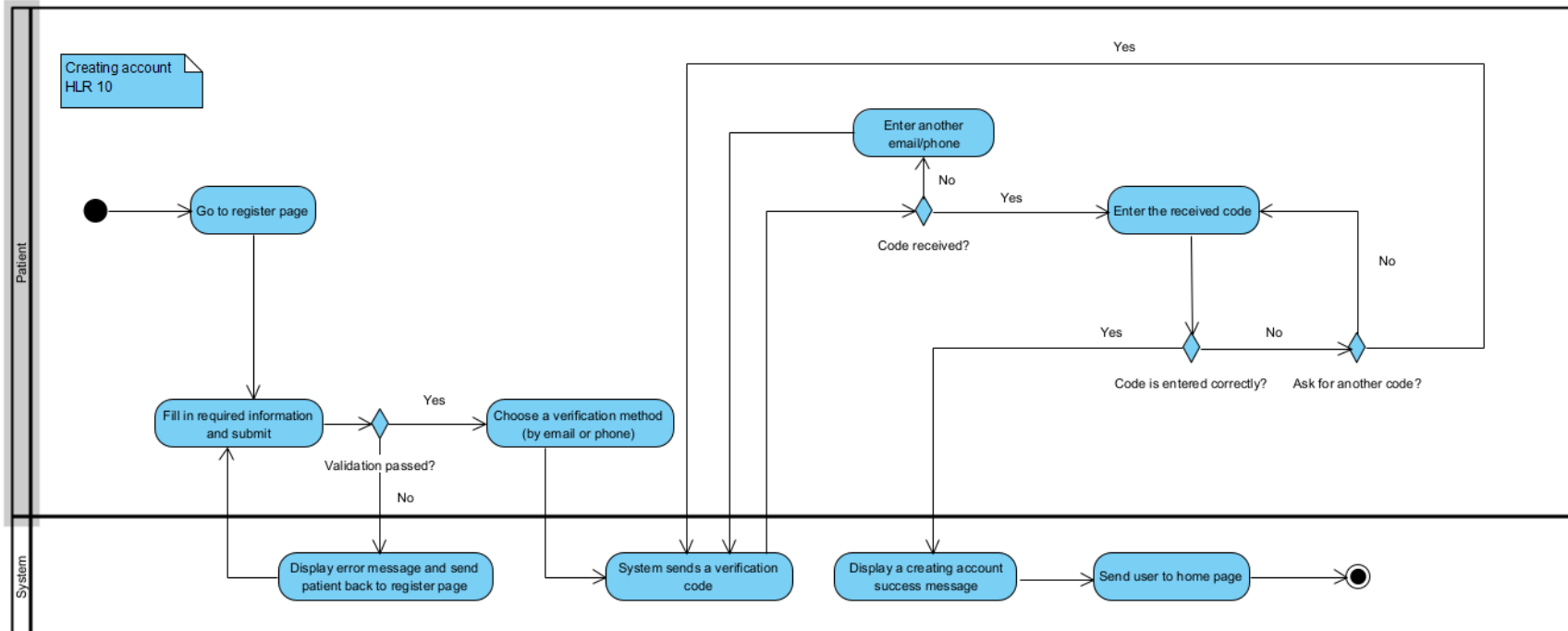
3.2.2.1 Browse Website



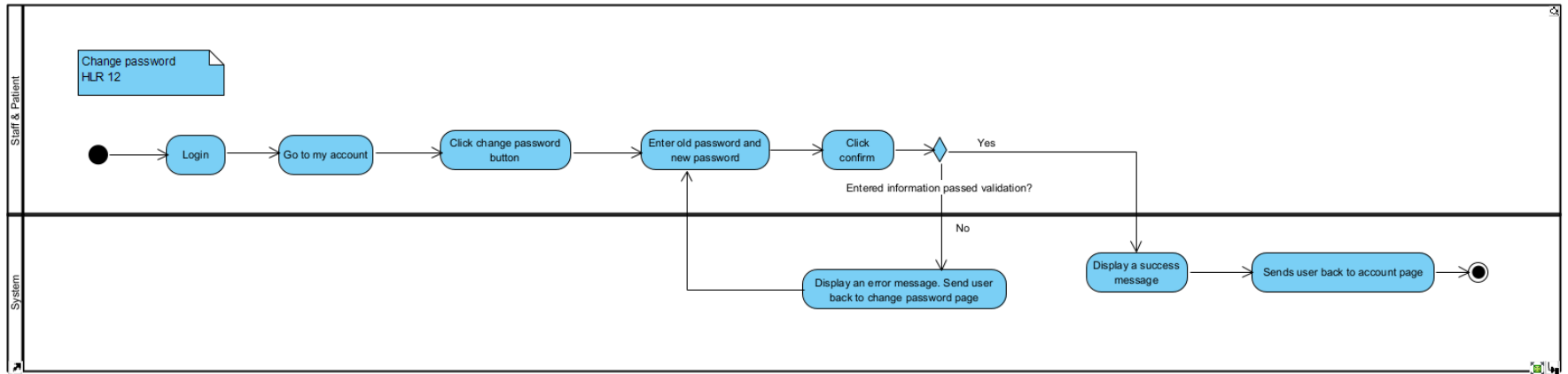
3.2.2.2 Login



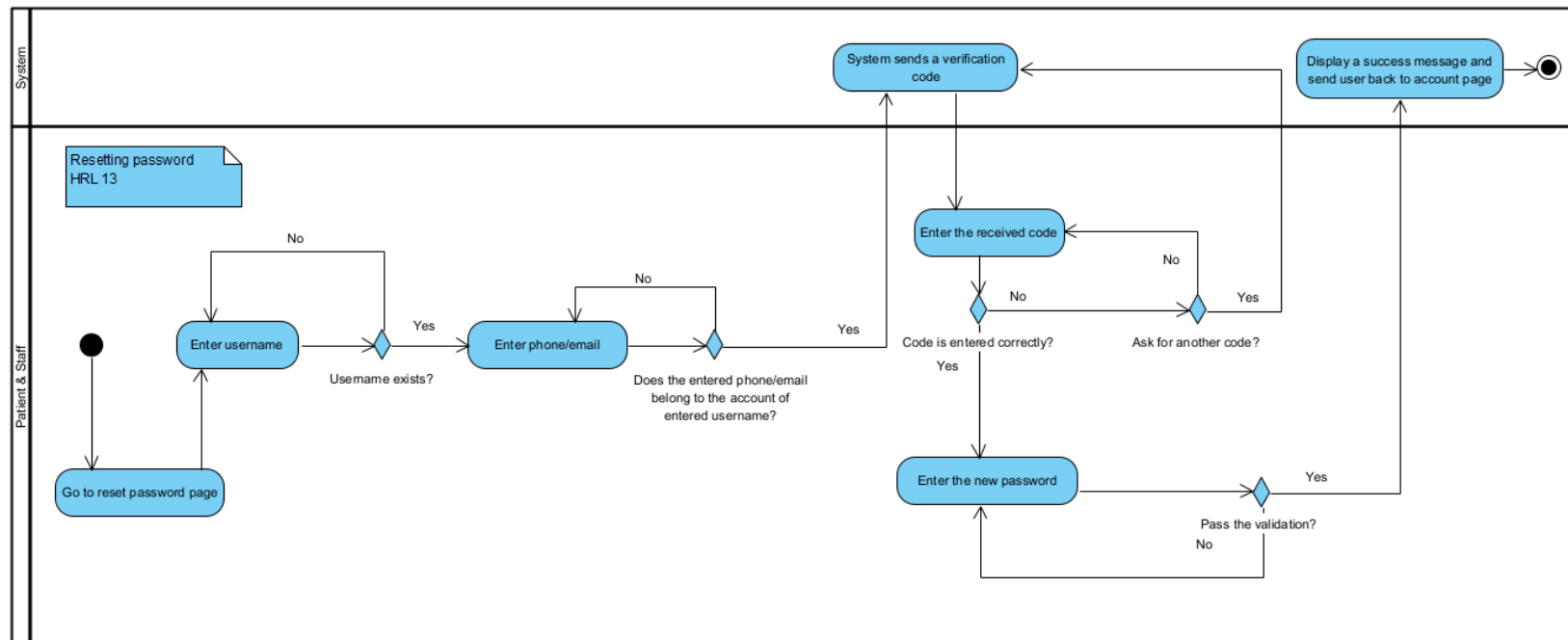
3.2.2.3 Registration



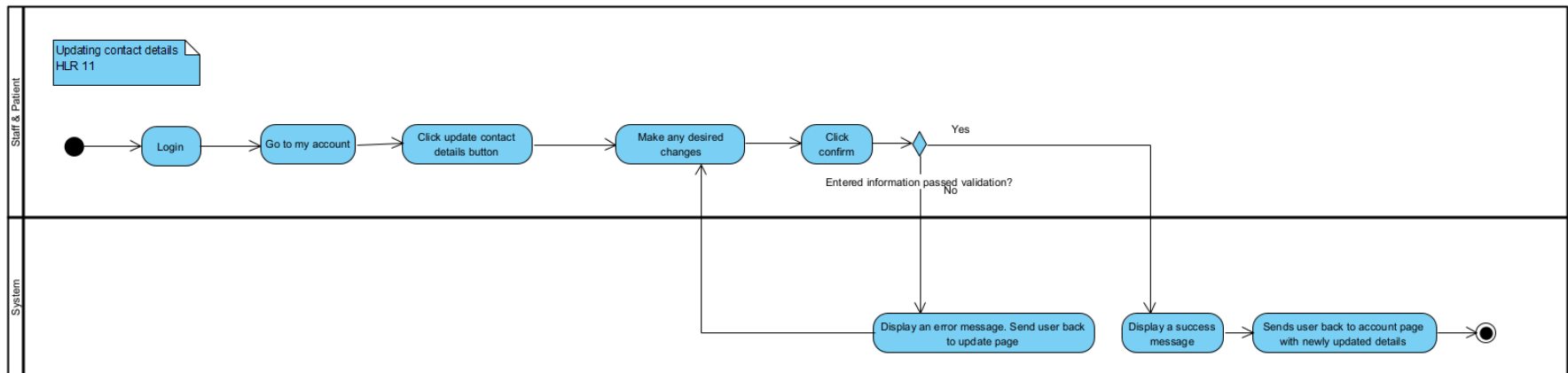
3.2.2.4 Change Password



3.2.2.5 Reset Password

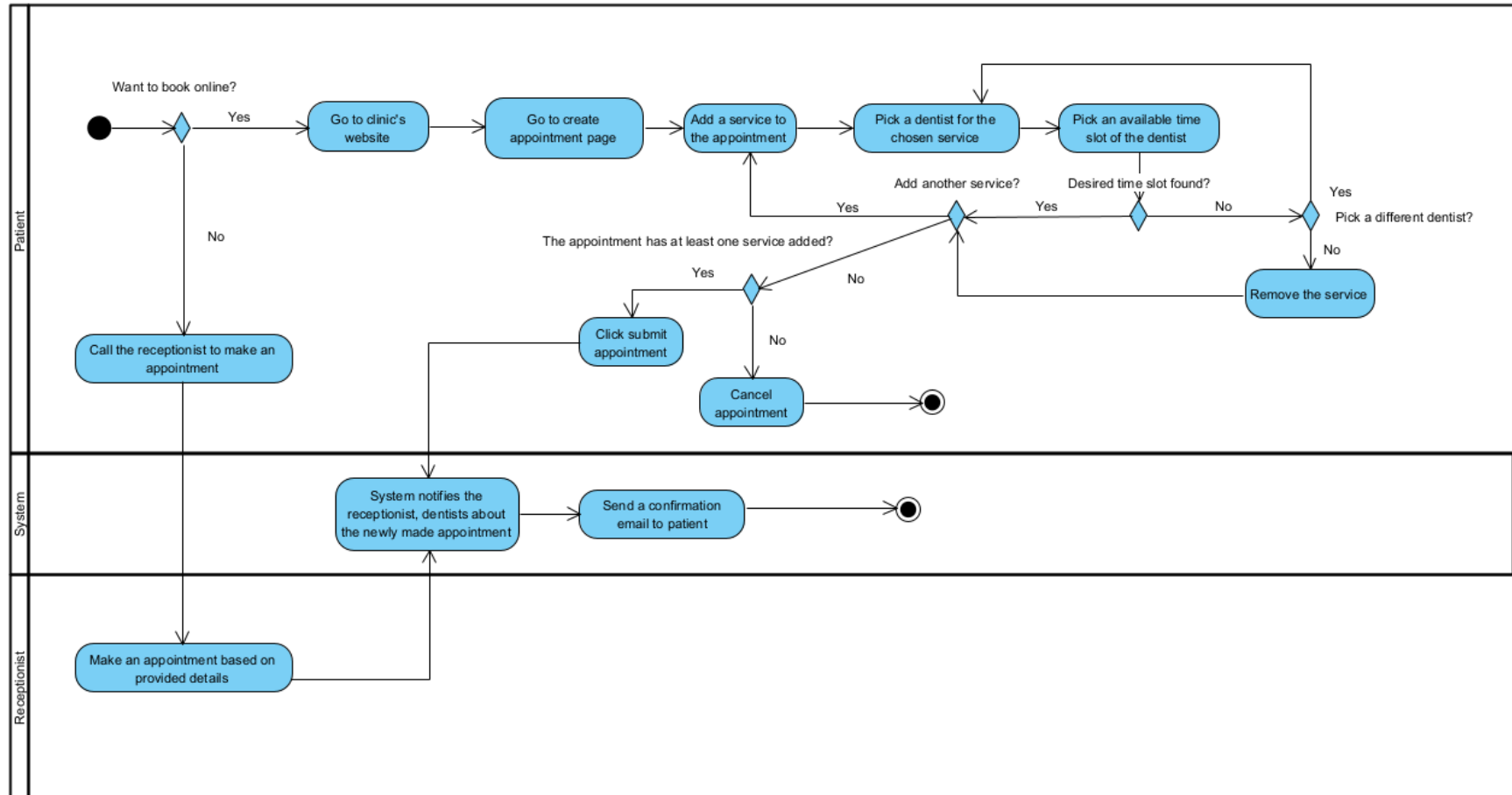


