

Hospital Management System

Introduction:

Hospital Management System is used to take the data from the patients and then store it for later use. The main goal of the Hospital Management System is to accurately treat as well as decrease overtime pay.

There are various features included in the HMS. Some of the system functions include Registration, Patient check out, Report generation, and more. In this documentation, let us check out the functional and non-functional requirements of the Hospital Management System in depth.



1.a.Functional Requirements:

There are many software requirements specifications included in the functional requirements of the Hospital Management System, which contains various process, namely Registration, Check out, Report Generation, and Database.

Sign In Process of SRS (Software Requirements Specification)

- Login to System: The Hospital Management enables the user (doctor, patient, staff, admin) enter into System and the user choose his dashboard.

Admin Process of SRS (Software Requirements Specification)

- Add User (Patient, Doctor): The Hospital Management enables the Admin to include new patients/Doctors to the system.
- Display for Doctor / Patient: The Hospital Management enables the Admin to display all doctor/patient that included in the system file.
- Search Doctor / Patient: The Hospital Management enables the Admin to search for specific doctor/patient by id.
- Delete The Doctor: The Hospital Management enables the Admin to delete specific doctor from the system.
- Update Doctor / Patient: The Hospital Management enables the Admin to update specific doctor/patient.

Doctor Process of SRS (Software Requirements Specification)

- Make Report: The Hospital Management enables Doctor will write a report for his patient status.
- Show Message History of Visitor: The Hospital Management enables Doctor to show message history of visitor patient.
- Show Message History of Resident: The Hospital Management enables Doctor to show message history of Resident patient.
- Display All Patients: The Hospital Management enables doctor to show all patients with their information from file system.

Staff Process of SRS (Software Requirements Specification)

- Display All Appointment: The Hospital Management enables staff member to enable doctor/patient to see the appointments.
- Display All Time Table: The Hospital Management enables staff shows the available time to patient from doctor timetable.
- Filter Appointment: The Hospital Management enables staff to filter the doctor timetable by doctor id and time.
- Book Appointment for Visitor: The Hospital Management enables staff to book an appointment to the visitor patient.
- Book Appointment for Resident: The Hospital Management enables staff to book an appointment to the resident patient who set in the hospital for a long time.

Patient Process of SRS (Software Requirements Specification)

- Show All Time Table: The Hospital Management enables patient shows the available time from the system
- Show My Report: The Hospital Management enables patient to see his report after doctor appointment.

Check Out of SRS:

- Deleting Patient ID: The Admin can delete the patient ID from the system when the patient's checkout from the hospital.
- Adding new bed: The Admin can add bed.

Report Generation of SRS:

- Information of the Patient: The Hospital Management System generates a report on every patient regarding various information time table and his history and appointment.
- Availability of the Bed: The Hospital Management system also helps in generating reports on the availability of the bed regarding the information like bed number unoccupied or occupied, ward name, and more.

File-Based-System of SRS:

- Mandatory Patient Information: Every patient has some necessary data like doctor first and last name, 'doctor's ID number , message, message-history ,Time(the patient appointment) , UserName , Password , Age , etc.
 - Updating information of the Patient: The hospital management system enables users to update the information of the patient as described in the mandatory information included.
-

1.b.Non-Functional Requirements

There are many software requirements specifications included in the non-functional requirements of the Hospital Management System, which contains various process, namely Security, Performance, Maintainability, and Reliability.

Security:

- Patient Identification: The system needs the patient to recognize herself or himself using the phone.
- Logon ID: Any users who make use of the system need to hold a Logon ID and password.
- Modifications: Any modifications like insert delete, update, etc. for the database can be synchronized quickly and executed only by the ward administrator.

- Front Desk Staff Rights: The staff in the front desk can view any data in the Hospital Management system, add new patient's record to the HMS but they do not have any rights alter any data in it.

- Administrator rights: The administrator can view as well as alter any information in the Hospital Management System.

Performance:

- Response Time: The system provides acknowledgment in just one second once the 'patient's information is checked.

- Capacity: The system needs to support at least 1000 people at once.

- User-Interface: The user interface acknowledges within five seconds.

- Conformity: The system needs to ensure that the guidelines of the Microsoft accessibilities are followed.

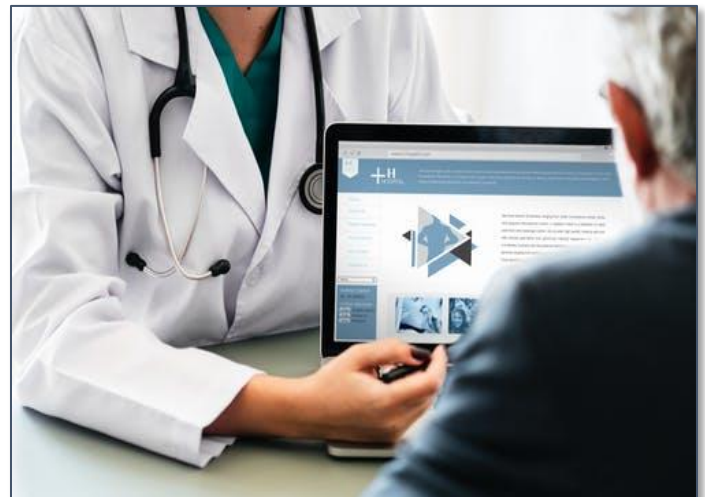
Maintainability:

- Back-Up: The system offers the efficiency for data backup.

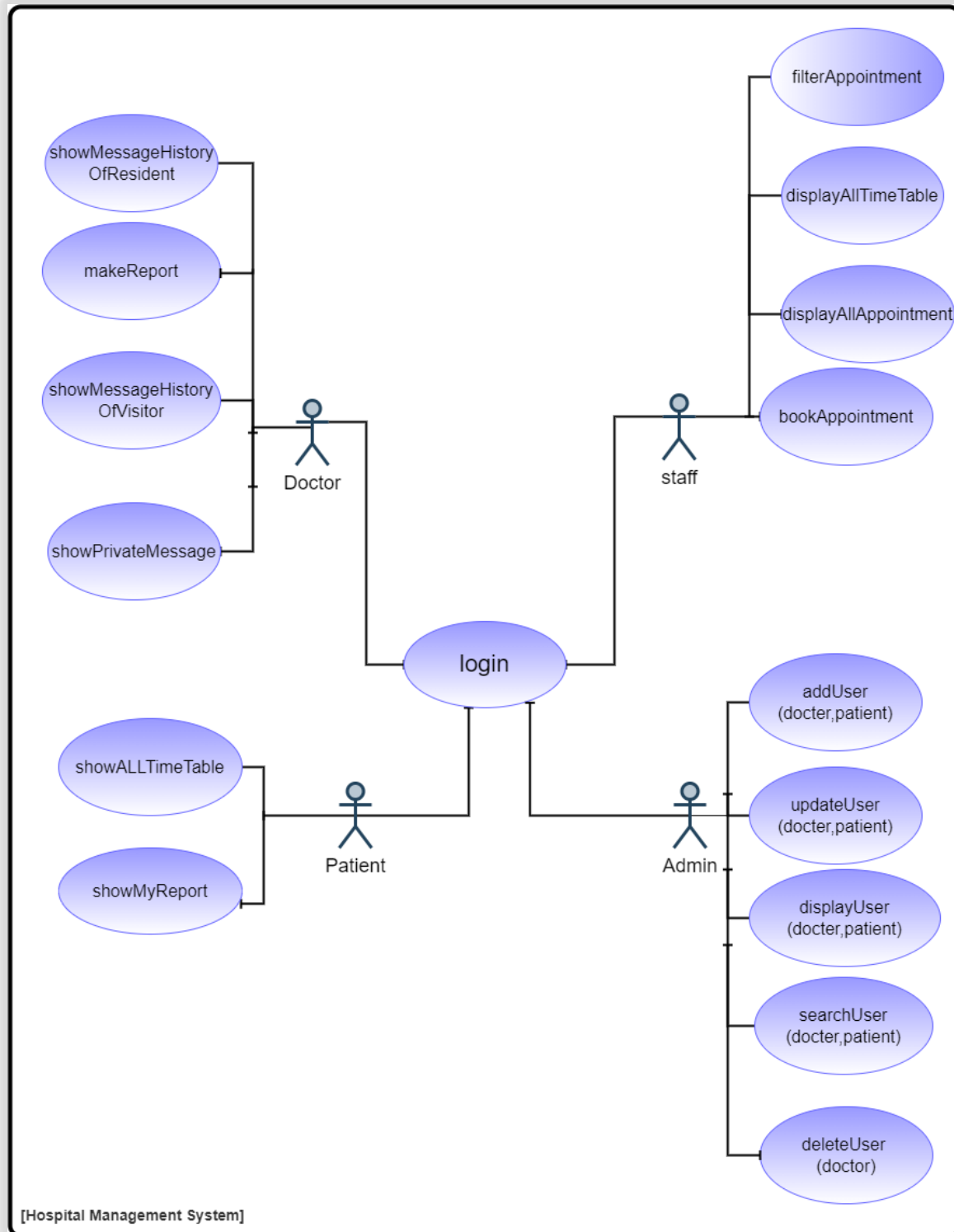
- Errors: The system will track every mistake as well as keep a log of it.

