

# Table of Contents

[Table of Contents 2](#_Toc137916575)

[1 Section 1: Problem statement 4](#_Toc137916576)

[1.1 Problem & Need 4](#_Toc137916577)

[1.2 Capabilities and benefits 4](#_Toc137916578)

[1.3 Stakeholders and their roles 4](#_Toc137916579)

[1.4 Sub-systems of the application 5](#_Toc137916580)

[1.5 Intended users of the SRS documentation? 5](#_Toc137916581)

[2 Section 2: A Context Flow – Structured Modeling 5](#_Toc137916582)

[2.1 Context Flow Diagram (CFD) 5](#_Toc137916583)

[3 Section 3: Requirements – Functional - UML Use Case Modeling 7](#_Toc137916584)

[3.1 Subsystem 1: Administration 7](#_Toc137916585)

[3.1.1 Goal Use Cases 7](#_Toc137916586)

[3.1.2 User Case Diagram 7](#_Toc137916587)

[3.1.3 User Story 8](#_Toc137916588)

[3.2 Subsystem 2: Scheduling 10](#_Toc137916589)

[3.2.1 Goal Use Cases 10](#_Toc137916590)

[3.2.2 User Case Diagram 10](#_Toc137916591)

[3.2.3 User Story 11](#_Toc137916592)

[3.3 Subsystem 3: Appointment Management 12](#_Toc137916593)

[3.3.1 Goal Use Cases 12](#_Toc137916594)

[3.3.2 User Case Diagram 13](#_Toc137916595)

[3.3.3 User story 13](#_Toc137916596)

[3.4 Subsystem 4: Patient Management Profile 16](#_Toc137916597)

[3.4.3 Goal Use Cases 16](#_Toc137916598)

[3.4.4 User Case Diagram 17](#_Toc137916599)

[3.4.5 User Story 17](#_Toc137916600)

[3.5 Subsystem 4: Doctor Profile 19](#_Toc137916601)

[3.5.3 Goal Use Cases 19](#_Toc137916602)

[3.5.4 User Case Diagram 20](#_Toc137916603)

[4 Section 4: UML Domain Class Diagram 22](#_Toc137916604)

[4.1 A List of the classes: 22](#_Toc137916605)

[4.2 Domain class diagram: 23](#_Toc137916606)

[5 Section 5: ERD Diagram 24](#_Toc137916607)

[5.1 A List of All Entities: 24](#_Toc137916608)

[5.2 ERD Diagram: 25](#_Toc137916609)

[6 Section 6: UML Systems Sequence diagram 26](#_Toc137916610)

[6.1 Introduction of the diagram 26](#_Toc137916611)

[6.2 UML Systems Sequence diagram: WaitingList\_sequence 26](#_Toc137916612)

[6.3 UML Systems Sequence diagram: Appointment\_sequence 27](#_Toc137916613)

[7 Section 7: UML State Diagrams 28](#_Toc137916614)

[7.1 Introduction of the diagram 28](#_Toc137916615)

[7.2 UML State Diagrams: Scheduling Class object 29](#_Toc137916616)

[7.3 UML State Diagrams: Appointment Class object 30](#_Toc137916617)

[8 Section 8: Technologies 31](#_Toc137916618)

[8.1 Relevant technologies: Web Application 31](#_Toc137916619)

[9 Section 9: Gantt Chart 32](#_Toc137916620)

[10 Reference: 33](#_Toc137916621)

Part A: Project Scope and Requirements

# Section 1: Problem statement

## Problem & Need

In the post-COVID era, people have concerns about the risk of contracting COVID-19 or other contagious diseases when visiting a clinic. It is understandable that patients who need to see a doctor are often in a vulnerable condition, making them more susceptible to infections. Even before the COVID-19 pandemic, scheduling appointments with doctors has been a source of frustration, adding to the challenges of seeking medical care in the first place.

One potential issue is the presence of communication barriers, especially when clinic staff members have heavy accents. This can lead to difficulties in understanding each other or, at worst, misunderstandings that could affect the quality of care. Additionally, imagine waking up in the middle of the night, feeling the need for medical attention, but having to wait until office hours to make a phone call and schedule an appointment. Naturally, you would want to secure an appointment as soon as possible, but the process can be time-consuming and inconvenient.

We are fully aware of the difficulties and struggles associated with these issues. Therefor, we create an appointment application providing another method to solve the above problems and needs.

## Capabilities and benefits

Our application aims to assist patients in making doctor appointments anytime, quickly, and efficiently. We prioritize providing a smooth and pleasant user experience. By utilizing our application, users can save valuable time compared to the traditional method of calling individual clinics.

With our application, users can easily access a comprehensive list of doctors along with their availability and specialties. They have the convenience of viewing, modifying, and canceling their appointments as needed.