3.2.2.6 Update Profile



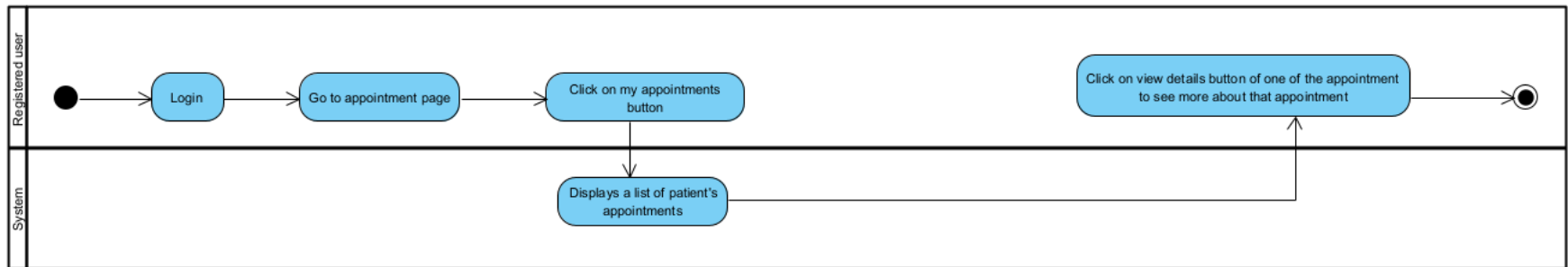
3.2.2.7 Patient Creates Appointment

Making appointment
HLR 06, 09, 17

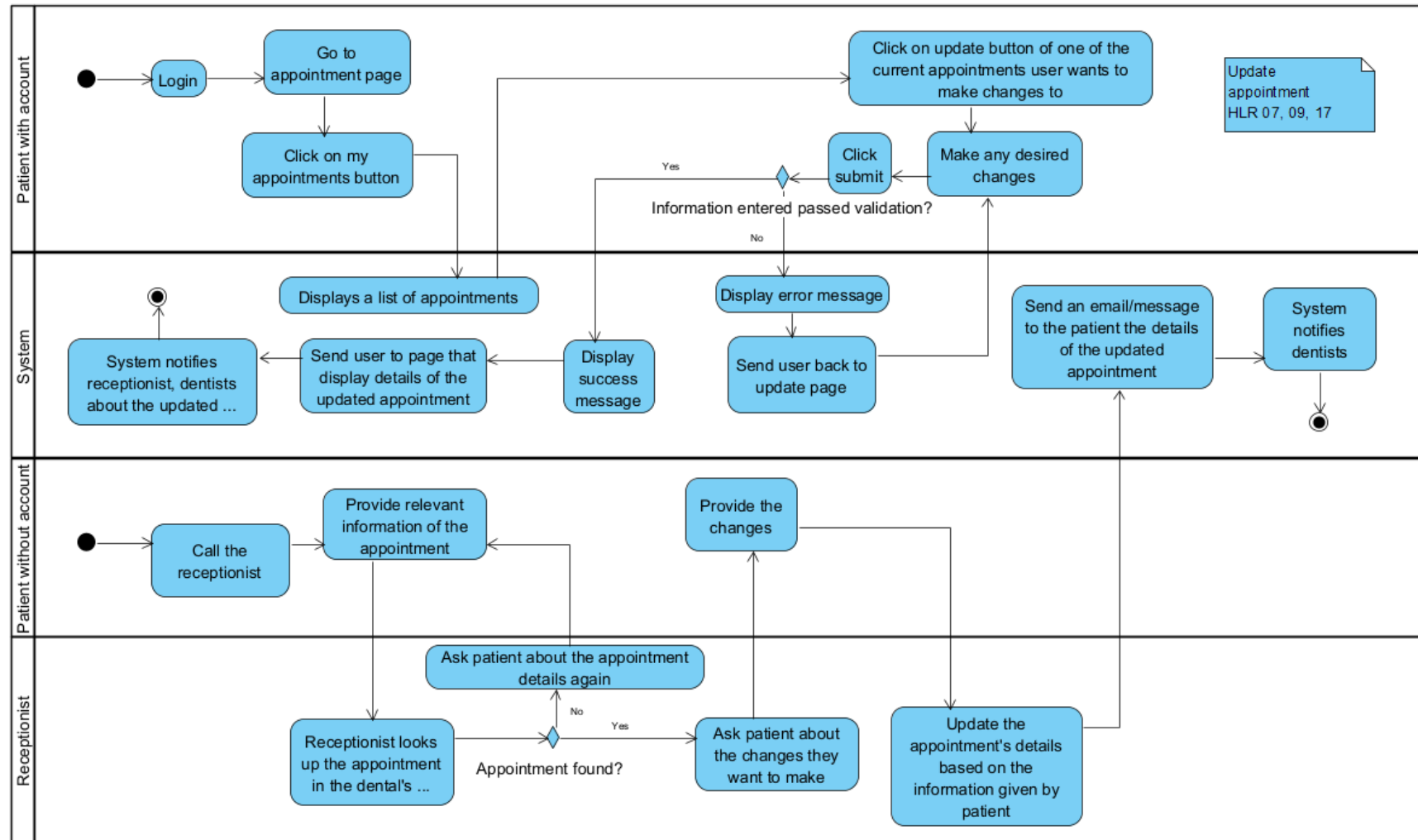


3.2.2.8 Patient Views Appointment

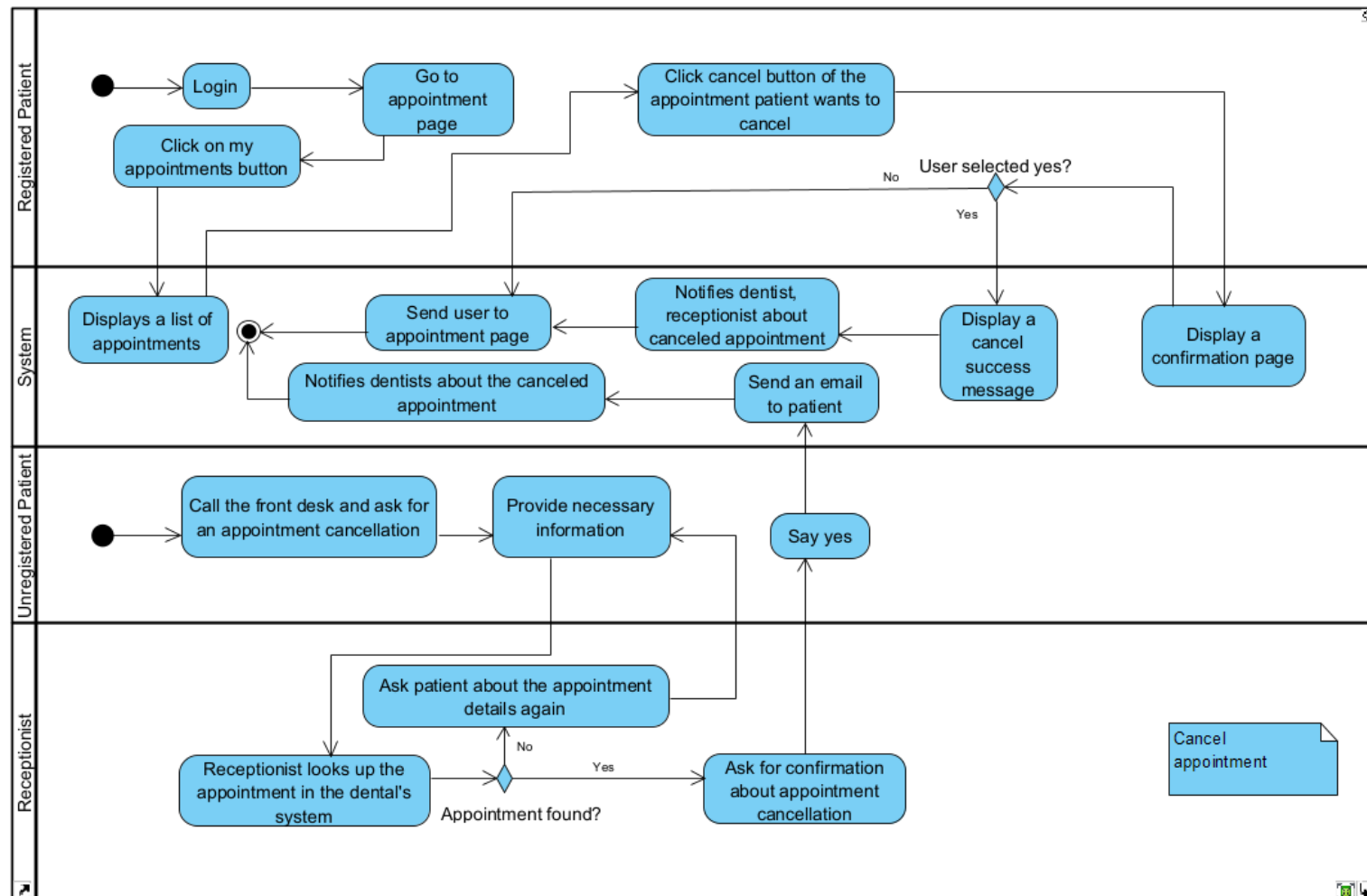
Registered patients view their appointment
HLR 08