Reliability:

- Availability: The system is available all the time.



2-usecase diagram



2-usecase scenarios

1-Use case scenario for login to the system

Use case name : Login

Goal: System validates the user

Actor: Staff/Admin/Patient/Doctor

Pre-condition: user must exist in file-based system and
Login page must be displayed

Post condition: user logged in successfully
(username and password of the user is Validated)

Main success scenario:

- user home page force him to make login not to choose.
- user Writes his email and password.
- user clicks on login.
- user logged in successfully.

Extensions:

- 1.a. his email or password isn't correct
 - 1.a.1. System alert him to write the right email and right password and show message "invalid user ..."
- 1.b. the user is an admin
 - 1.b.1. the system opens admin dashboard
- 1.c. the user is a Staff
 - 1.c.1. the system opens the system Staff dashboard
- 1.d. the user is a patient
 - 1.d.1. the system opens the system home page
- 1.e. the user is a doctor
 - 1.e.1 the system opens doctor page

2-Use case scenario for create patient report

Use case name : makeReport

Goal: System validates the patient report

Actor: Doctor

Pre-condition: doctor must be logged in
and opened his home page

Main success scenario:

- Doctor log in successfully.
- Doctor choose to click on my patient.
- Doctor start to write the patient report.
- Doctor click on save.

Post condition: report made successfully

3-Use case scenario for show message history of Resident patient to the system

Use case name : showMHistoryOfResident

Goal: System review message history of patient

Actor: Doctor

Pre-condition: patient/doctor must exist in file based system and Login page must be displayed to the doctor

Main success scenario:

- After login doctor will enter into his page.
- Then click into resident patient history.
- Then the doctor will write the patient id.
- the system display the history of resident patient successfully.

Post condition: Doctor logged in successfully and choose the doctor page
(username and password of the user is Validated)

Extensions:

1.a. the resident patient id that the doctor enter isn't correct.

1.a.1. System alert him to write the right resident patient id and show message "invalid patient id ..."

4-Use case scenario for show message history of visitor patient to the system

Use case name : showMHistoryOfVisitor

Goal: System review message history of patient

Actor: Doctor

Pre-condition: patient/doctor must exist in file based system and Login page must be displayed to the doctor

Main success scenario:

- After login doctor will enter into his page.
- Then click into visitor patient history.
- Then the doctor will write the patient id.
- the system display the history of resident patient successfully.

Post condition: Doctor logged in successfully and choose the doctor page
(username and password of the user is Validated)

Extensions:

- 1.a. the visitor patient id that the doctor enter isn't correct.
- 1.a.1. System alert him to write the right visitor patient id and show message "invalid patient id ..."

5-Use case scenario for show private message from patient

Use case name : showPrivateMessage

Goal: System display the private message of the patient

Actor: Doctor

Pre-condition: patient/doctor must exist in file based system and Login page must be displayed to the doctor

Main success scenario:

- After login doctor will enter into his page.
- Then click into private message and write message.
- Then the message will be displayed to the patient from his doctor.

Post condition: Doctor logged in successfully and choose the doctor page
(username and password of the user is Validated)

Extensions:

1.a. the name of patient that the doctor enter isn't correct.

1.a.1. System alert him to write the right name and show message "invalid patient name ..."

6-Use case scenario for show all time table of patient to the system

Use case name : showAllTimeTable

Goal: System review the time table of patient

Actor: Patient

Pre-condition: patient must exist in file based system and
Login page must be displayed to the patient

Main success scenario:

- After login patient will enter into patient page.
- Then click into show all timetable.
- The system display the timetable of patients successfully.

Post condition: patient logged in successfully and choose the patient
page
(username and password of the user is Validated)

Extensions:

1.a. the visitor patient id that the doctor enter isn't correct.

1.a.1. System alert him to write the right visitor patient id and show
message "invalid patient id ..."

7-Use case scenario for show patient report to the system

Use case name : showMyReport

Goal: System review the patient report

Actor: Patient

Pre-condition: patient must exist in file based system and
Login page must be displayed to the patient

Main success scenario:

- After login patient will enter into patient page.
- Then click into show my report.
- The system display the report of patient successfully.

Post condition: patient logged in successfully and choose the patient
page
(username and password of the user is Validated)

Extensions:

1.a. the email or password that patient enter isn't correct.

1.a.1. System alert him to write the right email or password and show
message "invalid Email/password ..."

8-Use case scenario to Filter Appointment on the system

Use case name : filterAppointment

Goal: System filter the doctor timetable

Actor: Staff

Pre-condition: Staff must be logged in

Main success scenario:

- After login staff will enter into his page.
- Then click into filter appointment.

Post condition: Appointments are filtered in the system

Extensions:

1.a. the email or password that patient enter isn't correct.

1.a.1. System alert him to write the right email or password and show message "invalid Email/password ..."

9-Use case scenario to enable staff to Display All Appointment to the(doctor/patient)

Use case name : displayAllAppointment

Goal: System enable Doctor/patient to see appointments

Actor: Staff

Pre-condition: Staff must be logged in

Main success scenario:

- After login staff will enter into his page.
- Then click into Display All Appointment.
- Staff display all appointment to the (doctor/patient)
- **Post condition:** staff display all appointment

Extensions:

- 1.a. the email or password that patient enter isn't correct.
- 1.a.1. System alert him to write the right email or password and show message "invalid Email/password ..."

-

10-Use case scenario to Display All Time Table

Use case name : displayAllTimeTable

Goal: staff shows the available time to patient

Actor: Staff

Pre-condition: Staff must be logged in

Main success scenario:

- After login, staff will enter into his page.
- Then click into Display All Time Table.
- Staff display all available time to the patient

Post condition: staff display all available time to patient from doctor timetable

Extensions:

1.a. the email or password that patient enter isn't correct.

1.a.1. System alert him to write the right email or password and show message "invalid Email/password ..."

11-Use case scenario for booking appointment .

Use case name : bookAppointment

Goal: Staff book an appointment of patient

Actor: Staff

Pre-condition: Staff must exist in file based system and
Login page must be displayed to the Staff

Main success scenario:

- After login Staff will enter into his page.
- Then click into book an appointment.
- Then the Staff will book an appointment by entering patient id.
- the system book an appointment for patient successfully.

Post condition: an appointment of patient is booked successfully.

Extensions:

1.a. the patient id that the Staff enter isn't correct.

1.a.1. System alert him to write the right patient id and show message
"invalid patient id ..."

12-Use case scenario for add user (doctor/patient) .

Use case name : addUser(doctor/patient)

Goal: Admin add new user into the system

Actor: Admin

Pre condition: Admin already logging in and want new user

Main success scenario:

- After login Admin will enter into his page.
- If Admin want to add new patient into the system Then he will click into patient page.
- Next Admin will choose what he want to add(addnew resident patient / add new visitor patient)
- Admin then is to write the full information of the new patient and “Submit”
- But, If Admin want to add new doctor into the system he will click on doctor page.
- Next Admin will choose add new doctor.
- Admin then is to write the full information of the new doctor and “Submit”

Post condition: adding new user (doctor/patient) successfully

13-Use case scenario to Update User(doctor/patient)

Use case name : UpdateUser(Doctor/Patient)

Goal: Admin update details of the (doctor/patient)

Actor: Admin

Pre-condition: Admin must be logged in

Main success scenario:

- After login admin will enter into his page.
- Then click into Update Doctor / Patient.
- specific Doctor /patient are updated

Post condition: Admin Update specific details of Doctor /patient

Extensions:

- The data is incorrect
- The system displays" updating error" message

14-Use case scenario for show admin all user(patient/doctor) in the system

Use case name : displayUser(doctor/patient)

Goal: Admin see all users(patients/doctors)

Actor: Admin

Pre-condition: patient/doctor must exist in file based system and Admin must Login to the system and show it is page.

Main success scenario:

- After login Admin will enter into his page.
- Then click into page of doctor/patient.
- Then the Admin click into display doctor/patient.
- the system display the doctor/patient information successfully.

Post condition: Admin logged in successfully and choose the doctor/patient page then Display all doctors/patients.
(username and password of the user is Validated)

Extensions:

1.a. the resident patient id that the Admin enter isn't correct.

1.a.1. System alert him to write the right Admin id and show message "invalid Admin id ..."

15-Use case scenario for search of users(patient/doctor) in the system by id

Use case name: searchUser(doctor/patient)

Goal: Admin search for users(patients/doctors) by id

Actor: Admin

Pre-condition: patient/doctor must exist in file-based system and its id is correct and Admin must Login to the system and show it is page.

Main success scenario:

- After login Admin will enter his page.
- Then click into page of doctor/patient.
- Then the Admin click into search doctor/patient.
- the system displays the matched doctor/patient information successfully.