## Stakeholders and their roles

There are three types of users who interact with the system: client-side users, i.e.: doctors, patients, system administrators, testers, and developers.

**Physicians** include doctors, nurses, and clinics. Physicians will be able to create/ update their profile, enter their upcoming timetable, view scheduled patients and cancel/ revise appointments.

**Patients** can create/ update their profiles, view available doctors, and make new/ manage existing appointments.

**System Administrator** is responsible for a smooth operation. The system administrator can manage permissions of other roles and manually amend appointments.

**Testers** are responsible for testing the application to ensure that it meets the requirements and is free of defects. They are responsible for identifying issues and reporting them to developers.

**Developers** are responsible for designing and developing the application according to the requirements provided by stakeholders. They are responsible for ensuring that the application is functional and meets the requirements of stakeholders.

## Sub-systems of the application

There are five functional components, i.e.: doctor profile management sub-system, patient profile management sub-system, administrative sub-system, scheduling sub-system, and appointment management sub-system.

**Doctor profile management sub-system** allows doctors to create new/ update existing profile, log in to the system and allows the administrator to change permission.

**Patient profile management sub-system** allows doctors to create new/ update existing profile, log in to the system and allows the administrator to change permission.

**Administrative sub-system** allows administrators to manipulate doctor-patient appointments.

**Scheduling sub-system** collects and generate GUI for application user and them to check the upcoming timetable and shows patients available doctors.

**Appointment management sub-system** allows patients to create new appointments, both doctors and patients to manage existing appointments, and for doctors to view scheduled patients.

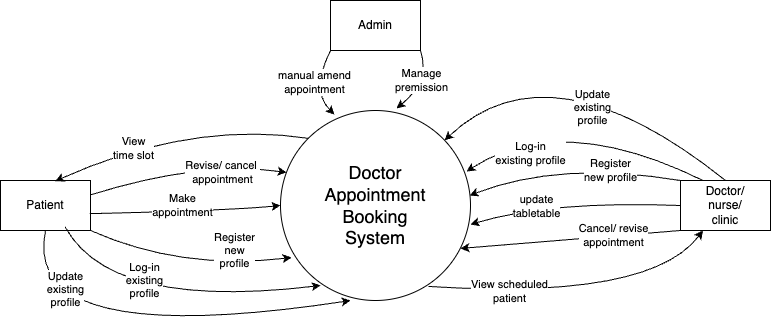
## Intended users of the SRS documentation?

This SRS document is intended to be read by

* Software Developers
* Project Managers: Organization Administrators.
* Users: Employees and their supervisors.
* Testers: Deployment Technology Experts
* Documentation Writers: The Project/Development Team.

# Section 2: A Context Flow – Structured Modeling

## Context Flow Diagram (CFD)



# Section 3: Requirements – Functional - UML Use Case Modeling

## Subsystem 1: Administration

### Goal Use Cases

| FR # | Goal Use Case | Role Player | Description |
| --- | --- | --- | --- |
| FR-01 | Manage Permissions | System Administrator | Provide permissions to control or access the system based on user role. |
| FR-02 | Error Tracking and alerting notifications | System Administrator | Provide a quick way to identify system issues, minimize system downtime to ensure stable and reliable operation, aiming for a better user experience. |
| FR-03 | Manage Data Source | System Administrator | Provide a system which allows administrators to upload and manage data sources from customers and the medical team. |
| FR-04 | Create, update, and delete user profiles | System Administrator | A mechanism which allows users with administrative privileges to create, update and delete profiles. |

### User Case Diagram

A picture containing text, screenshot, line, diagram

Description automatically generated

### User Story

#### Manage Permissions

As a user, I want to manage permissions so that I can ensure access control.

Acceptance Criteria:

* The system administrator should be able to revoke permission or modify access rights when necessary.
* The system administrator, with administrative privileges, should be able to edit and modify existing roles and define hierarchical permissions.

#### Error Tracking and alerting Notifications

As a user, I want to track errors and receive alert notifications of the system tasks so that I can troubleshoot the system and ensure smooth operation for all users.

Acceptance Criteria:

* The system administrator should be able to monitor the system.
* The system administrator should receive alerting and system notifications to ensure prompt attention and resolution of the problem.
* The system administrator should be able to track all onboarding and offboarding metrics to measure the efficiency and productivity of the procedures.

#### Manage Data Source

* As a user, I want to manage all user’s data so that I can generate diagnosis reports for reference and analysis.

Acceptance Criteria:

* The system administrator should be able to generate diagnosis reports for reference and analysis.
* The system administrator should be able to export reports in a preferred file format.
* The system administrator should be able to customize the content and format of reports, selecting specific data fields and layouts.
* The system administrator should be able to securely access and manage user data, ensuring data protection, confidentiality, and integrity.
* The system administrator should be able to search and filter data users to find user information quickly.