3.2.2.9 Patient Updates Appointment

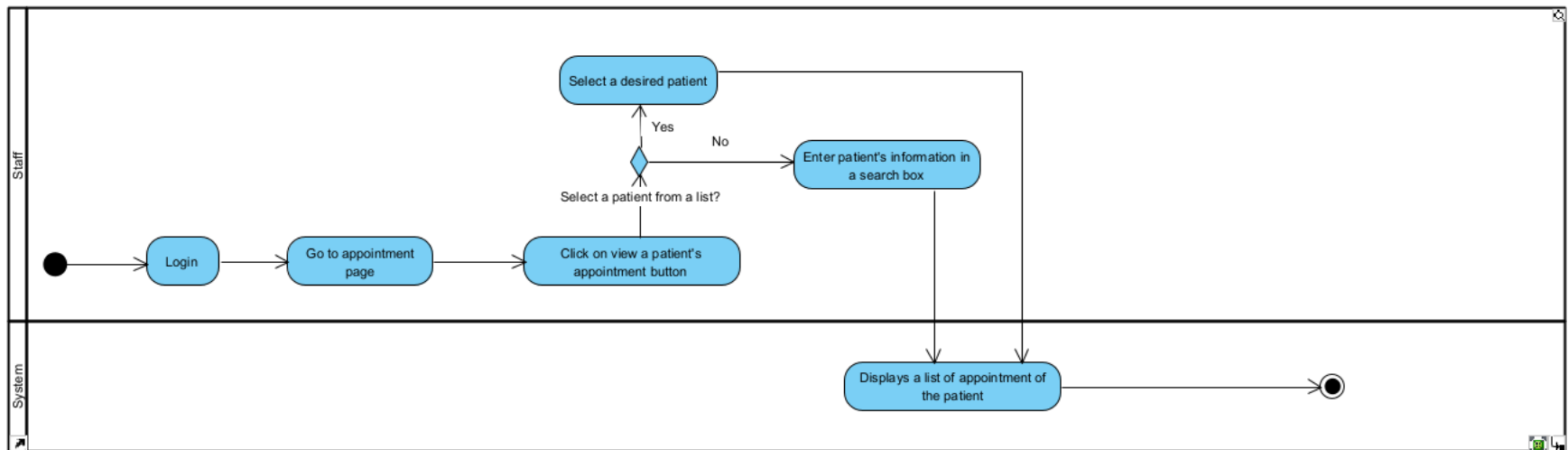


3.2.2.10 Patient Cancels Appointment



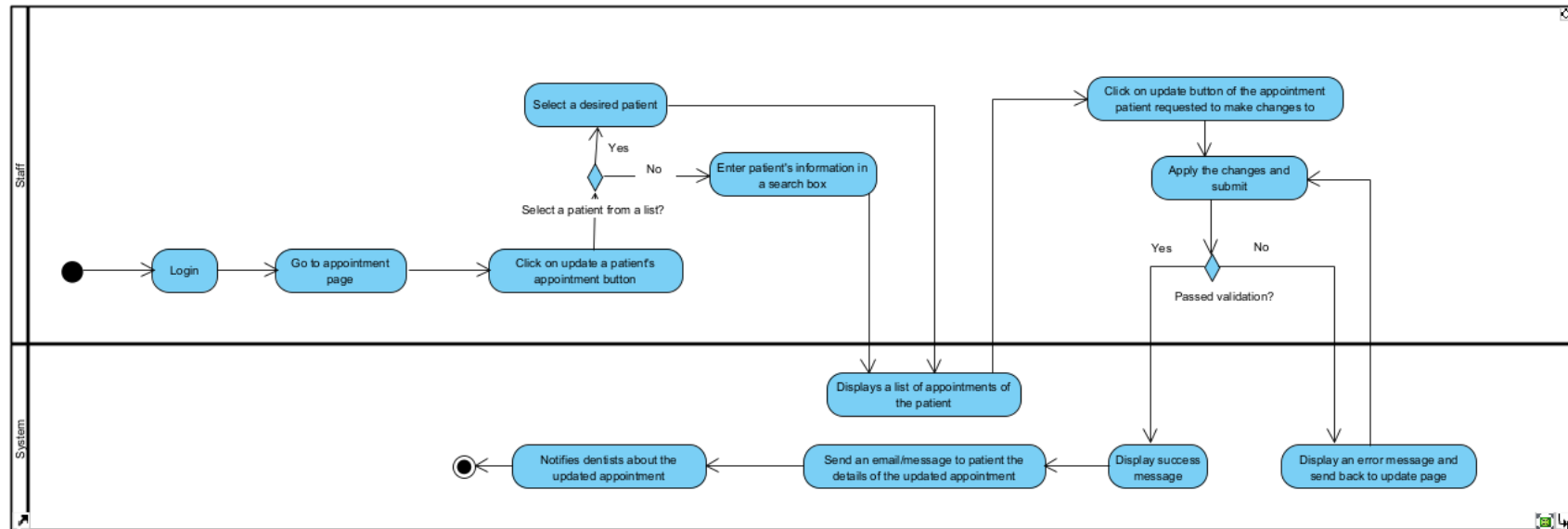
3.2.2.11 Staff Views Patients' Appointments

Staff viewing patient's
appointment
HLR 15



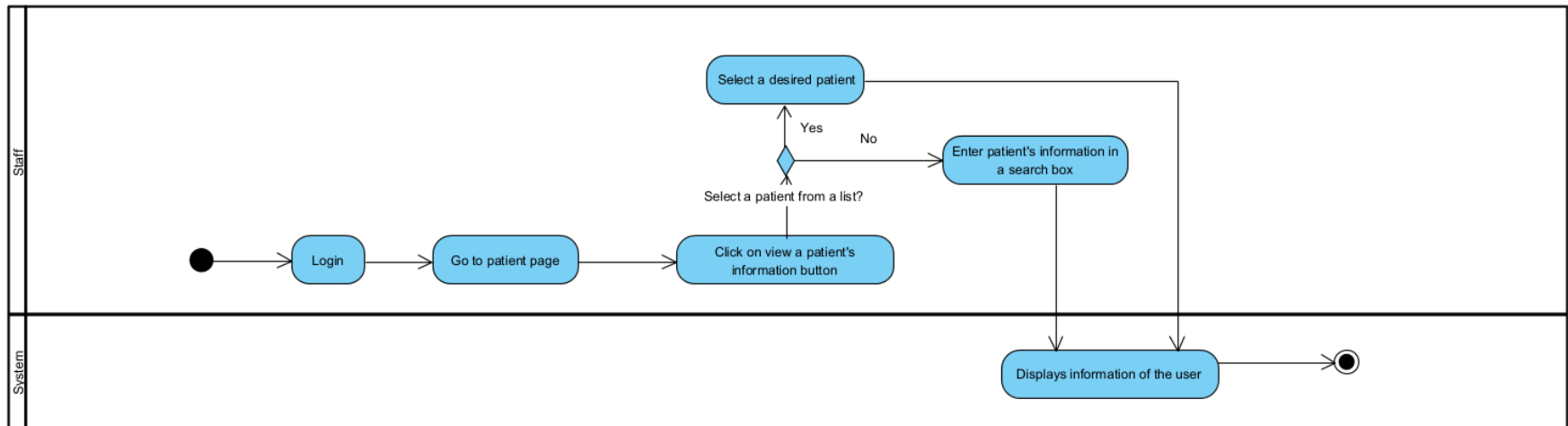
3.2.2.12 Staff Updates Patients' Appointment

Staff update a patients
appointment
HLR 16



3.2.2.13 Staff Views Patient Information

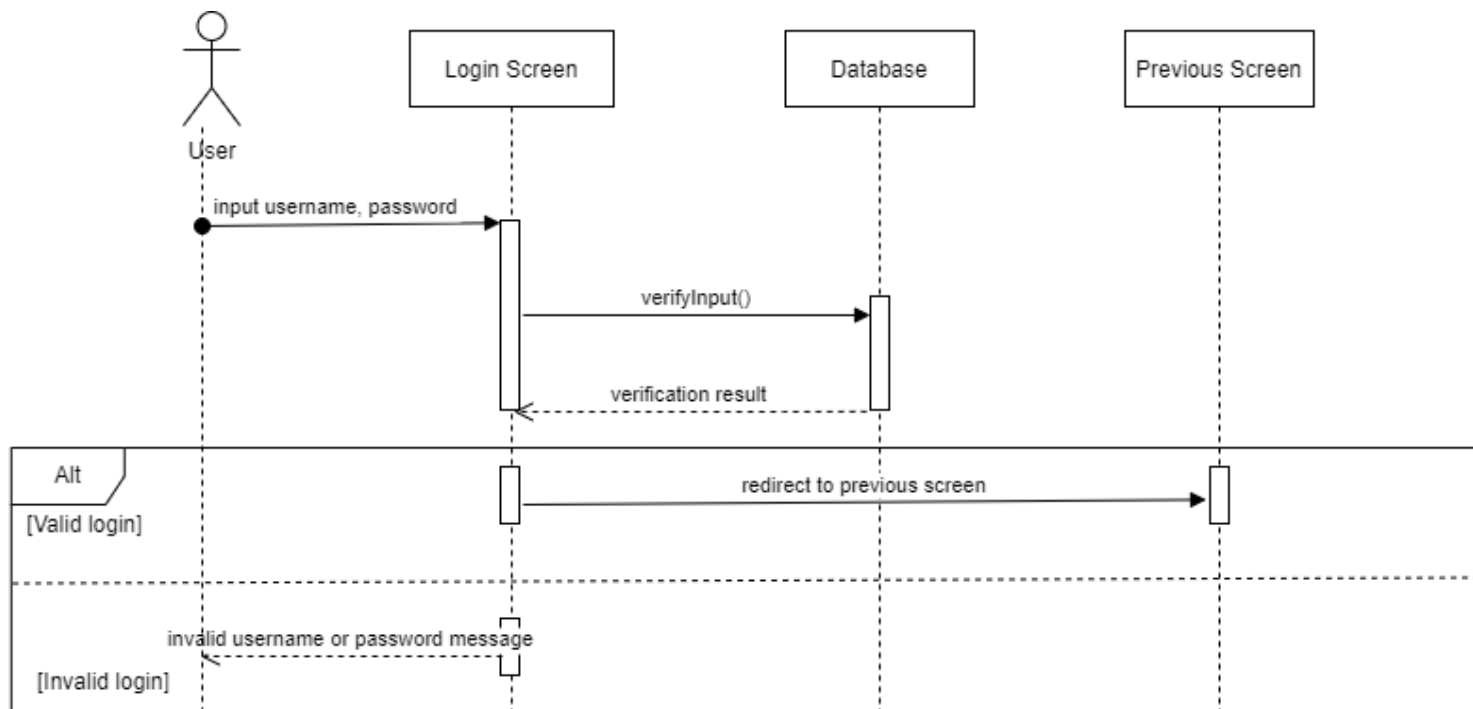
Staff viewing registered
patient's info
HLR 18