Post condition: Admin logged in successfully and choose the doctor/patient page then search doctors/patients.
(Username and password of the user is Validated)

Extensions:

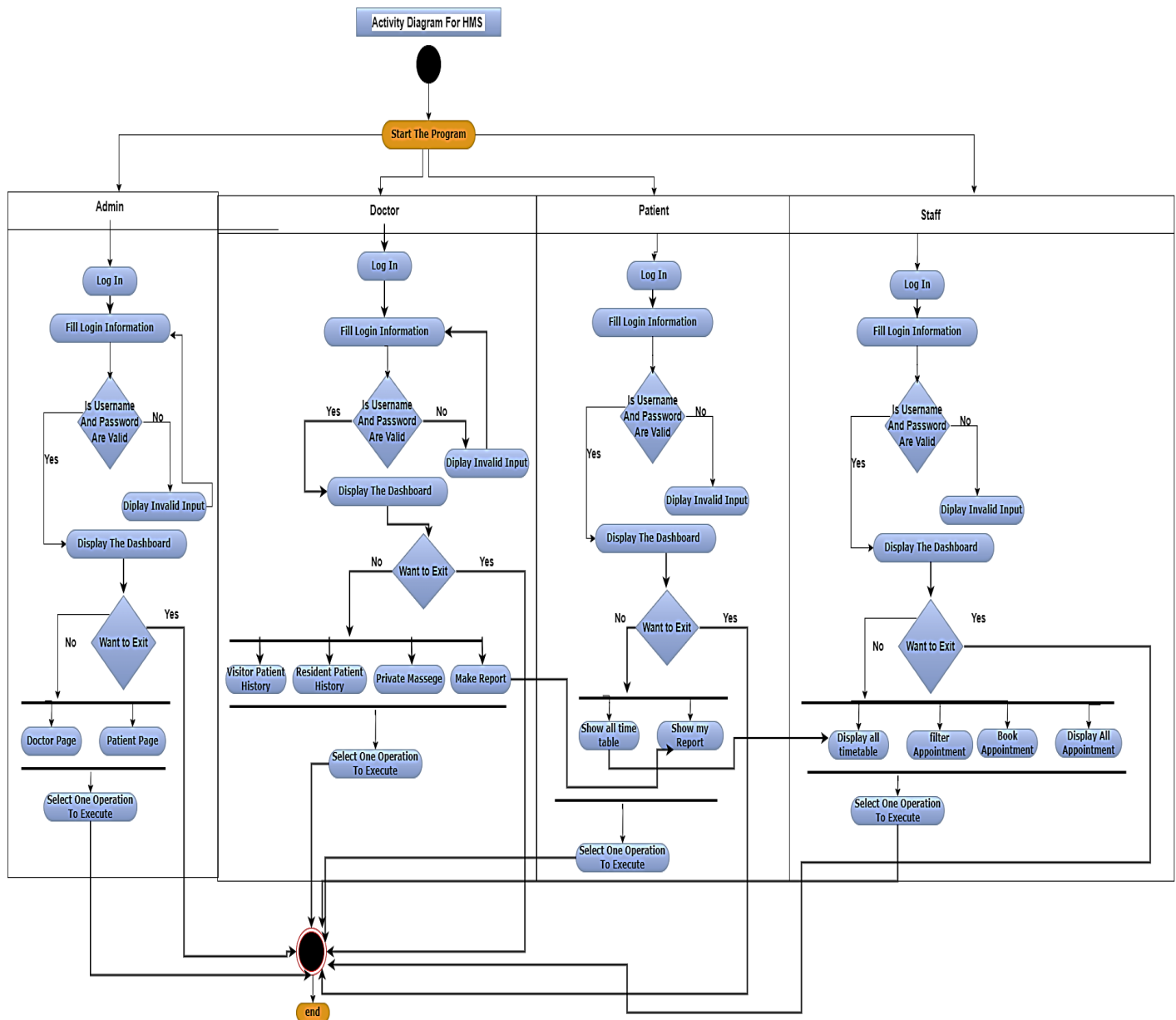
1.a. the resident patient id that the admin enter isn't correct.

1.a.1. System alert him to write the right Admin id and show message "invalid Admin id ..."

16-Use case scenario for delete doctor/resident patient from the system

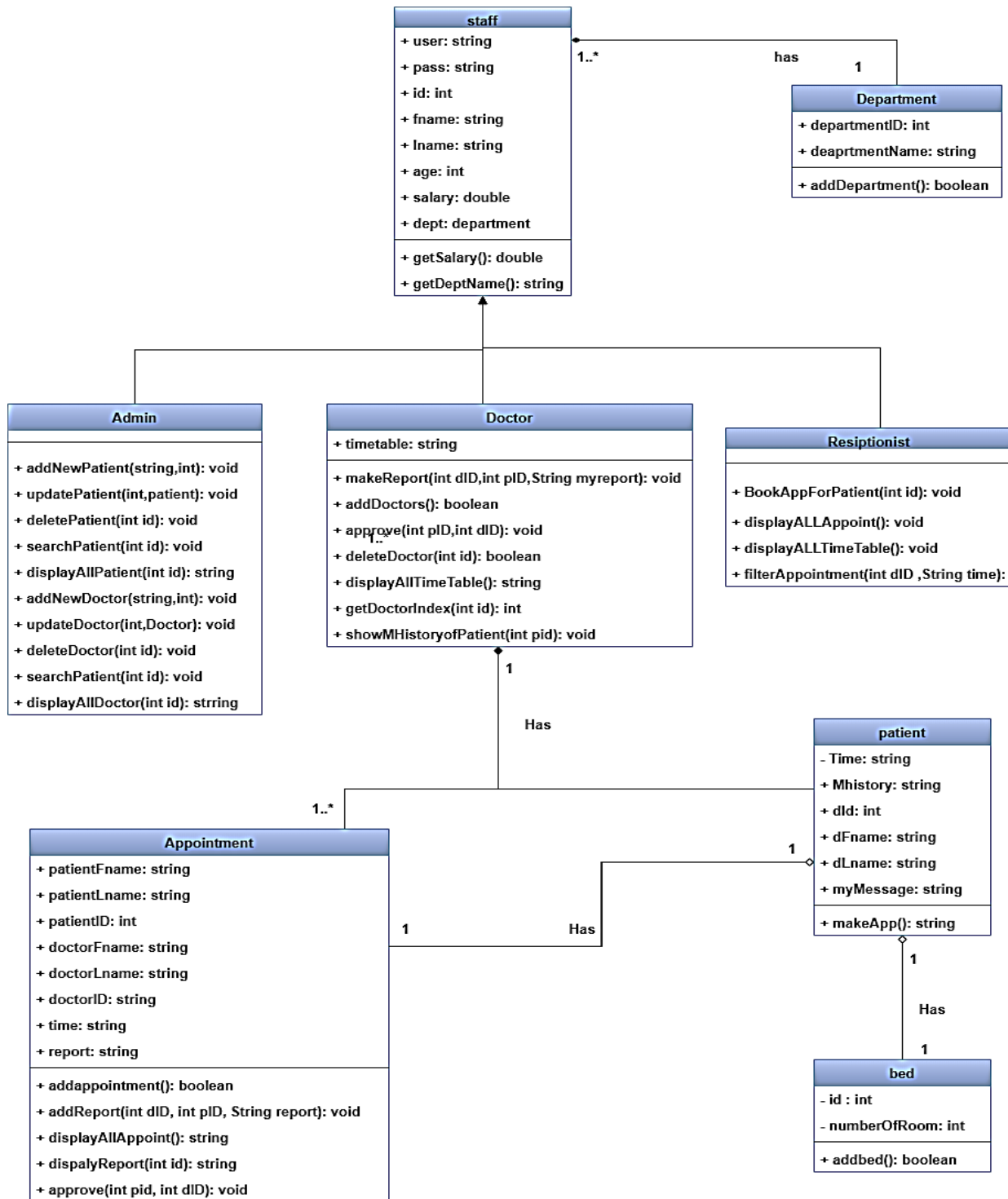
Use case name:	deleteUser (doctor/ resident patient)
Goal:	Admin delete users (doctor/ resident patient) from the system
Actor:	Admin
Pre-condition:	patient/doctor must exist in file-based system and Admin must Login to the system and show it is page.
Main success scenario:	<ul style="list-style-type: none">- After login Admin will enter his page.- Then click into page of doctor/patient.- Then the Admin click into delete doctor/resident patient.- the system deleted the matched doctor/patient information successfully.
Post condition:	Admin logged in successfully and choose the doctor/patient page then delete doctors/resident patient. (Username and password of the user is Validated)
Extensions:	<p>1.a. the resident patient id that the admin enter isn't correct.</p> <p>1.a.1. System alert him to write the right Admin id and show message "invalid Admin id ..."</p>