#### Create, update, and delete user profiles.

* As a user, I would like to have the ability to manage user profiles by creating, updating, and deleting them, so that I will have complete control and administration over the application.

Acceptance Criteria:

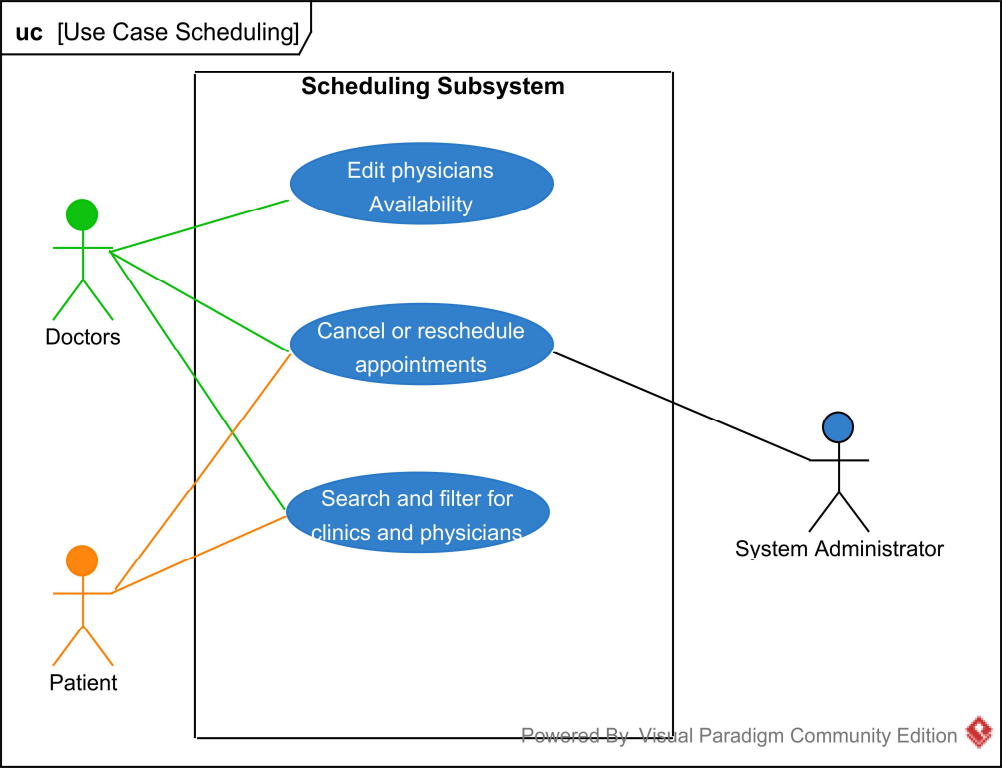
* The system administrator should be able to create new user profiles by entering their information.
* The system administrator should be able to update existing user profiles.
* The system administrator can view a list of all user profiles.

## Subsystem 2: Scheduling

### Goal Use Cases

| FR # | Goal Use Case | Role Player | Description |
| --- | --- | --- | --- |
| FR-05 | Edit doctors Availability | Doctors and System Administration | Provide a calendar of available time slots for each doctor, considering working hours, and service providers and resources availability. |
| FR-06 | Cancel or reschedule appointments | All users | Provide a mechanism to modify or cancel the appointment~~.~~ |
| FR-07 | Search and filter for clinics and doctors | Patients | A mechanism to allow patients to search and filter for doctors and clinics based on specific criteria, such name, speciality, location, or other relevant data. |

### User Case Diagram



### User Story

#### Edit doctors' Availability.

As a user, I want to edit the availability of doctors so that I can manage doctors' schedules, ensuring accurate information for patients.

Acceptance Criteria:

* The user can view the current availability status.
* The user can set unavailable time slots.
* The user can modify their availability for appointments.
* The user can revert the availability status to their previous state.

#### Cancel or reschedule the appointment.

As a user, I want to cancel or reschedule appointments so that I can easily manage my appointments.

Acceptance Criteria:

* The patient should be able to cancel or reschedule an appointment within the app or system.
* The user can select a specific appointment to reschedule or cancel.

#### Search and filter for clinics and doctors

As a user, I want a way to search and filter doctors and clinics based on certain criteria, so I can quickly find what suits my requirements.

Acceptance Criteria:

* The patient should be able to search by name, specialty, availability or location for clinics and doctors.
* The patient needs to have the ability to filter the search results based on what is most relevant to them.