3.2.3 Sequence Diagrams

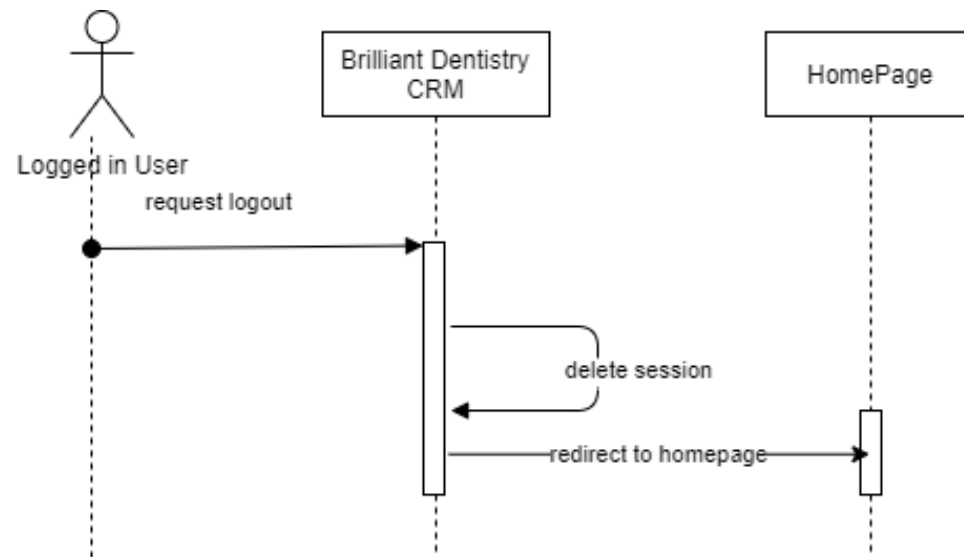
3.2.3.1 Login

Actor: Registered Patients, Staff



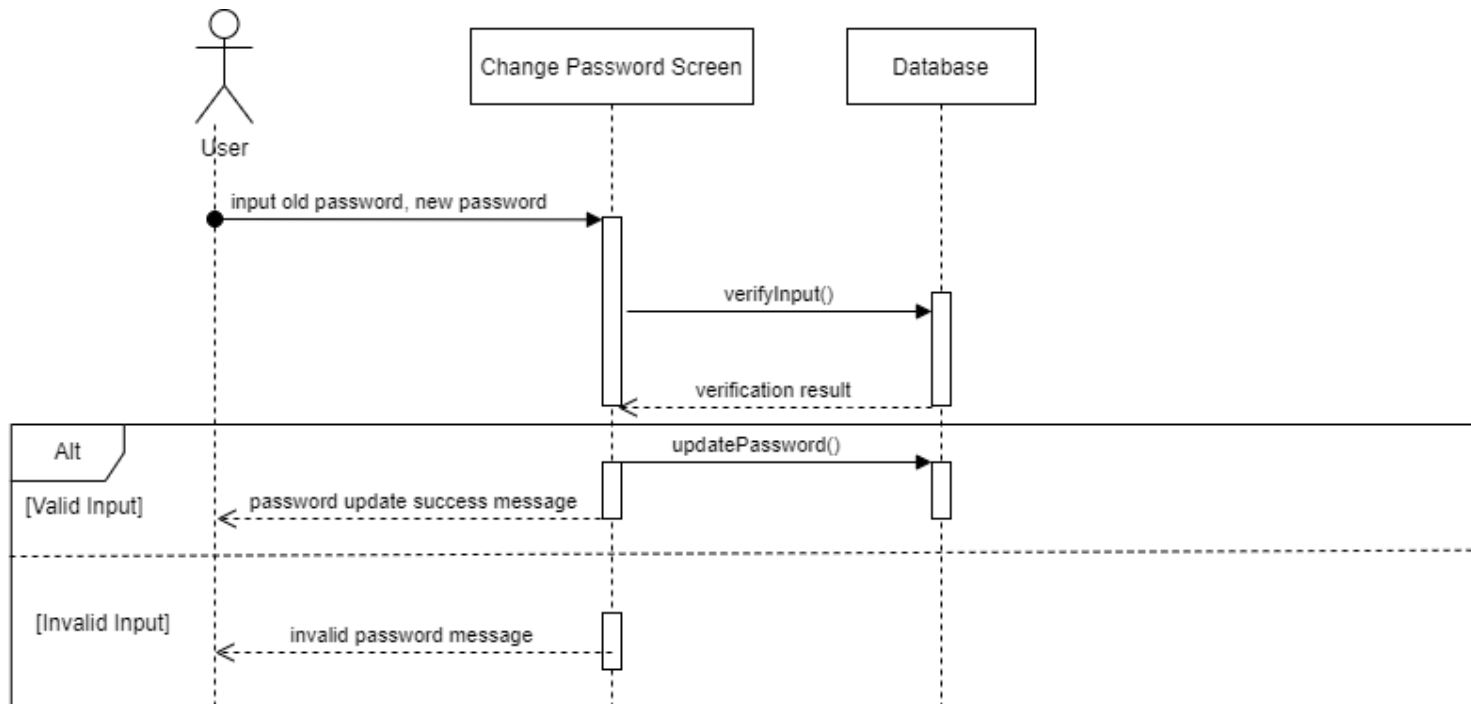
3.2.3.2 Logout

Actor: Logged In Patients, Logged In Staff



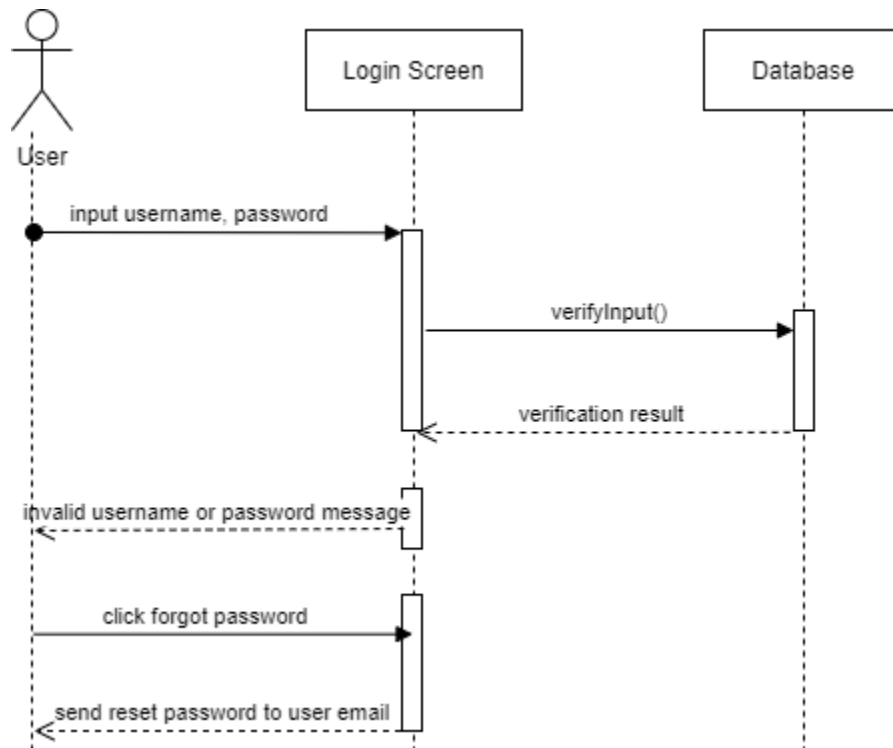
3.2.3.3 Change Password

Actor: Registered Patients, Staff



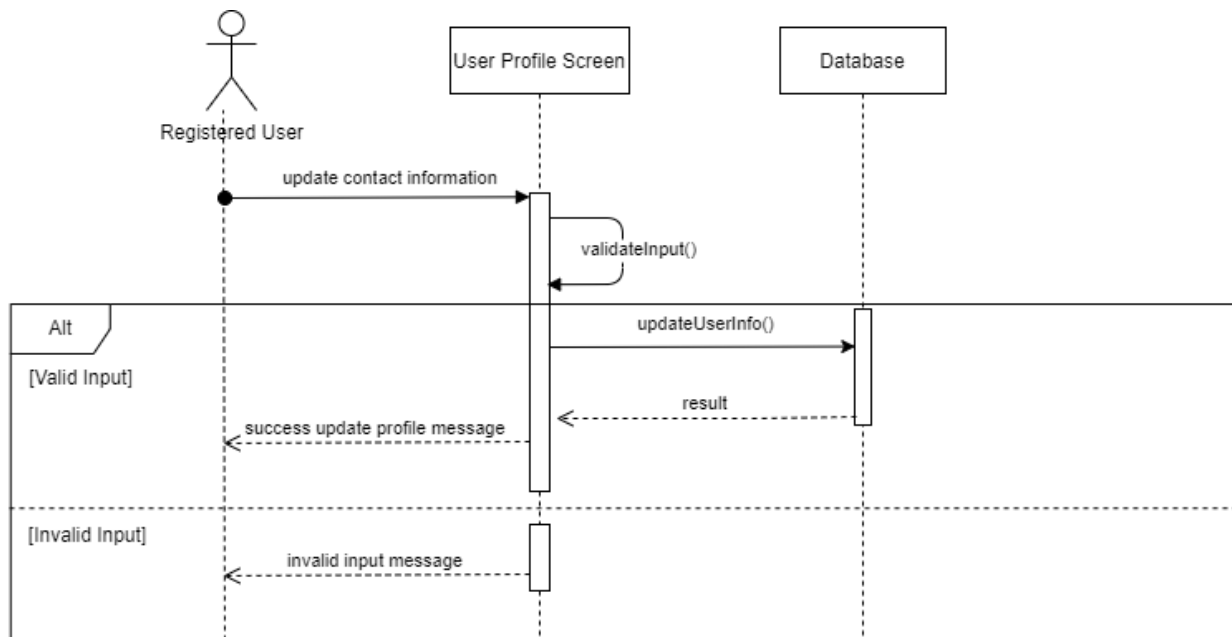