3-Activity Diagram

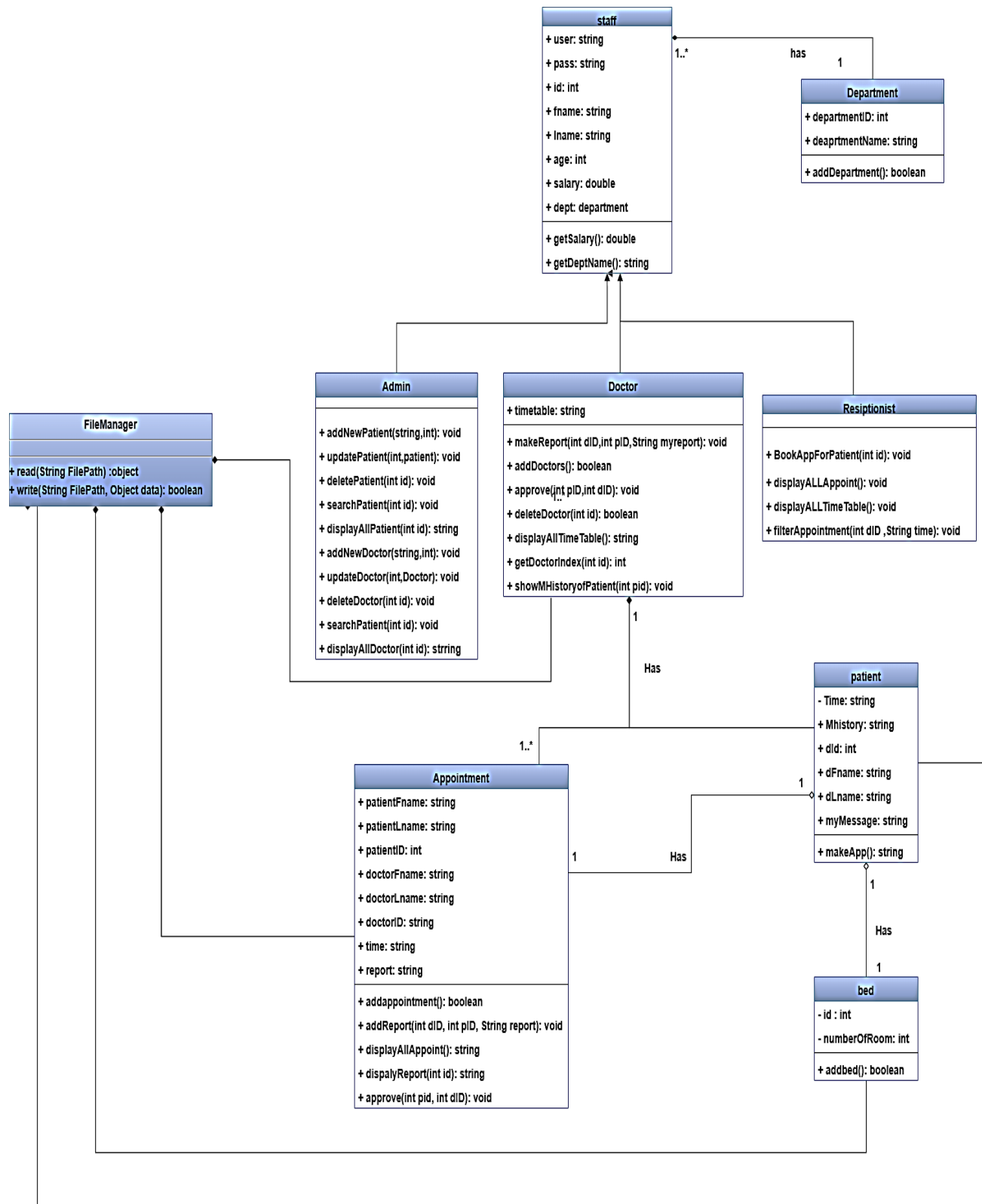


4-UML class diagram:

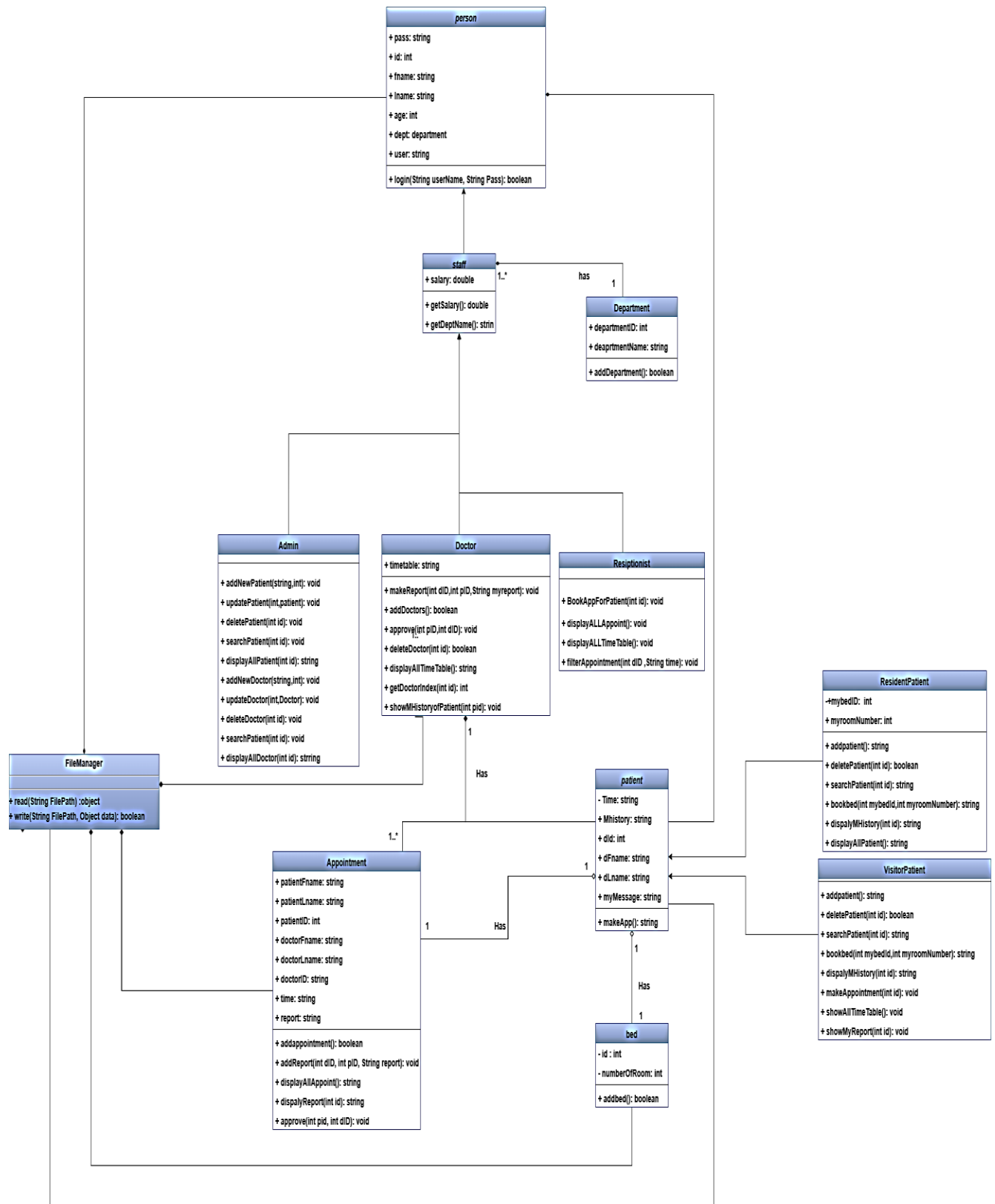
Version 1:



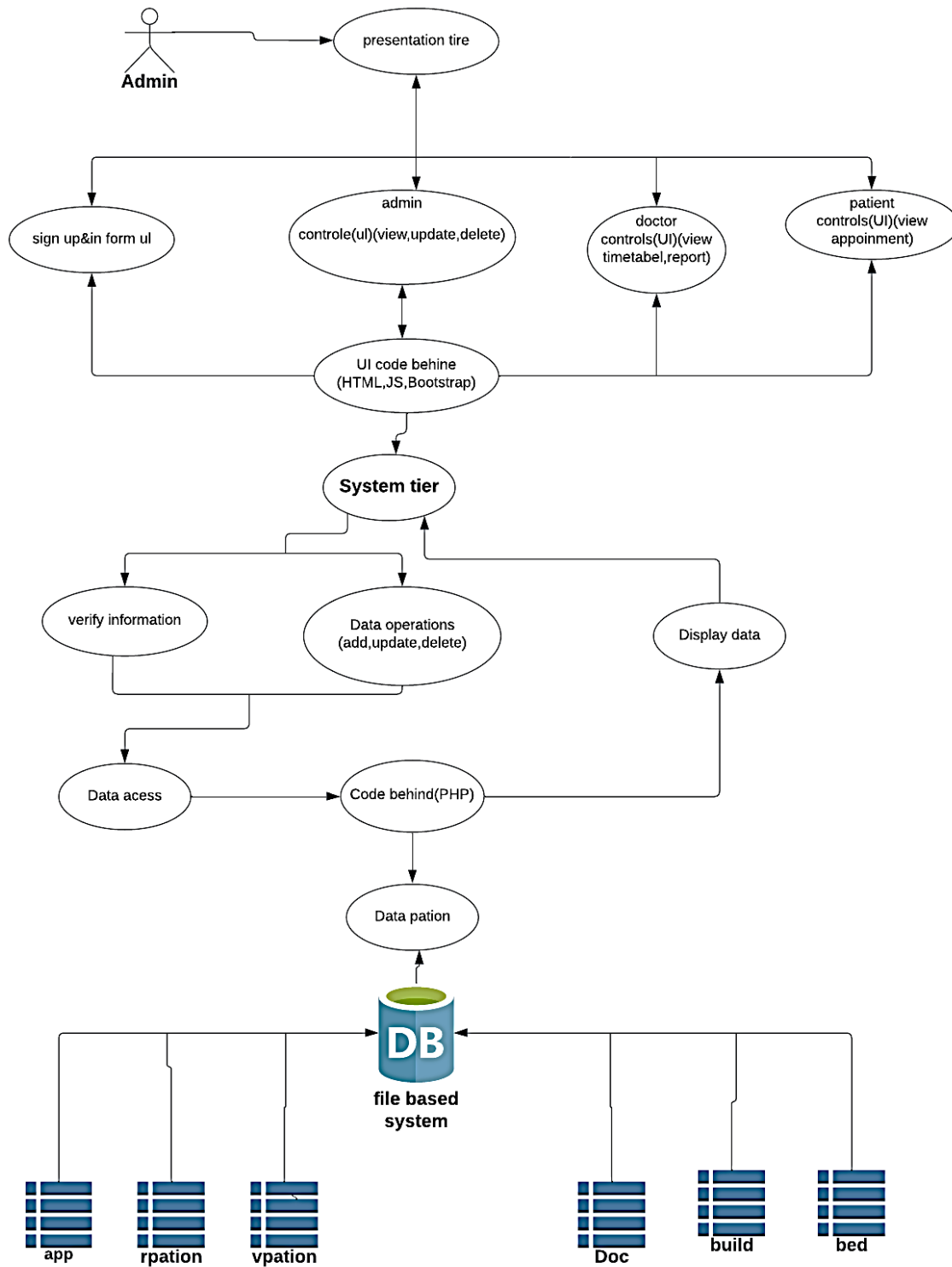
Version 2:



Version 3:

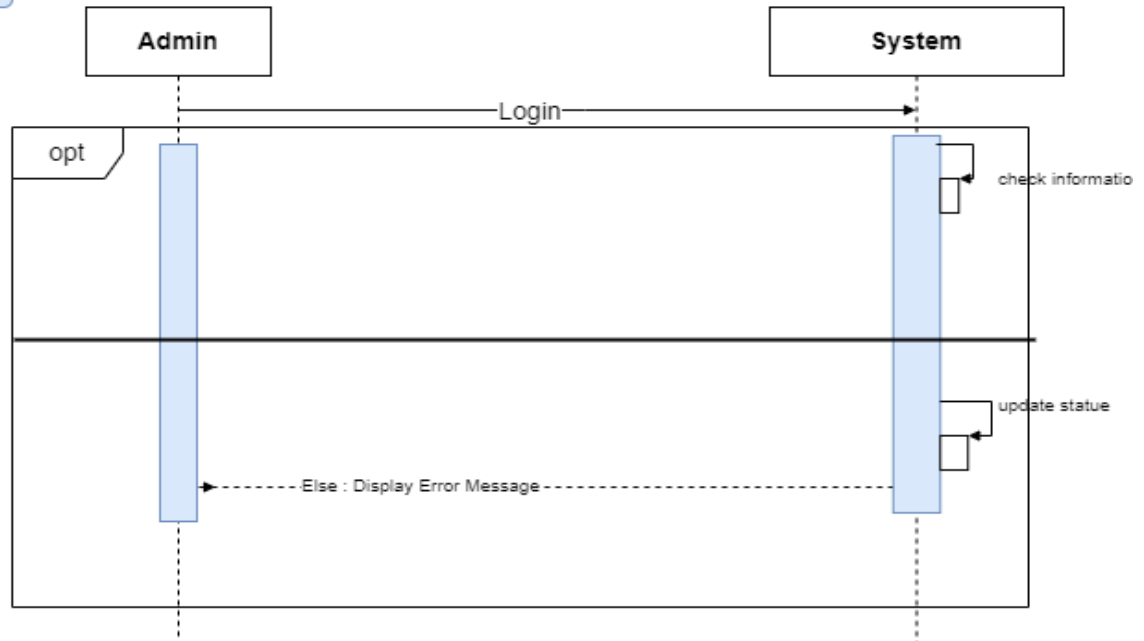


5-System Architecture:

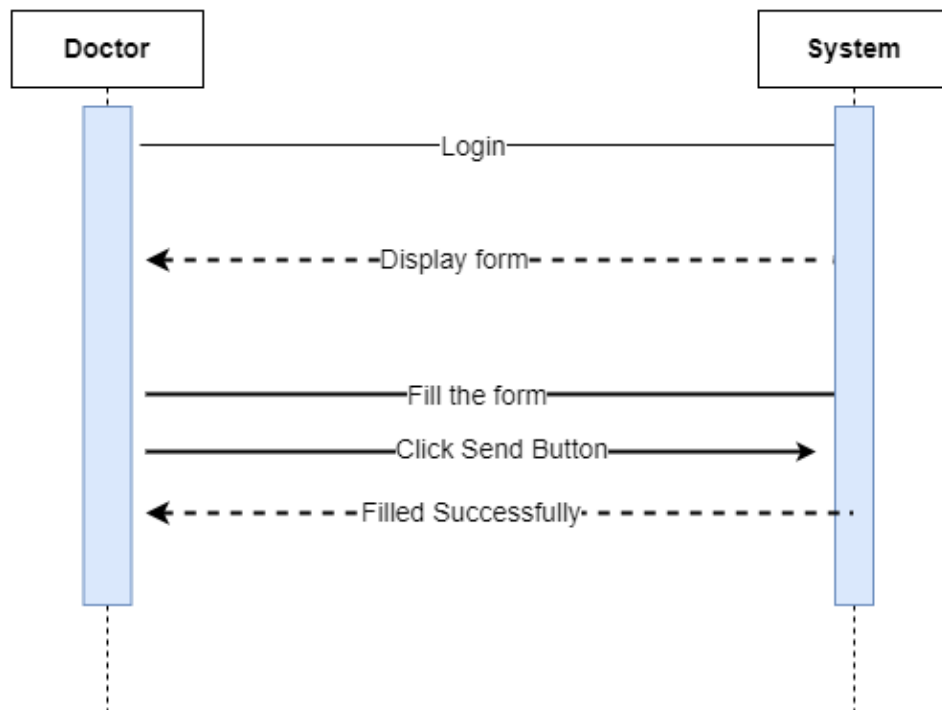


6-Sequence Diagram

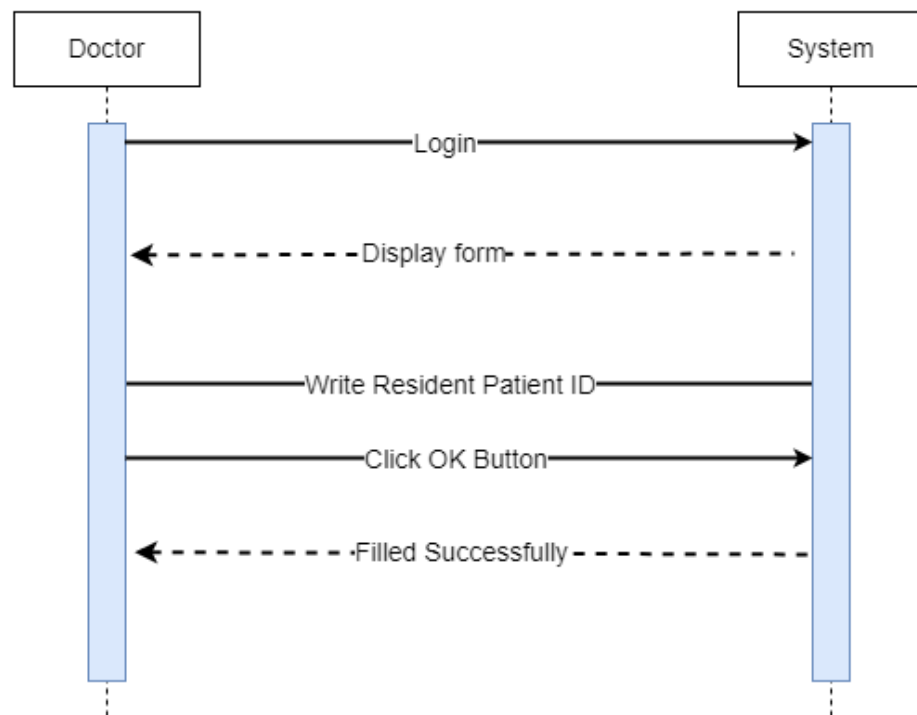
Login



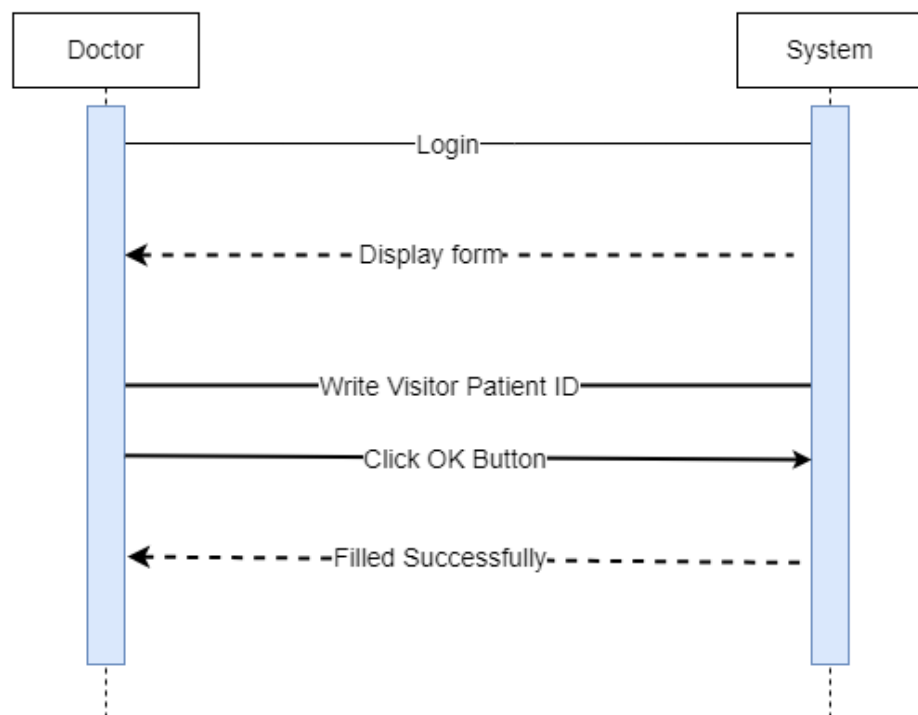
Make Report



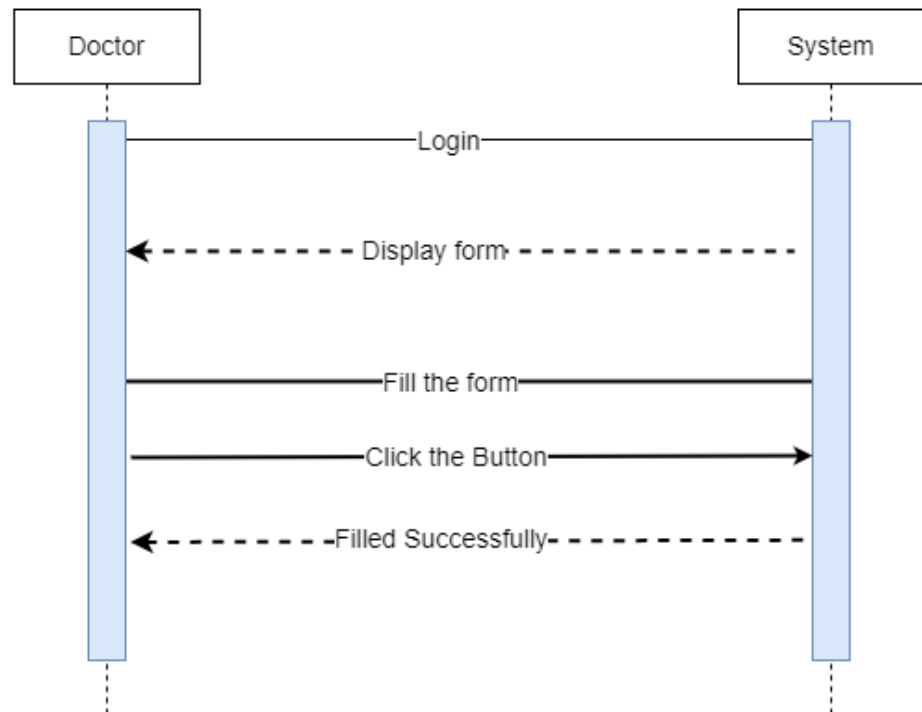
Show Message History of Resident Patient