#### List of patients' appointments

As a user, I want to view a list of upcoming appointments, including relevant details, such as data, time, location, and name of the patient and doctor, so I can manage my schedule.

Acceptance Criteria:

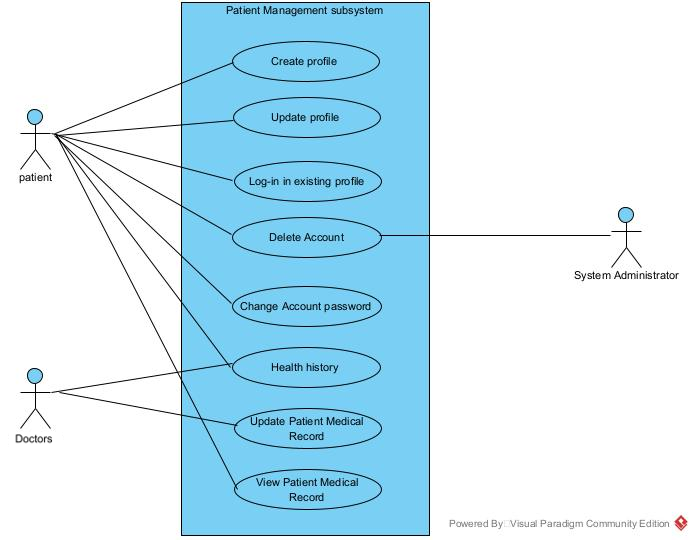
* The patient should be able to view his/her appointments in a user-friendly display.
* Both the patient and doctor should have access to important details regarding the appointment, including specialty, location, time, and the names of both parties involved.

## Subsystem 3: Appointment Management

### Goal Use Cases

| FR # | Goal Use Case | Role Player | Description |
| --- | --- | --- | --- |
| FR-08 | Book an appointment | Patients | Patients select date, time, and doctor to request an appointment. |
| FR-09 | Edit Appointment | All users | Users can change details of existing appointments, with restrictions based on healthcare setting rules |
| FR-10 | Appointment Reminder | Doctors and Administration | System automatically sends a reminder to doctors and administration staff about upcoming appointments. This could be via email, SMS, or in-app notifications. |
| FR-11 | Appointment Confirmation | Patients and Administration | Once an appointment is booked or changed, the system sends out a confirmation to both the patients and administration |
| FR-12 | Appointment Waitlist | Patients | If preferred appointment slots are unavailable, patients can opt for a waitlist. If a slot opens due to cancellations, the patient gets notified for potential booking. |
| FR-13 | Approve requested appointments and changes. | Administration | Admin staff can review, and sanction requested appointments. Considerations include doctor availability, patient's medical needs, and existing appointments. |
| FR-14 | View patients’ appointment history | All users | Every user can access a patient's past appointment records. Useful for patients' personal records, doctors' reference, and tracking no-shows or cancellations. |

### User Case Diagram



### User story

#### 3.3.3.1 Add an appointment.

As a patient, I want to add an appointment with a specific doctor so that I can receive necessary medical care.

Acceptance Criteria:

• The patient can search for and select a specific doctor.

• The patient can view the available slots and select a suitable time.

• The system confirms the booking.

#### ****Edit an appointment.****

As a user, I want to change the details of an existing appointment, adhering to healthcare setting rules, so that I can accommodate changes in my schedule or medical needs.

Acceptance Criteria:

• The user should be able to modify details of an existing appointment.

• Changes should be in line with the healthcare setting's rules and regulations

#### Appointment Reminders

As a user, I want to receive automatic reminders for my upcoming appointments so that I can ensure I don't miss any important dates.

Acceptance Criteria:

* The patient should be able to receive reminders via the application or email.
* The patient should be able to customize the timing and frequency of these reminders.

#### Appointment confirmation

As a patient or admin staff, I want to receive confirmation when an appointment is booked or changed, to ensure accurate scheduling and effective communication.

Acceptance Criteria:

• The system should send out a confirmation when an appointment is booked or modified.

#### Appointment Waitlist

As a patient, I want to be placed on a waitlist if my preferred appointment slot is unavailable, so that I can book it if it becomes available.

Acceptance Criteria:

• The patient should be able to opt for a waitlist for unavailable slots.

• The patient should receive a notification if the slot opens due to cancellations.

#### Approve Requested Appointments and Changes

As an admin staff, I want to review and approve requested appointments or changes, considering doctor availability and patient's medical needs, to ensure effective healthcare delivery.

Acceptance Criteria:

• The admin staff should be able to review and approve requested appointments or changes.

#### Patients’ Appointment History

As a user, I want to access a patient's past appointment records for personal record-keeping, medical reference, and tracking of no-shows or cancellations.

Acceptance Criteria:

• Every user should be able to access a patient's past appointment records.