3.2.3.4 Reset Password

Actor: Registered Patients, Staff



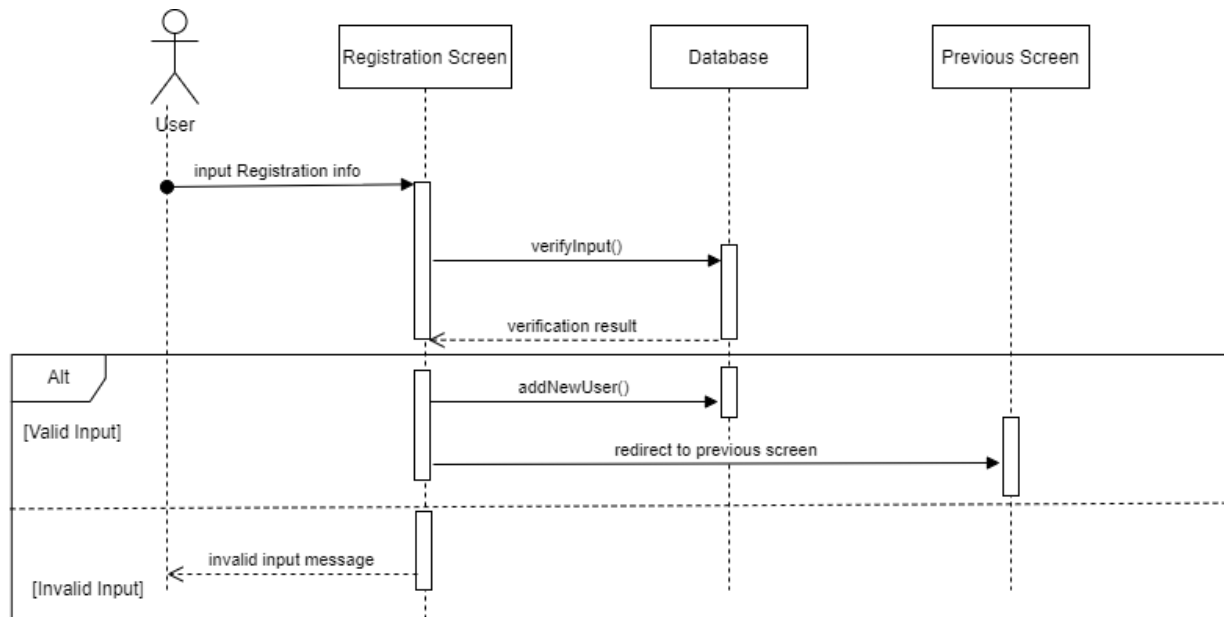
3.2.3.5 Update Profile Information

Actor: Registered Patients, Staff



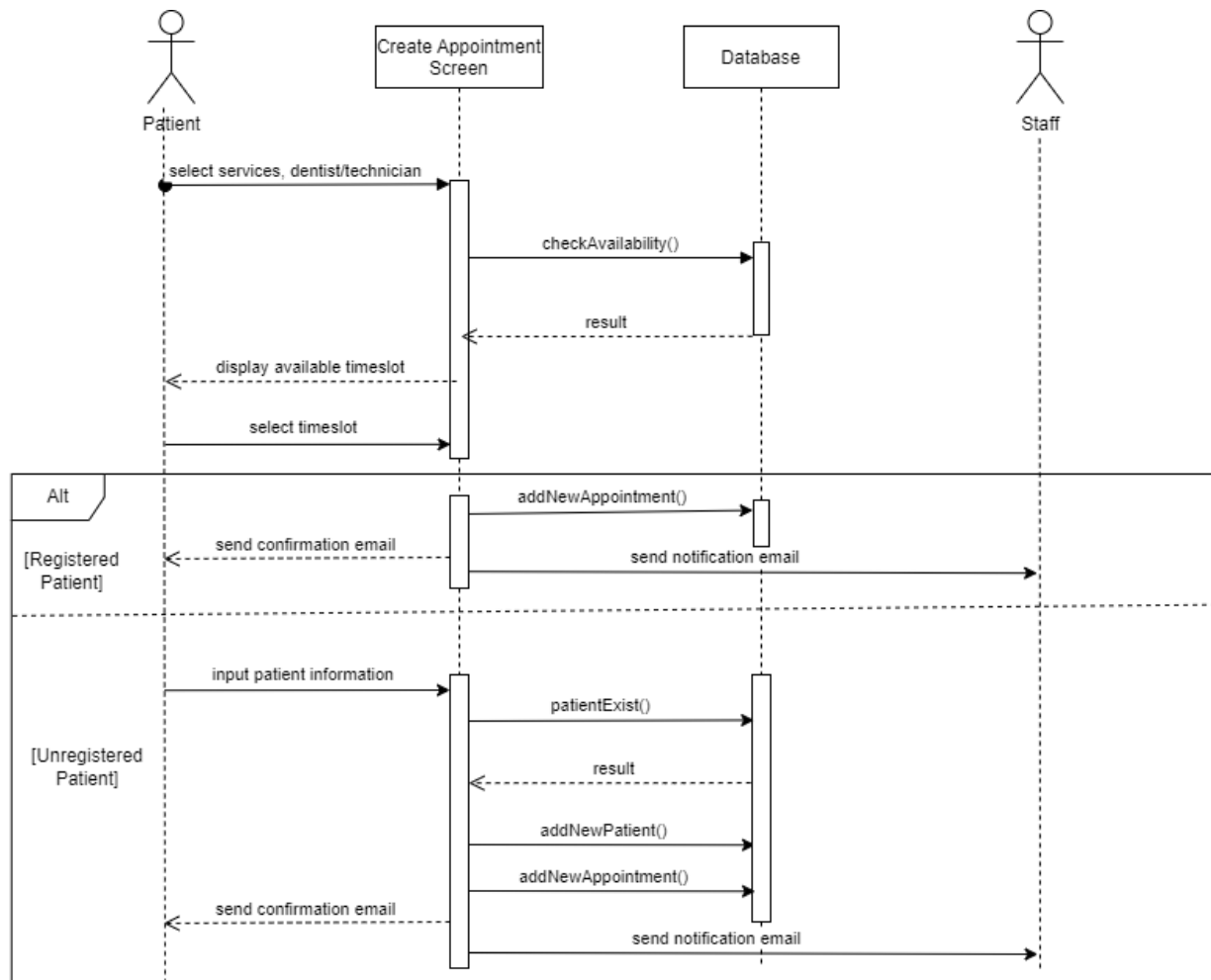
3.2.3.6 Registration

Actor: New Patients



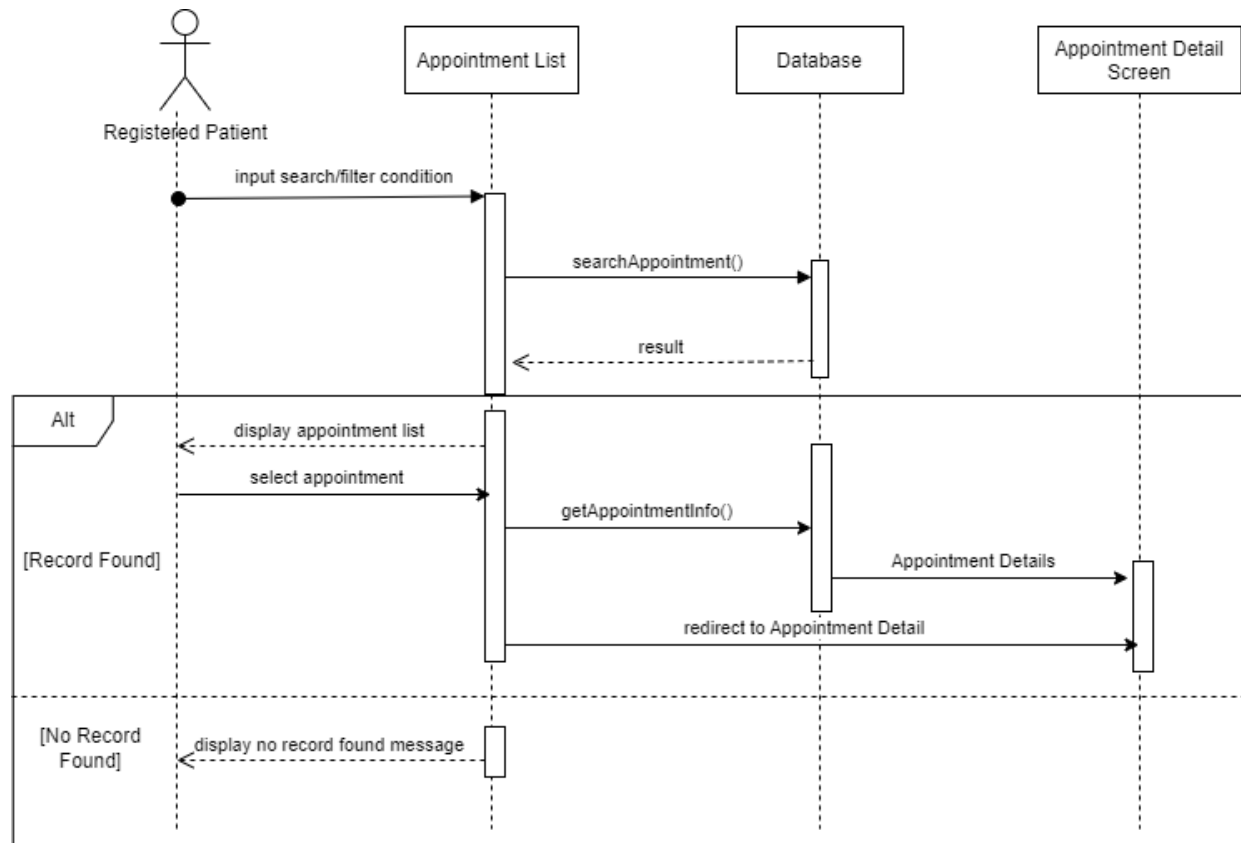
3.2.3.7 Patient Create Appointment