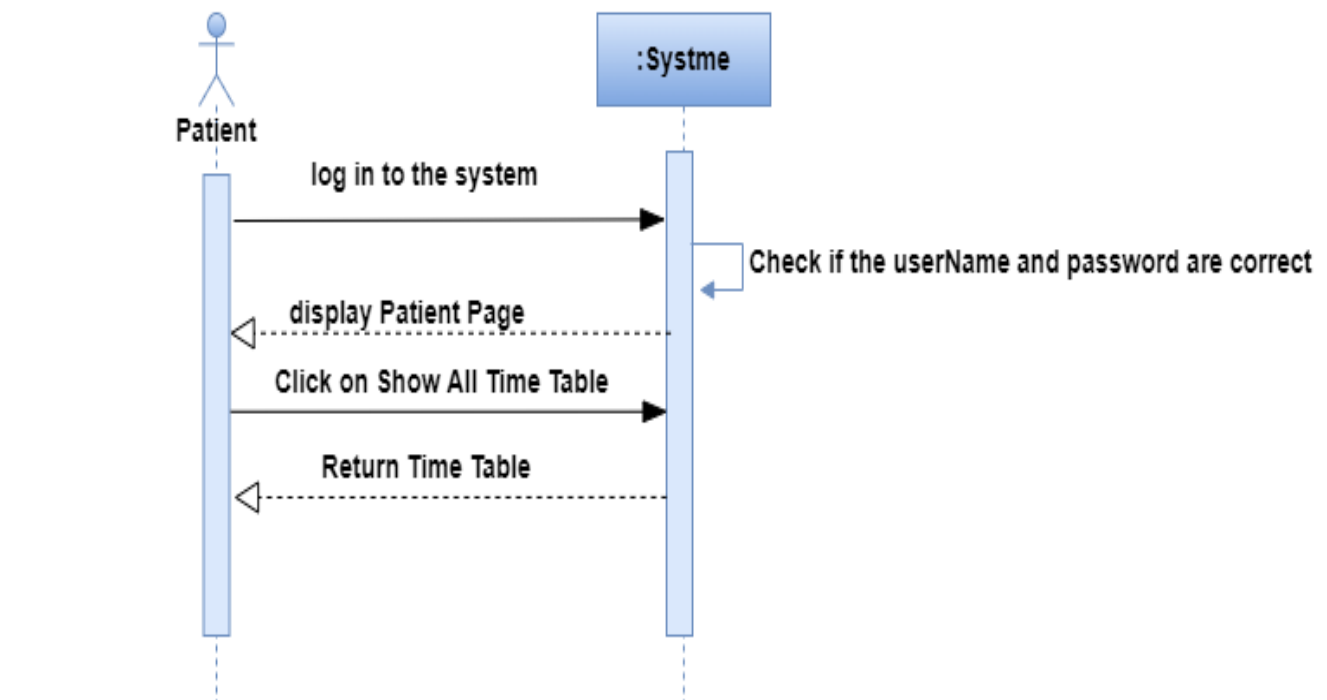
Show Message History of visitor Patient



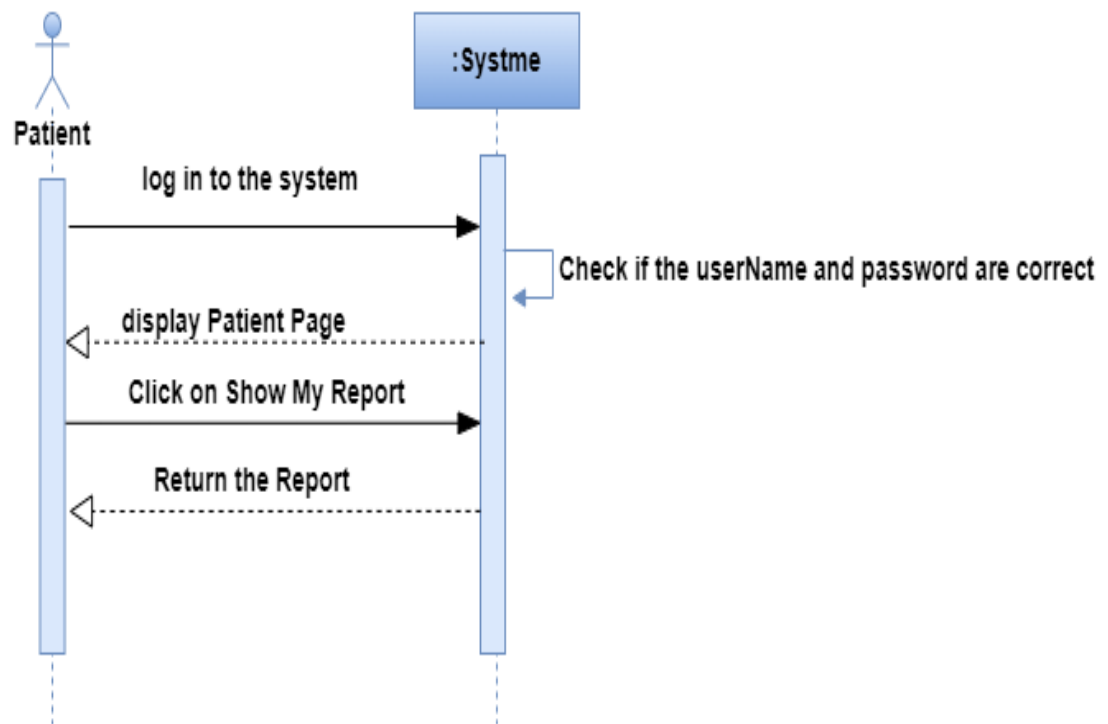
Show Private Message



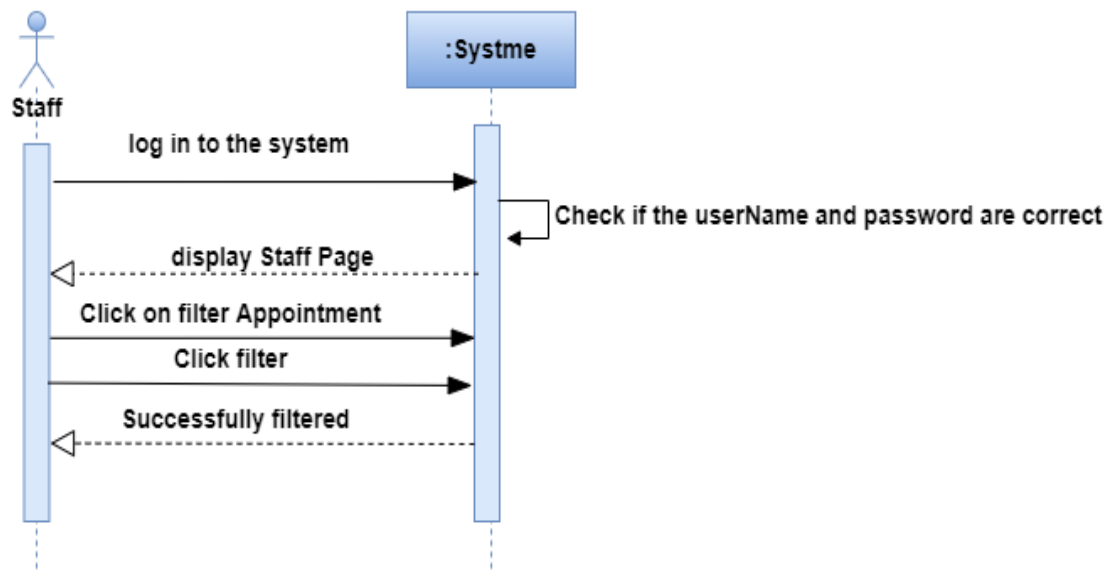
Show All Timetable



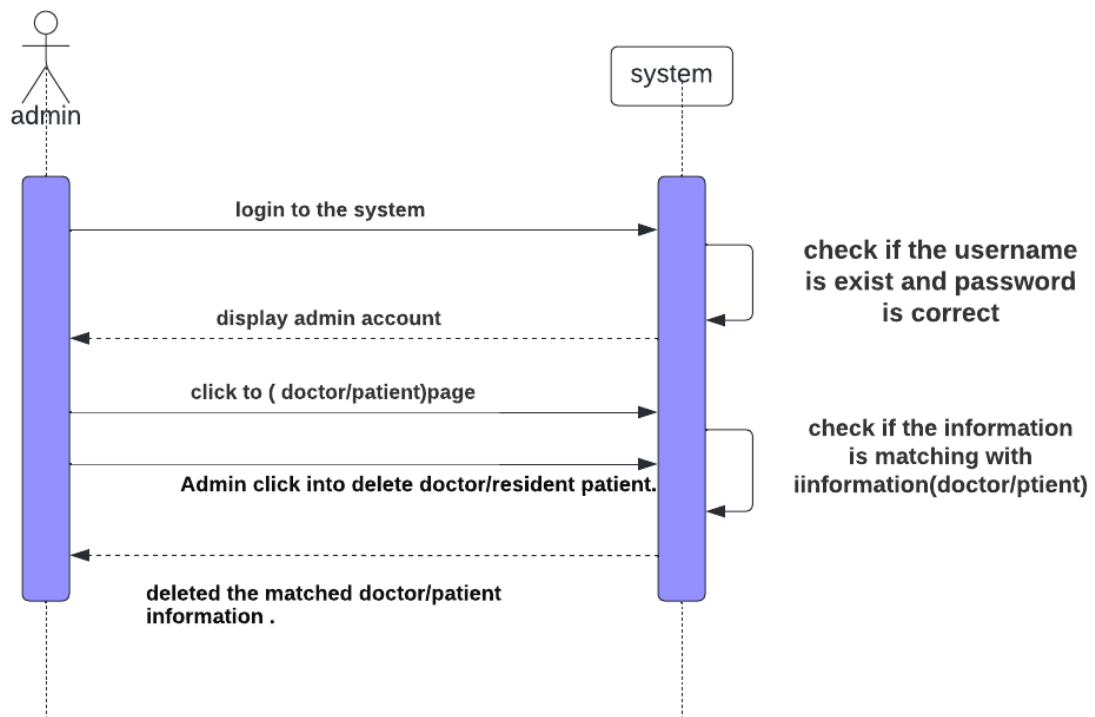
Show My Report



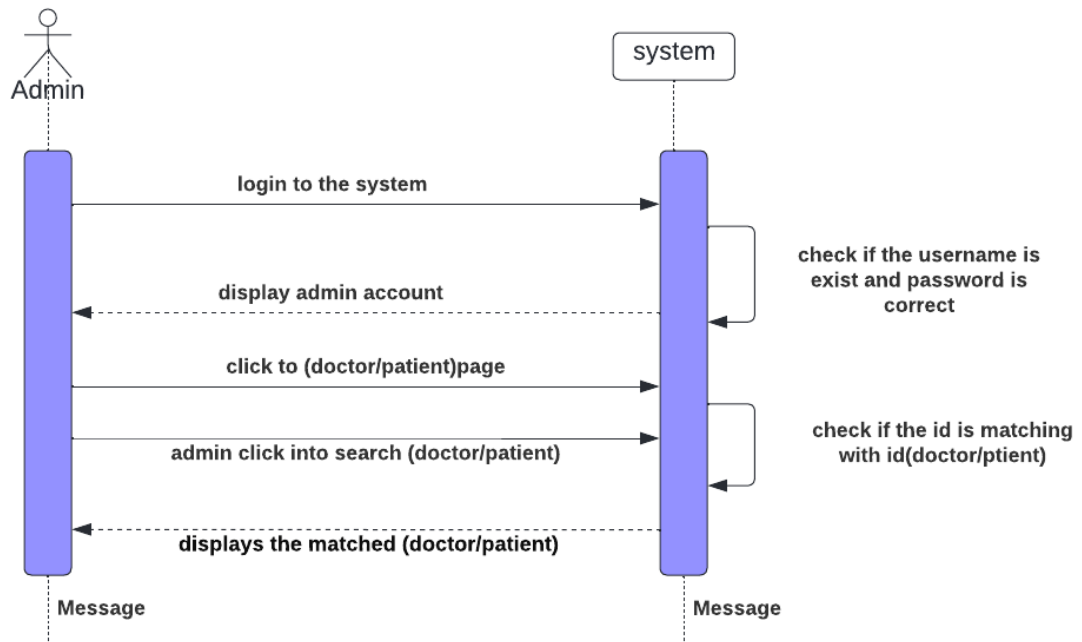
Filter Appointment



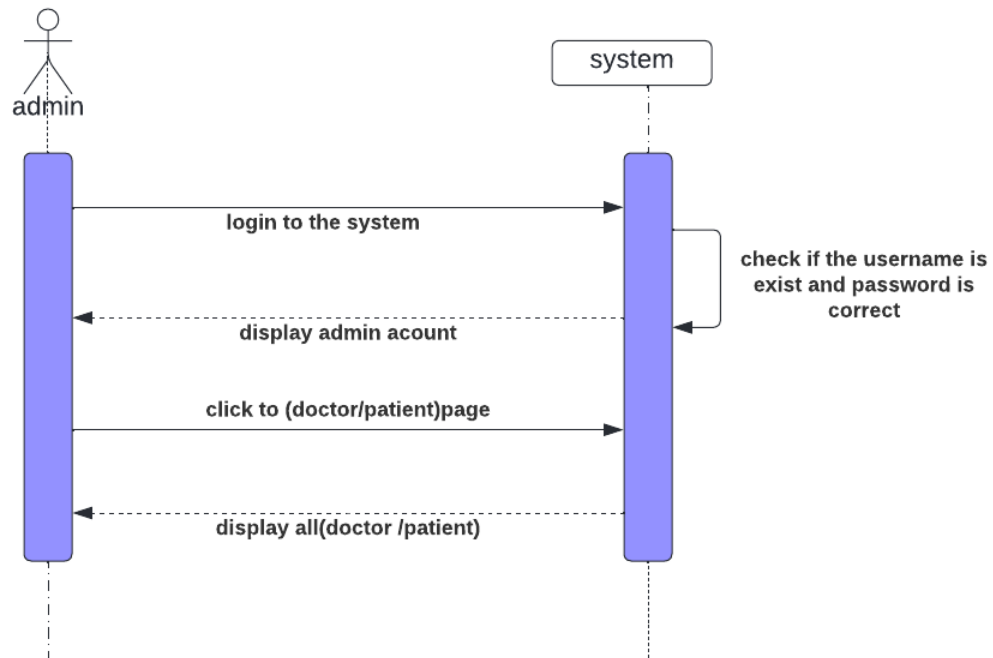