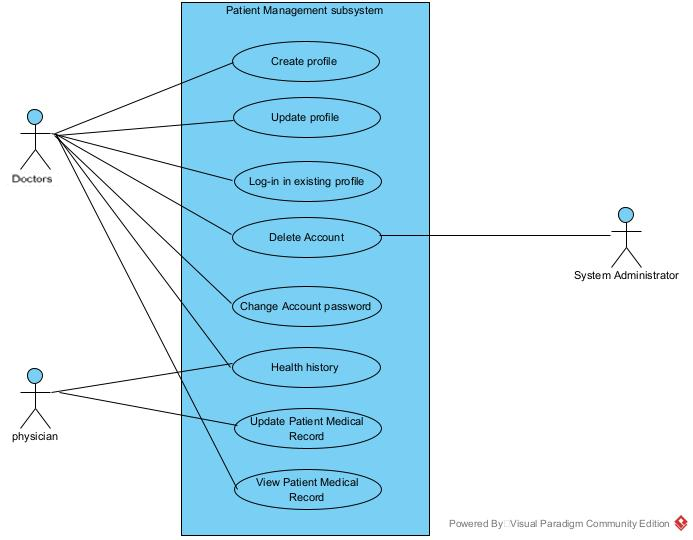
## Subsystem 4: Patient Management Profile

### Goal Use Cases

| FR # | Goal Use Case | Role Player | Description |
| --- | --- | --- | --- |
| FR-15 | Create profile | Patients | Patients can set up their account, providing necessary details such as name, contact info, medical history, etc. |
| FR-16 | Update profile | Patients | Patients can edit their profile information, allowing for current and accurate details to be reflected. |
| FR-17 | Log-in in existing profile | Patients | Patients can securely access their existing profile using their credentials. |
| FR-18 | Delete Account | Patients and System administration | Patients or system administrators can remove a patient's account from the system when necessary, following appropriate procedures. |
| FR-19 | Change Account password | Patients | Patients can update their password for added security or in case they forget their current one. |
| FR-20 | Health history | Patients and Doctors | Patients can view their health history, and doctors can access this information for better medical understanding and treatment planning. |
| FR-21 | Update Patient Medical Record | Doctors | Doctors must be able to update the patient's health history based on test results and medical analysis. |
| FR-22 | View Patient Medical Record | Patients | Patients can view their medical record. |

### User Case Diagram

### User Story



#### 3.4.5.1 Create profile.

As a patient, I want to create a profile by providing necessary details so that I can have a personalized experience and manage my appointments efficiently.

Acceptance Criteria:

* The patient must be able to create an account by supplying information about themselves, including their name, contact information, medical background, etc.

• The system must verify that the account was successfully created.

#### Update profile

As a patient, I want to edit my profile information so that my current and accurate details are reflected in the system.

Acceptance Criteria:

• The patient should be able to update their profile information.

• The system should confirm the updates made to the profile.

#### Log-in in existing profile

As a patient, I want to securely access my existing profile using my credentials so that I can manage my account and appointments.

Acceptance Criteria:

• The patient should be able to log into their profile using their credentials.

• The system should verify the credentials and grant access to the account.

#### Delete Account

As a patient or system administrator, I want to delete a patient's account when necessary, following appropriate procedures, to maintain data integrity and privacy.

Acceptance Criteria:

• The patient or system administrator should be able to remove a patient's account from the system.

• The system should confirm the account deletion.

#### Change Account password

As a patient, I want to update my password for added security or in case I forget my current one so that I can maintain secure access to my account.

Acceptance Criteria:

• The patient should be able to update their password.

• The system should confirm the password change.

#### Health history

As a patient or doctor, I want to access my health history for better understanding and planning of medical treatment.

Acceptance Criteria:

• The patient should be able to view their health history.

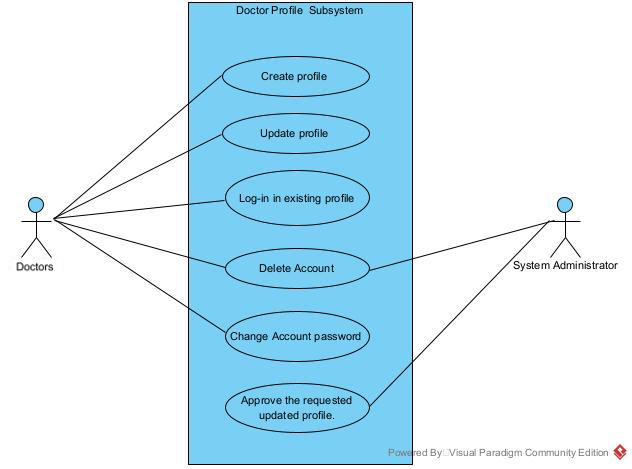
• The doctors should have access to the patient's health history for better understanding and treatment planning.