Actor: Registered/unregistered Patient



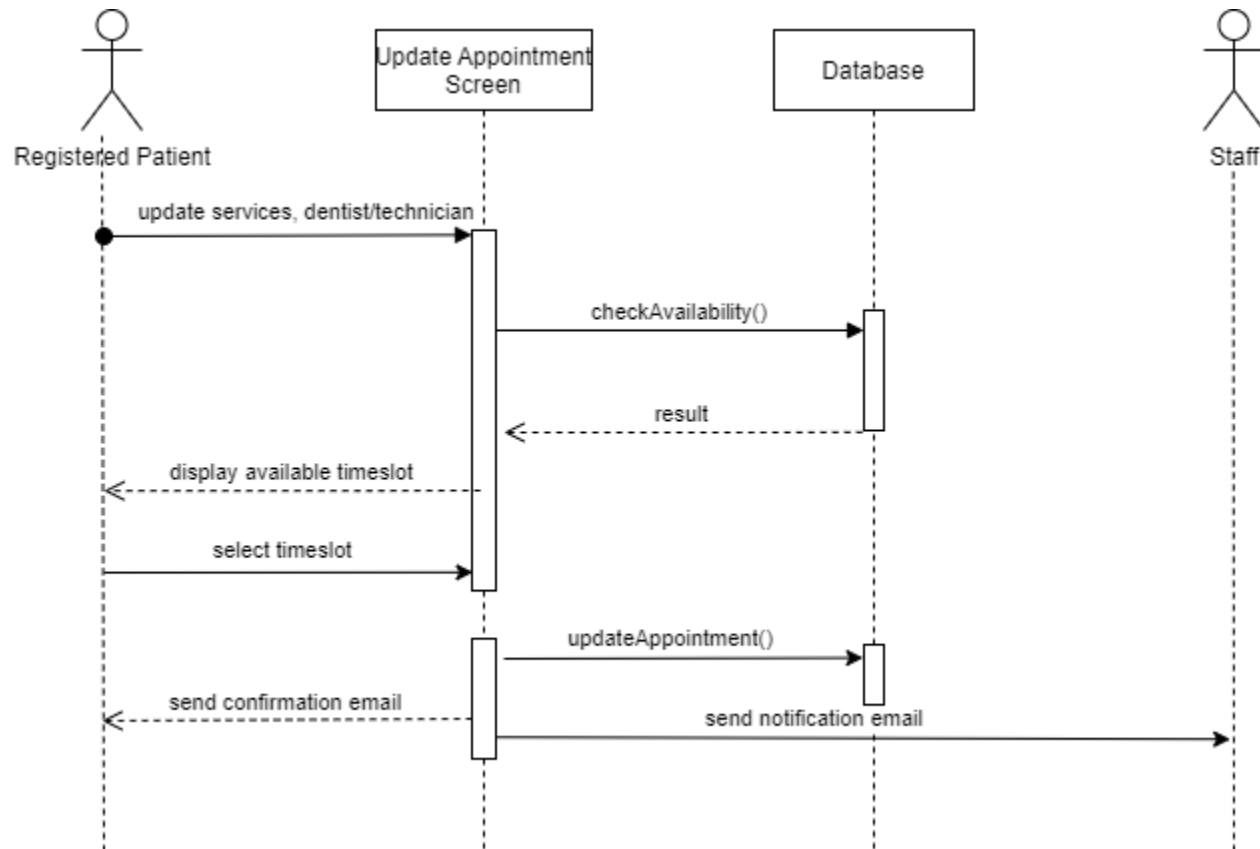
3.2.3.8 Patient View Appointment

Actor: Registered Patient



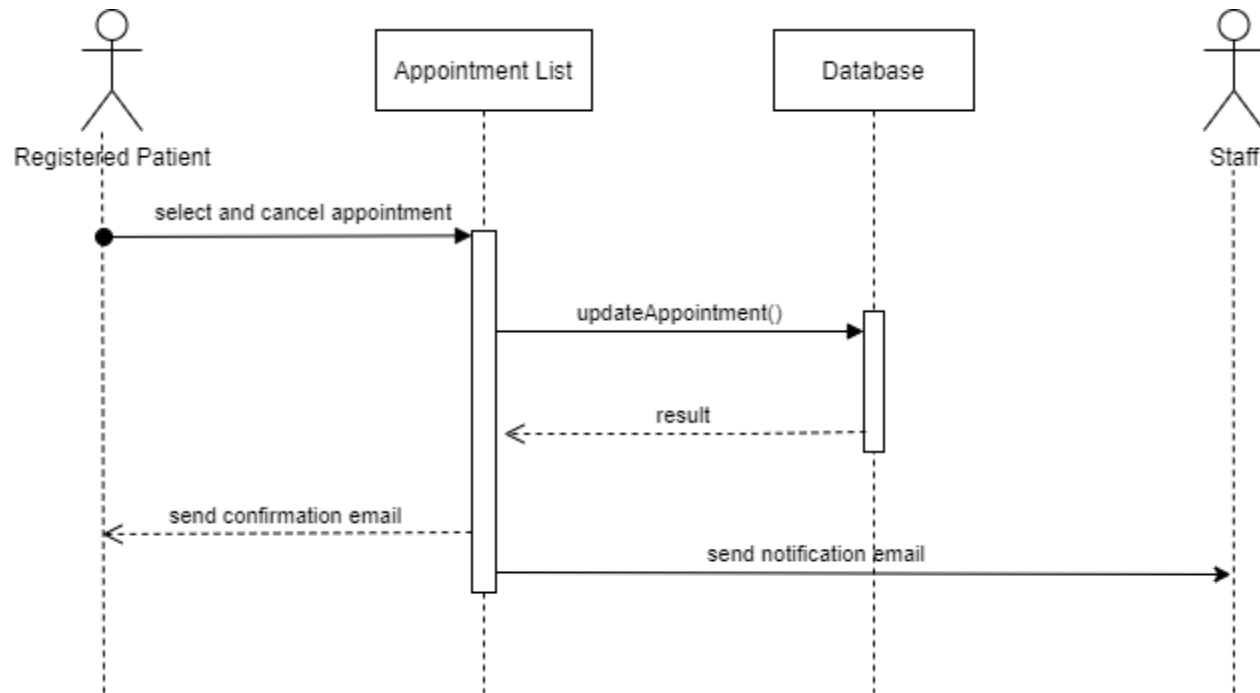
3.2.3.9 Patient Update Appointment

Actor: Registered Patient



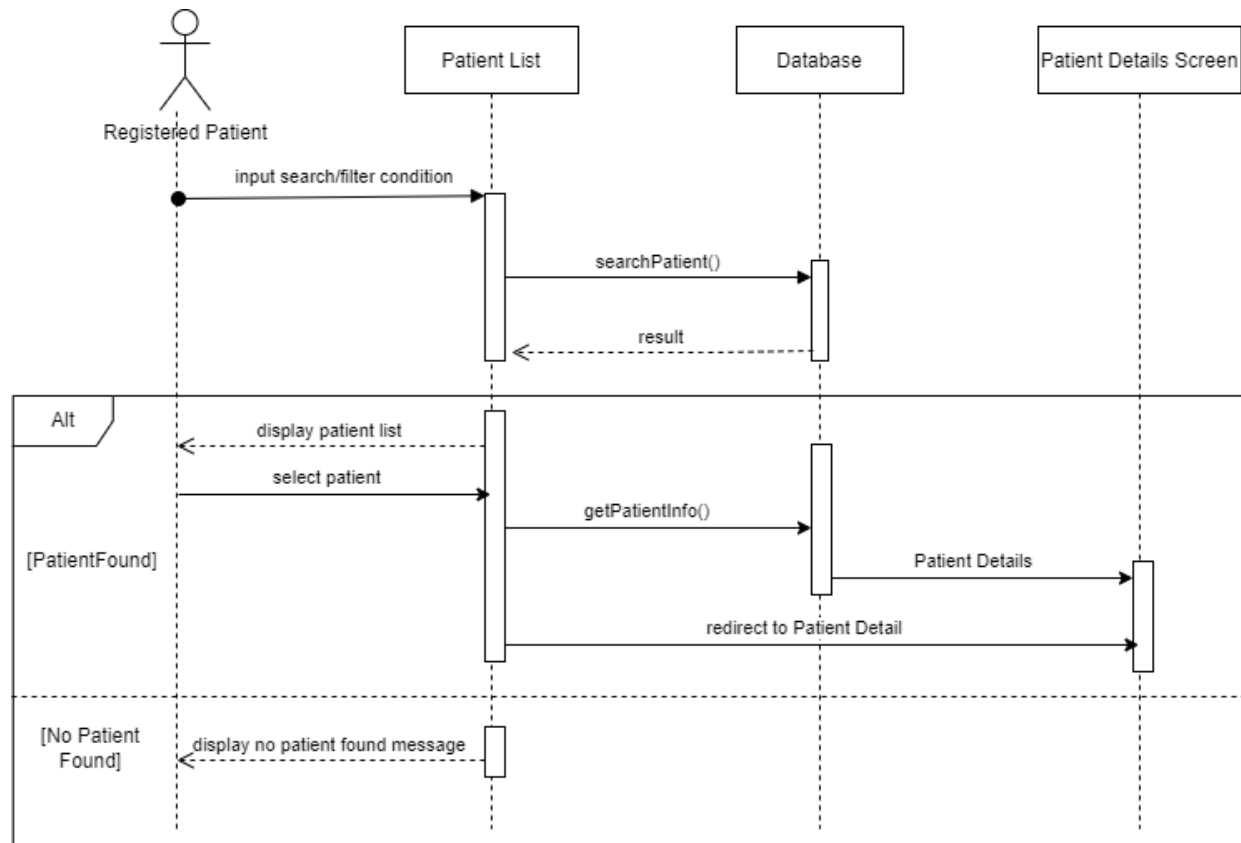
3.2.3.10 Patient Cancel Appointment

Actor: Registered Patient