display Doctor/patient



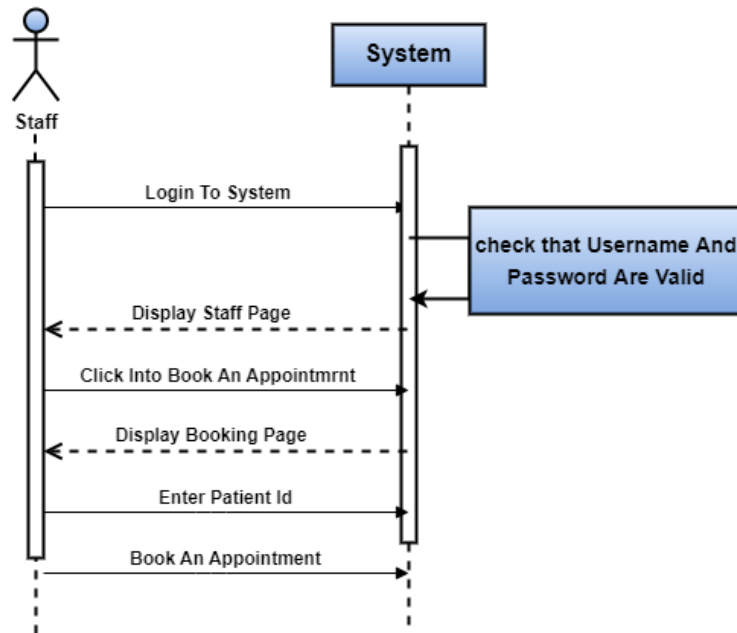
searchUser(Doctor/patient)



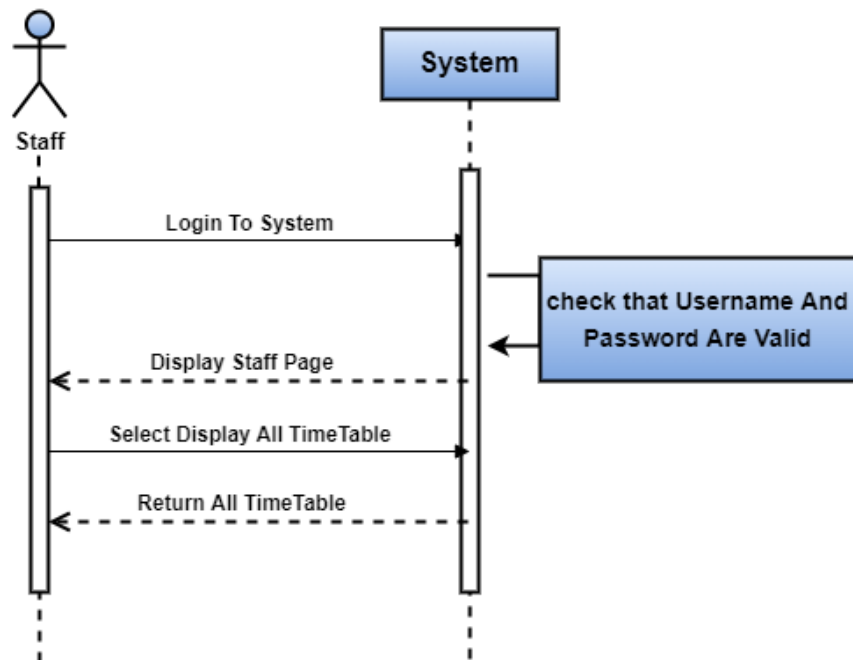
deleteUser(Doctor)



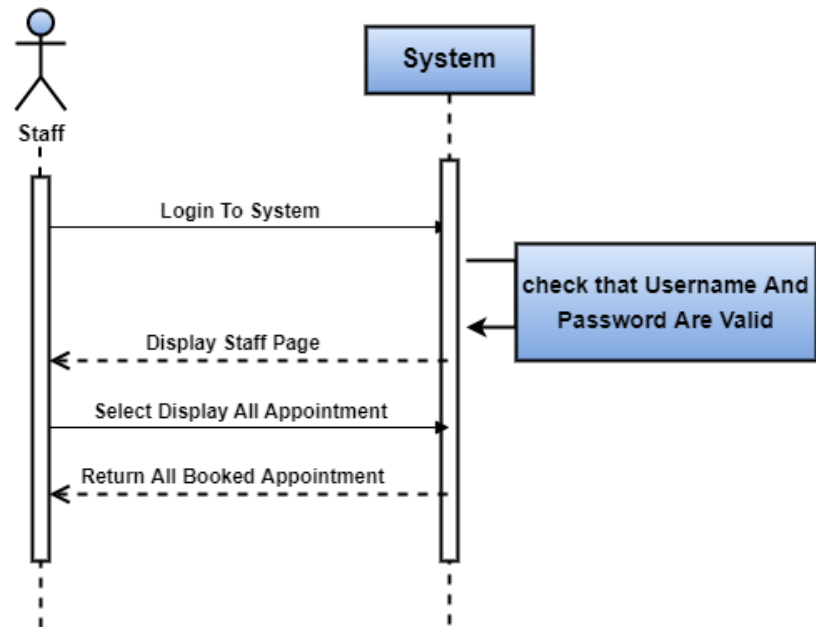
Book An Appointment



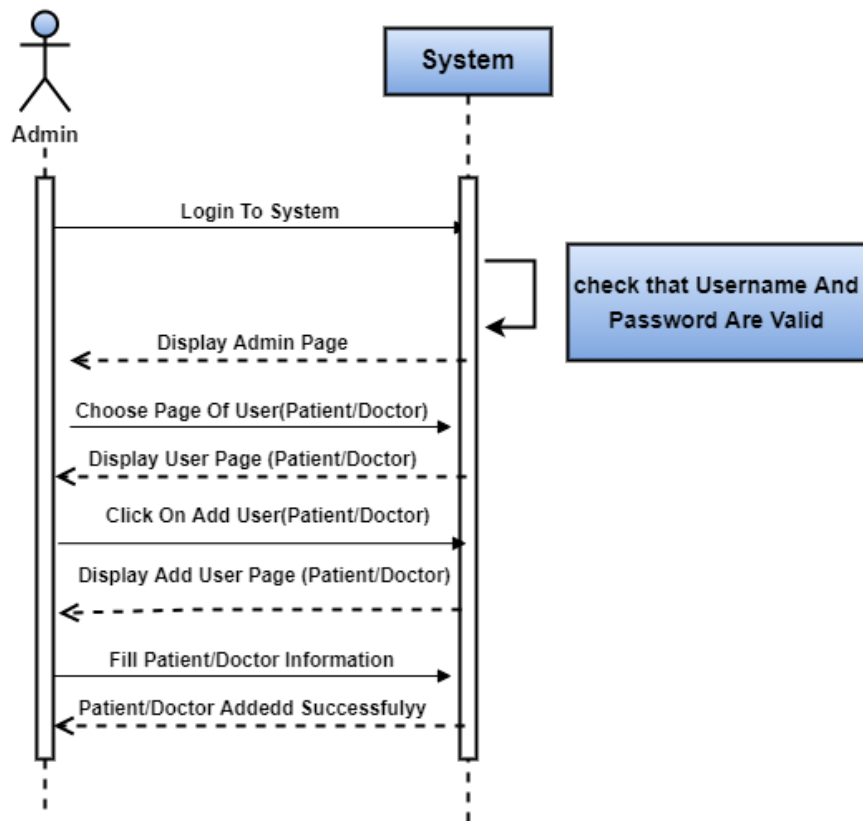
Display All TimeTable



Display All Appointment



Add User



Update User