## Subsystem 4: Doctor Profile

### Goal Use Cases

| FR # | Goal Use Case | Role Player | Description |
| --- | --- | --- | --- |
| FR-23 | Create profile | Doctors | Doctors can establish their profiles, including professional details like qualifications, specialties, years of practice, etc. |
| FR-24 | Update profile | Doctors | Doctors can modify their profile details, keeping them up to date with any changes in their professional status or information. |
| FR-25 | Log-in in existing profile | Doctors | Doctors can securely access their existing profile using their valid credentials. |
| FR-26 | Delete Account | Doctors and System Administration | Doctors or system administrators can delete a doctor's account from the system as required, following proper protocols. |
| FR-27 | Change Account password | Doctors | Doctors have the capability to change their account password to ensure account security or in case of forgetting the current password. |
| FR-28 | Approve the requested updated profile. | System Administration | System administrators have the authority to review and approve changes requested on a doctor's profile to maintain the accuracy and integrity of the system's data |

### User Case Diagram



#### Create profile

As a doctor, I want to create my profile with professional details like qualifications, specialties, and years of practice, to help patients make informed choices and facilitate appointment scheduling.

Acceptance Criteria:

• The doctor should be able to establish their profile by inputting professional details.

• The system should acknowledge the successful creation of the profile.

#### 3.5.5.2 Update profile

As a doctor, I want to be able to modify my profile details to keep them current with any changes in my professional status or information.

Acceptance Criteria:

• The doctor should have the capability to revise the details in their profile.

• The system should verify and save the updates made to the profile.

#### Log-in in to the existing profile

As a doctor, I want to securely sign in to my existing profile using my credentials, so I can manage my appointments and personal account.

Acceptance Criteria:

• The doctor should be able to access their profile using valid credentials.

• The system should authenticate the credentials and grant access to the doctor's profile.

#### Delete account

As a doctor or a system administrator, I want to erase a doctor's account from the system when needed, adhering to the right procedures, to maintain data integrity and privacy.

Acceptance Criteria:

• The doctor or the system administrator should be able to delete a doctor's account from the system.

• The system should confirm the removal of the account.

#### Change Account password

As a doctor, I want to change my account password to ensure the security of my account or in case I forget my existing password.

Acceptance Criteria:

• The doctor should have the ability to update their password.

• The system should validate and acknowledge the change in password.

#### Approve the requested updated profile

As a system administrator, I want to review and approve modifications requested on a doctor's profile to maintain the reliability and validity of system data.

Acceptance Criteria:

• The system administrator should have the authority to review and sanction changes to a doctor's profile.

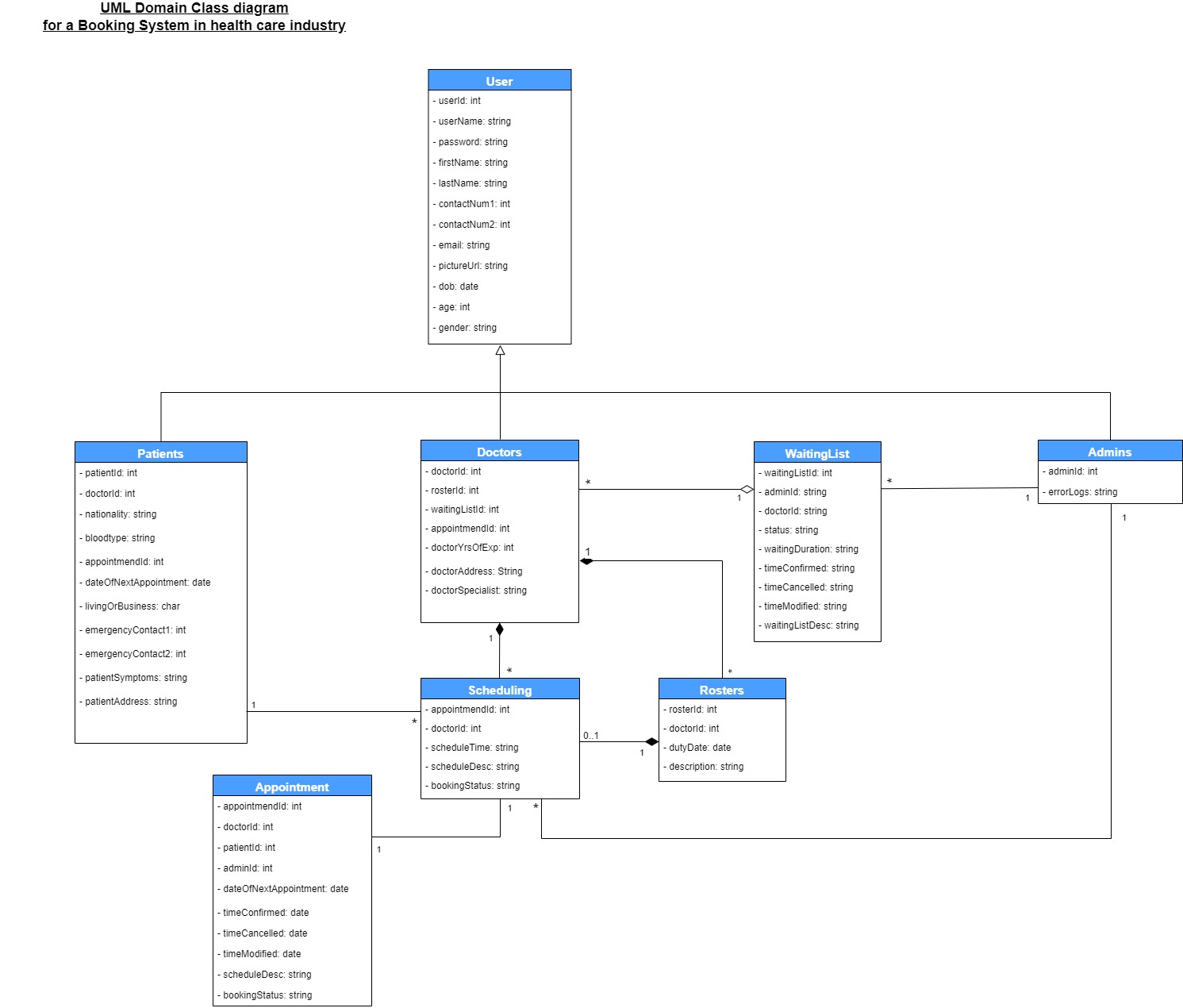
• The system should confirm the approval of the updated profile.