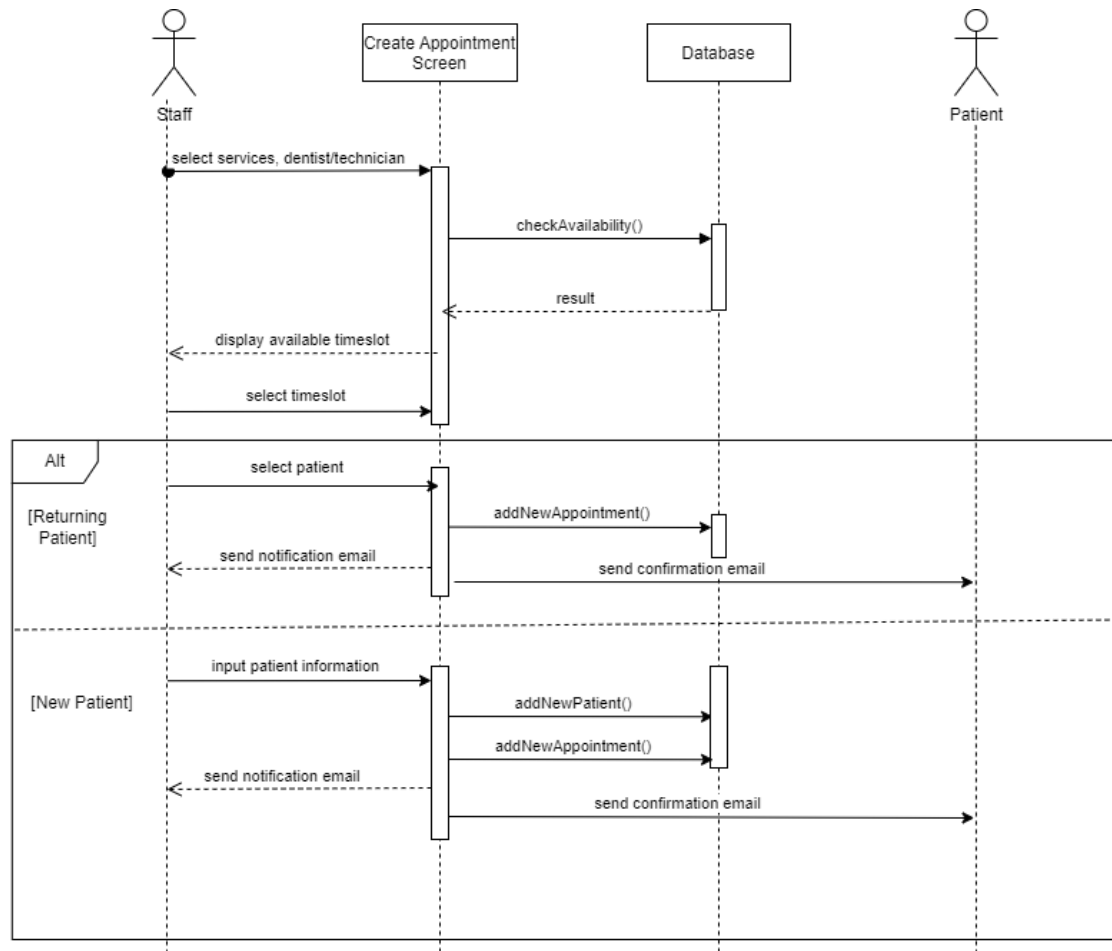
3.2.3.11 Staff View Patient Information

Actor: Staff



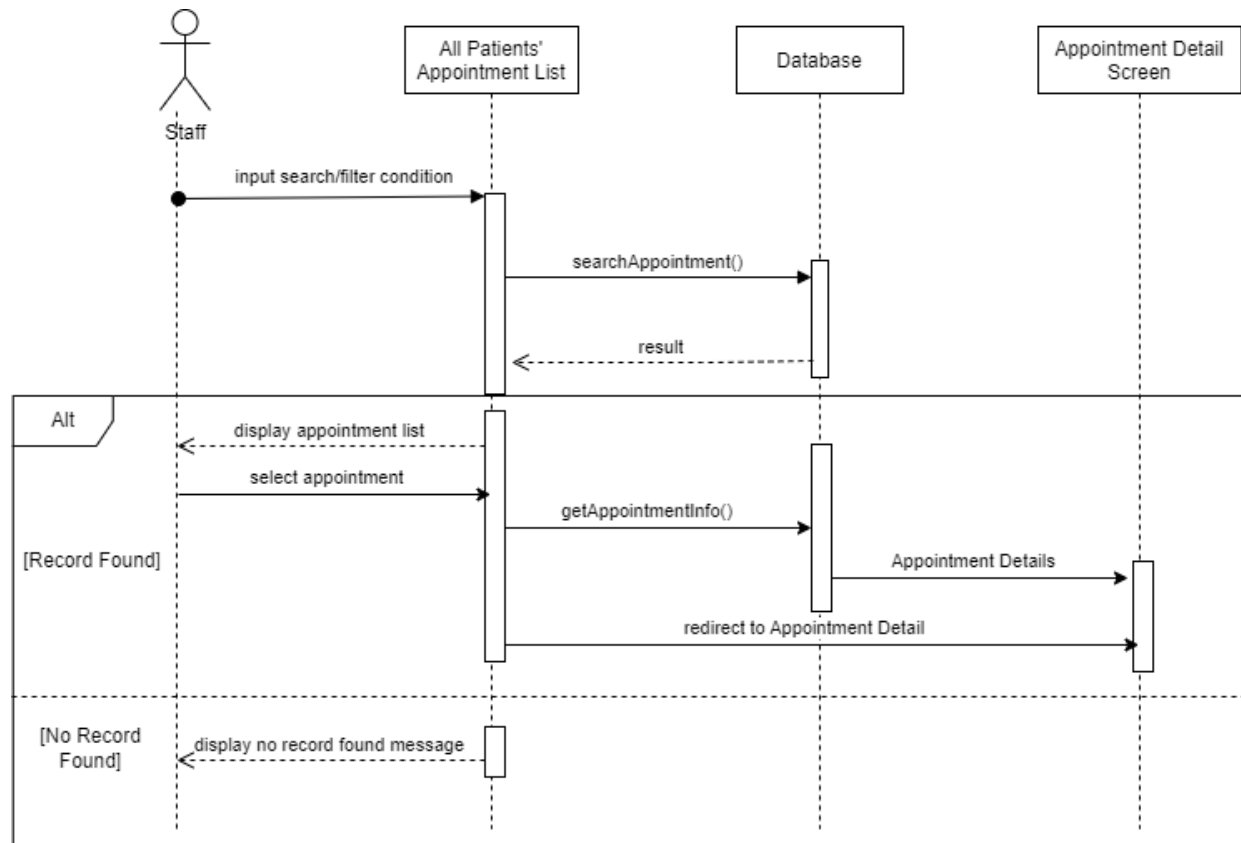
3.2.3.12 Staff Create Appointment

Actor: Staff



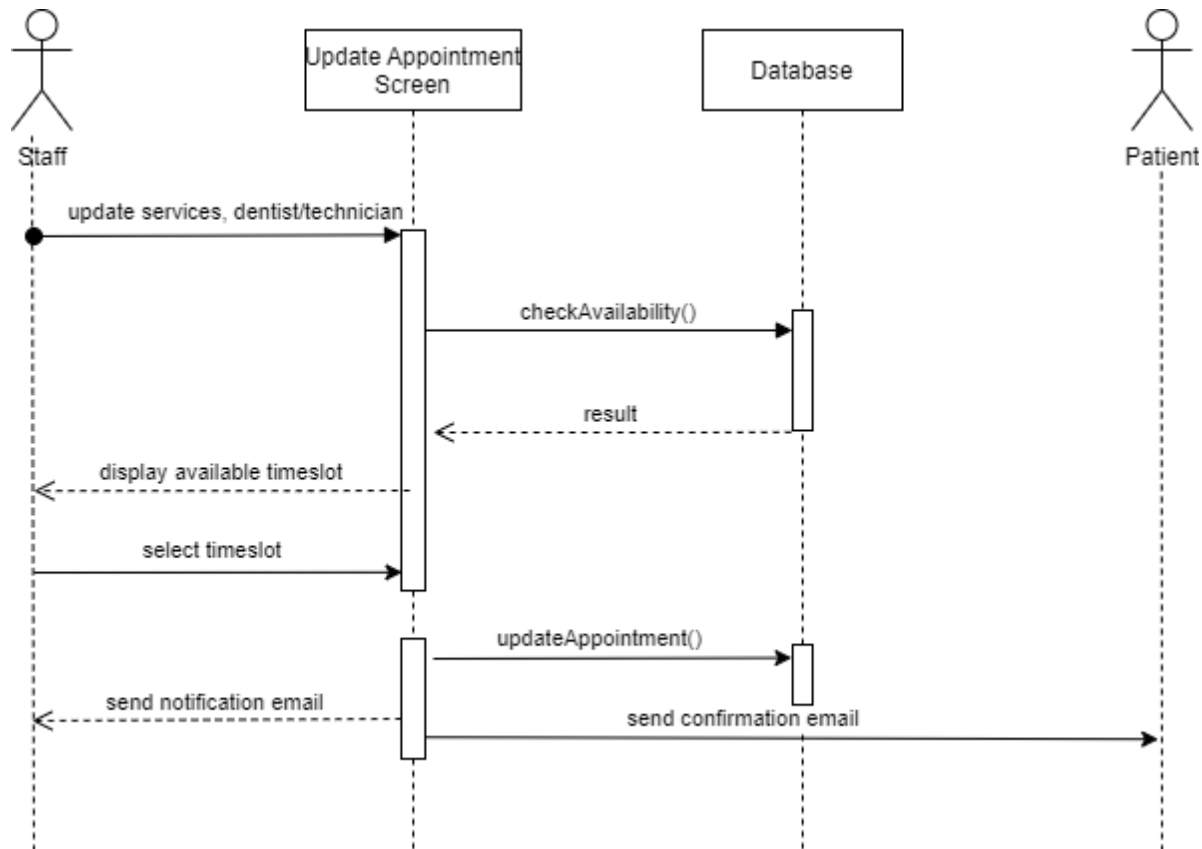
3.2.3.13 Staff View Patient Appointment

Actor: Staff



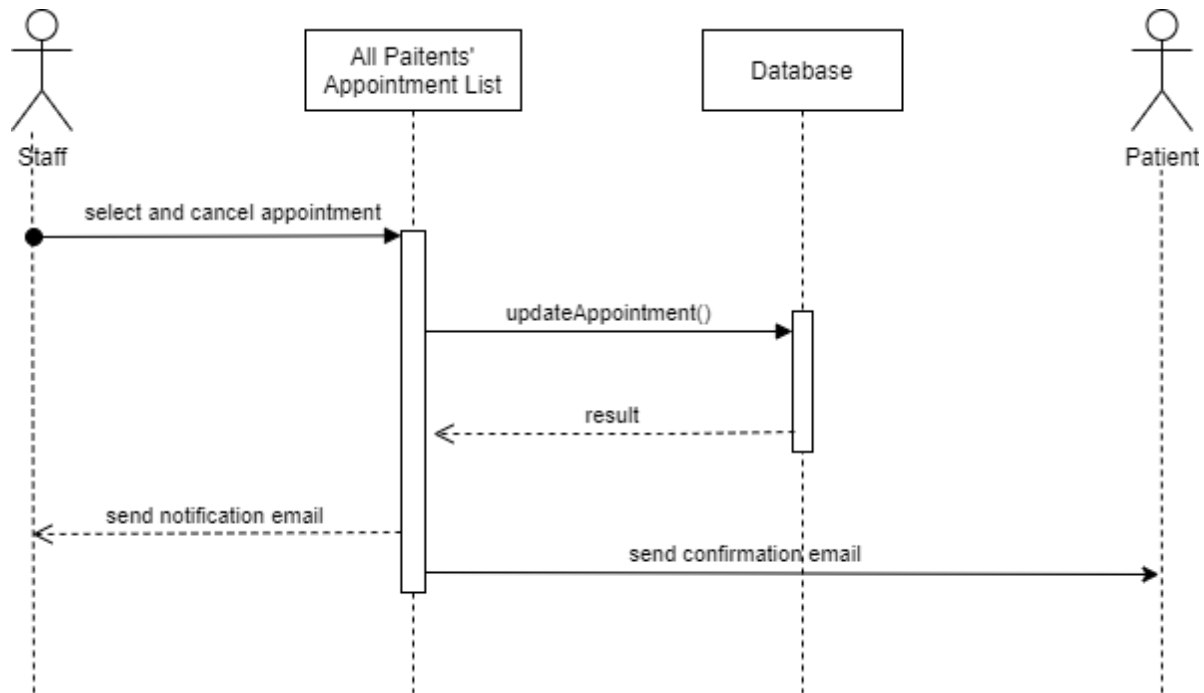
3.2.3.14 Staff Update Patient Appointment