# Section 4: UML Domain Class Diagram

## A List of the classes:

* **'User' class:** It represents individuals who utilize the system. The class contains attributes that include basic information of the users such as username and password.
* **'Patients' class:** It inherits attributes from the 'User' class and contains more detailed attributes to represent patients such as 'patientSymptoms' and ‘patientMedRecords’.
* **'Doctors' class:** It inherits attributes from the 'User' class and contains more detailed attributes to represent doctors such as 'doctorSpecialist' and 'doctorYrsOfExp'.
* **'Rosters' class:** It represents rosters of 'Doctors' in 'Clinics' class. This class is an association class.
* **'Admins' class:** It inherits attributes from the 'User' class and contains more detailed attributes to represent admins such as 'errorLogs'.
* **'WaitingList' class:** It represents a waiting list for doctors’ identity verification. This class is managed by 'Admins' class which will verify the doctor’s identity before adding to, updating, or removing from the list of doctors available in the system.
* **'Appointment' class:** It represents booked sessions between 'Patients' and 'Doctors'. This class is for making, searching, editing, and canceling appointments.
* **'Scheduling' class:** It represents all the available timeslots of the 'Doctors' class for users to pi

## Domain class diagram:



# Section 5: ERD Diagram

## A List of All Entities:

* **User:** It represents the common field that system user may have such as all users will need to provide their name, email, and gender etc.
* **Patient:** It represents the unique patient information including emergency contact and blood type.
* **Doctor:** It represents the unique doctor information including doctor roster id, doctor status, and their feedbacks.
* **Admin:** It represents the unique admin information such as the error logs information.
* **Appointment:** It represents the appointment information including patient id, doctor id, appointment status, and time etc.
* **MedicalRecord:** It represents the information of patient medical history when they start using the system.
* **WaitingList:** It represents the information for upcoming doctors who need to create an account to access the system.
* **clinics:** It represents the information of clinics where doctors work including the address, email, and phone number.
* **roster:** It is an associate entity between doctors and clinics due to a many-to-many relations

## ERD Diagram:

**A computer screen shot of a computer

Description automatically generated**

# Section 6: UML Systems Sequence diagram

## Introduction of the diagram

* **Create Profile :** This diagram refers to the FR-15 for the patient set up their account, providing necessary details such as name, contact info, medical history, etc.
* **Update Patient Medical record :** This diagram refers to the FR-21 for the doctors so that they are able to update the patient's health history based on test results and medical analysis.

## A diagram of a project Description automatically generatedUML Systems Sequence diagram: Create Profile

## UML Systems Sequence diagram:Update Patient Medical record

**A diagram of a project

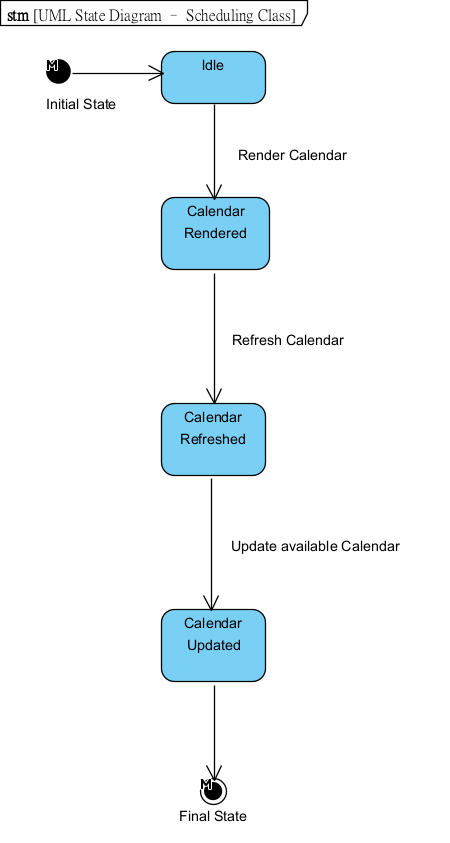
Description automatically generated**

# Section 7: UML State Diagrams

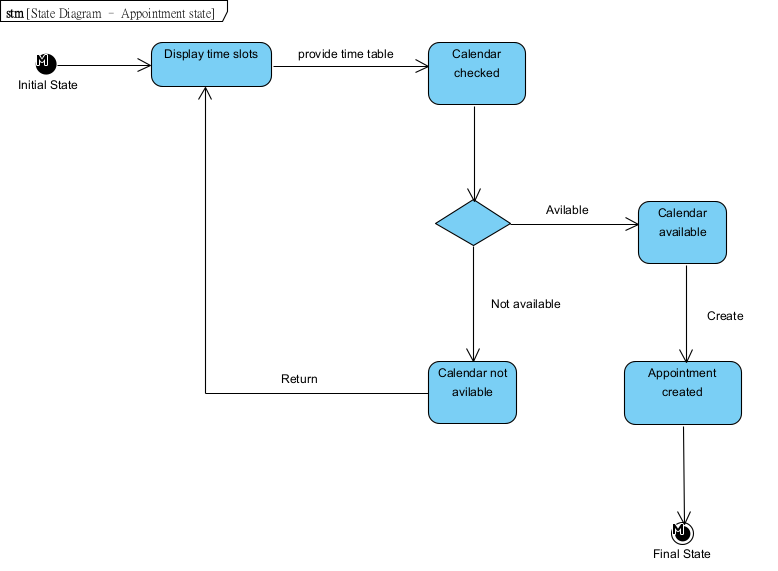
## Introduction of the diagram

* **Scheduling Class object:** This diagram shows the process of how scheduling system work by changing the state and generate the updated calendar and display to the user.
* **Appointment Class object:** This diagram shows the process of how appointment system work by accepting the scheduling class result and create the appointment.

## UML State Diagrams: Scheduling Class object



## UML State Diagrams: Appointment Class object



# Section 8: Technologies

## Relevant technologies: Web Application

We chose to develop our system as a web application initially because our college coursework has primarily focused on PC-based applications rather than mobile ones. However, the system can still be accessed on mobile devices, but the graphical user interface (GUI) will be designed for web use with a smaller resolution.

**Client-side (Web Application):**

The client-side focus on the UI/UX to provide the best experience for the user. Therefore, some web development framework will be adopted. Below are the possible tools that may be used:

1. **Angular**
2. **React**
3. **jQuery**
4. **Html**
5. **CSS**
6. **VS Code**

**Business Logic (Web Application):**

Business Logic focuses on how the system provides service to the user including the creation of the user account, searching feature, and booking an appointment. It acts as the bridge between the database and client-side. Therefore, programming languages will be the main tools. Below are the possible tools the system may be used.

1. **RESTful API**
2. **Java**
3. **Nodejs**

**Data side (Web Application):**

The data side is focused on storing the information across the system including all user information and system performance etc. Therefore, DBMS will be the tools, and below are some possible DBMS that may be used in the system.

1. **MongoDB**
2. **MySQL**
3. **Oracle SQL**

# Section 9: Gantt Chart

# Reference:

*Medical banner with stethoscope Free Photo*. (2022, August 11). Freepik. https://www.freepik.com/free-photo/medical-banner-withstethoscope\_30555914.htm#query=health& position=3&from\_view=search&track=sph

Meditab. (n.d.). *Appointment Booking System | Meditab*. https://www.meditab.com/products/appointment-booking-system

*EDIT.org - online editor*. (n.d.). EDIT.org. https://edit.org/edit#