Actor: Staff

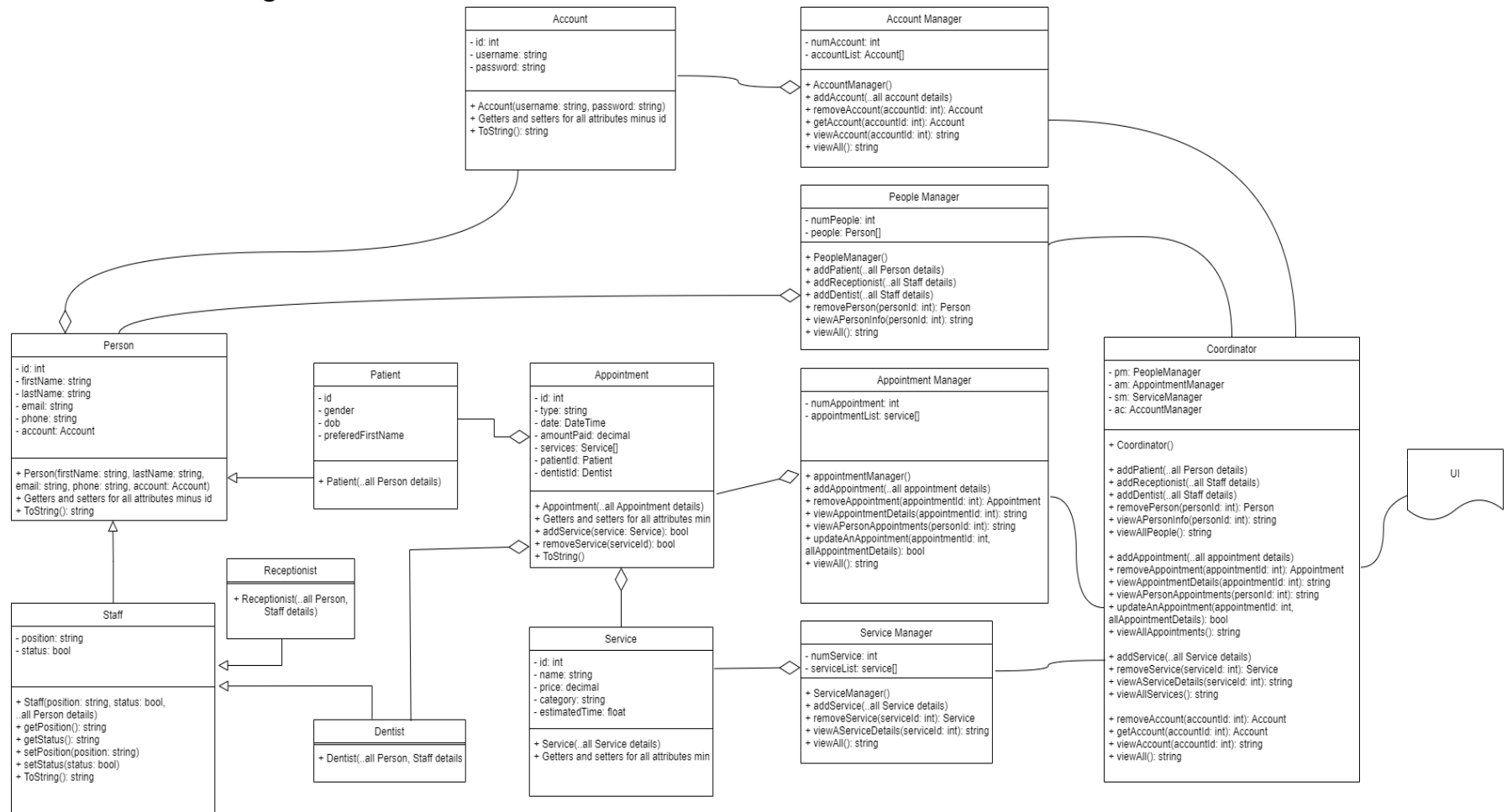


3.2.3.15 Staff Cancel Patient Appointment

Actor: Staff



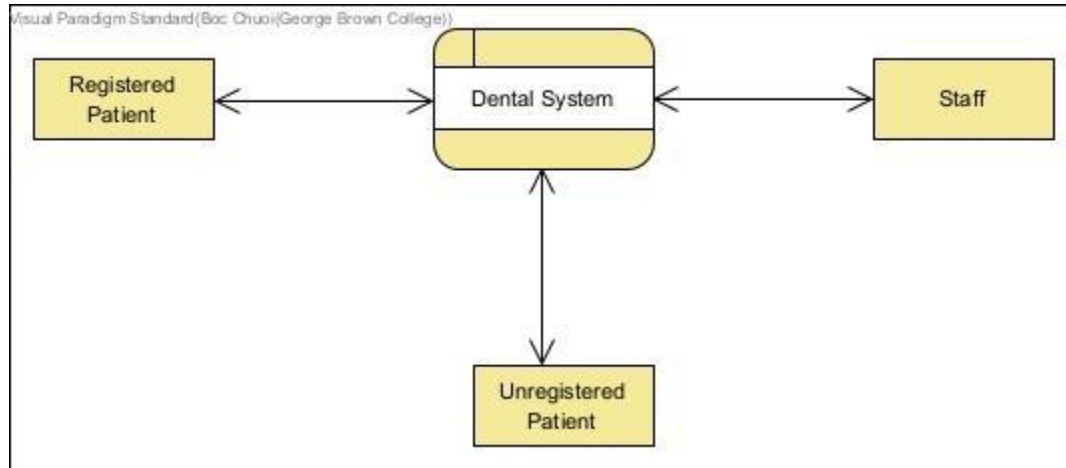
3.2.4 UML Class Diagram



3.3 Process Modelling

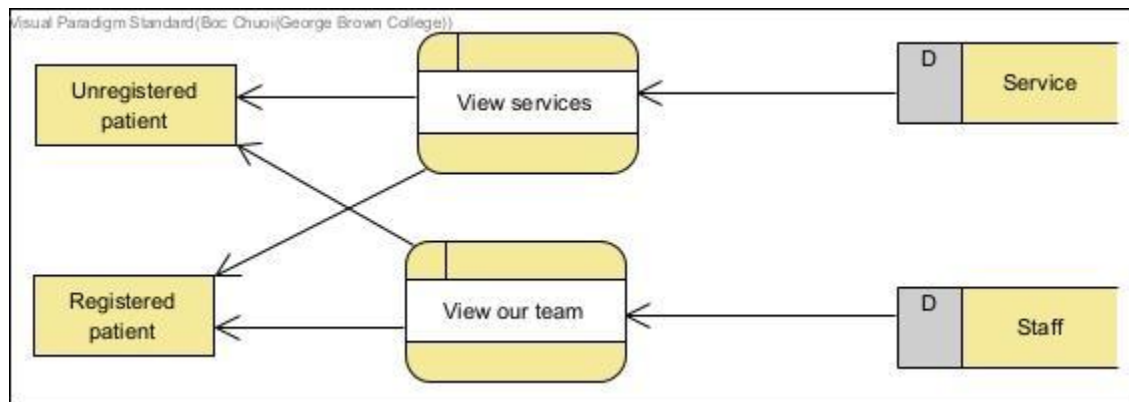
3.3.1 Data Flow Diagram

3.3.1.1 Context Diagram

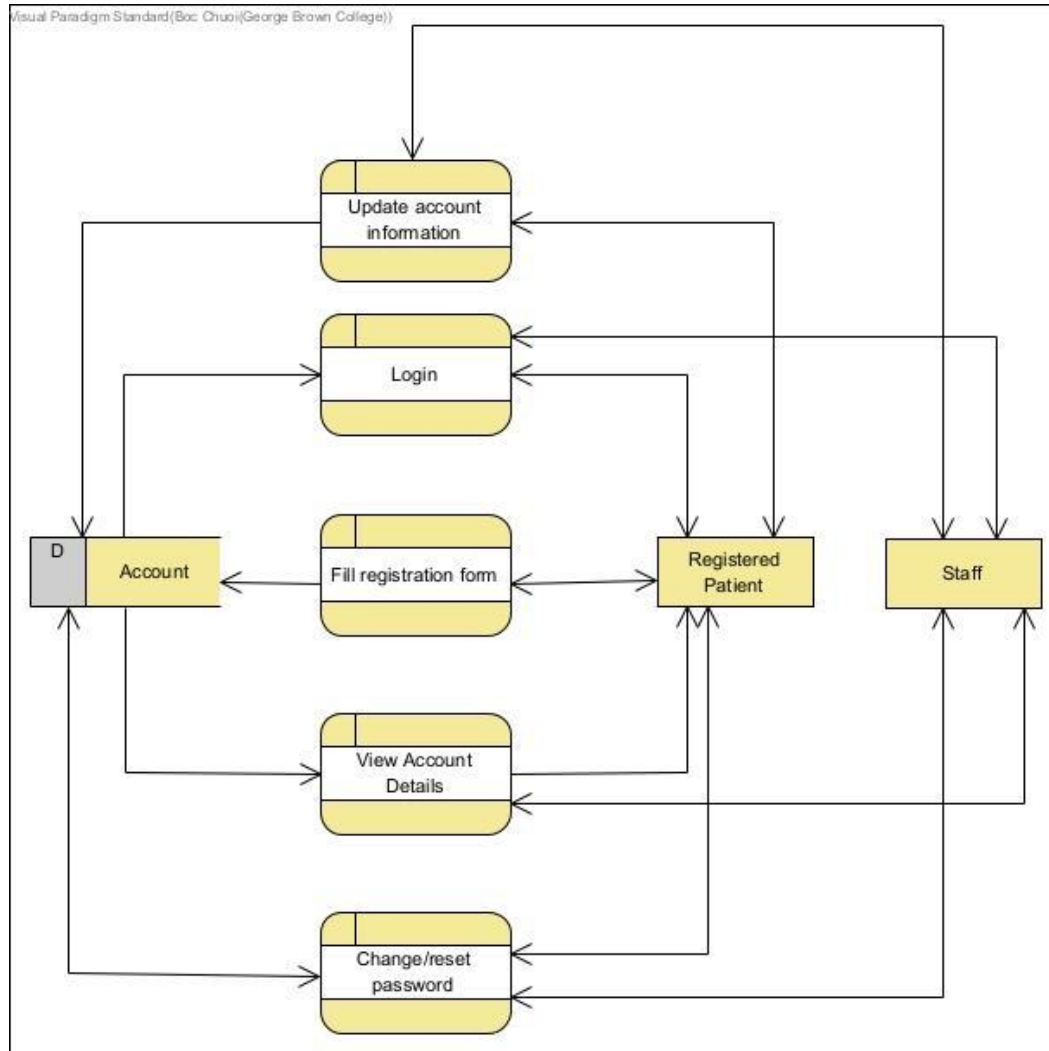


3.3.1.2 DFD

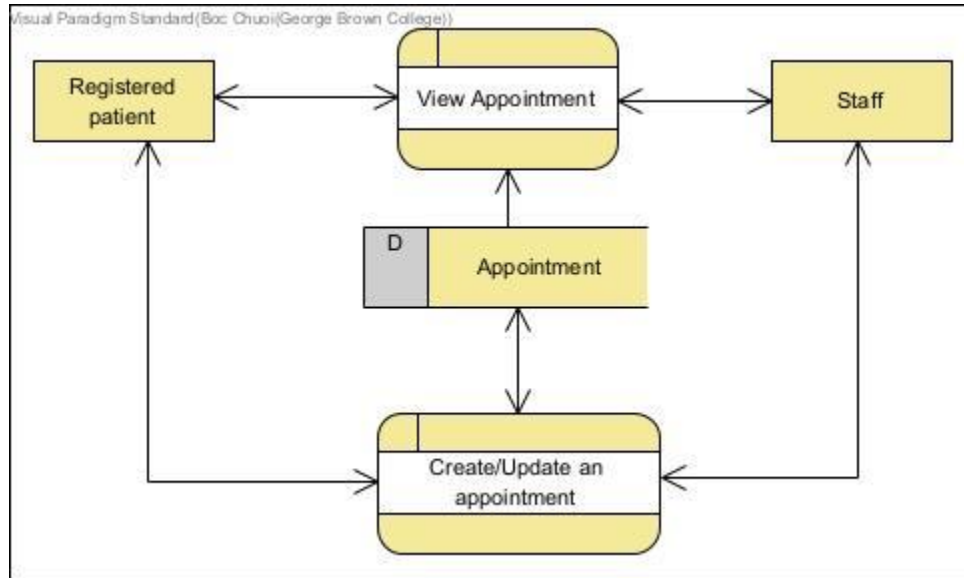
Page Contents



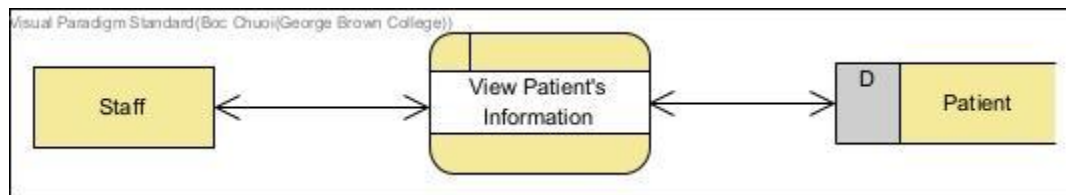
User Accounts



Appointments



Patient Information



4.0 Non-Functional Requirements

4.1 Performance

- The system should return the results within 10 seconds for any queries requested from client side.

4.2 Reliability

- The search results must follow the parameters entered in the filters of the search; no single search result can be not relevant / outside of the parameters specified in the filters.

4.3 Availability

- The system should be available at least 98% of the time.
- Any scheduled downtime or maintenance must be informed to client 5 business days in advance.
- Any unexpected downtime or service corruption must be resolved within 1 business day.

4.4 Security

- User credentials must be stored safely and encrypted in our system.
- Unauthorized users shall have no access to features that are restricted to registered users.
- Data inside the system or its part will be protected against malware attacks or unauthorized access
- Staff users must change their password every 90 days to reduce the risk of password compromise and exposure of patient confidential information.
- Staff access must be reviewed regularly. As soon as a staff resigned from her position, the development team must be informed and take action to ensure that the resigned staff has no access to the system.
- System must maintain a log of user activities for further investigation in case of incidents.

4.5 Maintainability

- The application will be implemented with React, Flask and MySQL.
- The application must comprise well-defined and independent components/modules for better maintainability and scalability.
- New features can be added to the application without major changes to the underlying architecture.

4.6 Portability

- Brilliant Dentistry CRM is a web application that any user with reliable access to internet and web browser can easily access to.
- With responsive design, the web application can be accessed using different type of devices including but not limited to desktop, laptop, Chromebook, tablet, phone, etc.

4.7 Usability

- The user interface must be intuitive, simple, easy to navigate and require the minimum amounts of click.
- Default values if available must be selected so that they are a good choice for most of the users.
- If default values are not available, there should be hints to help end users understand the expected input
- In case of validations or errors, the message must be informative, specific, concise, and easy to interpret.

5.0 Logical Database Requirements

- The Database Manager should be MySQL
- All data will be saved in the database: user accounts and profiles, booking information (except files which are stored on the disk.)
- Information data and files must be accessible from a single source (endpoint)
- The database allows concurrent access and will be always kept consistent, requiring a good database design
- All data must be retained with a minimum period of 2 years. Data that is more than 2 years must be archived and accessible using MySQL management tools.

- Data backup must be scheduled weekly
- Security protocols and data encryption must implement to protect sensitive information

6.0 Approval

The signatures below indicate their approval of the contents of this document.

Project Role	Name	Signature	Date
Counselor/Liaison	Reza Dibaj	Reza Dibaj	2022/11/08
Client	Shiva Aghaha	Shiva Aghaha	2022/11/08
Professor	Anjana Shah		