MINI PROJECT REPORT

Hospital management System

UE20CS352 – Object Oriented Analysis and Design using Java

Submitted by:

TEAM 13

Gnanesh. A	**PES2UG20CS122**
Murali Sai	**PES2UG20CS083**
Deepak	**PES2UG20CS104**
Saketh	**PES2UG20CS100**

Under the guidance of

Swati Shah
Professor
PES University

January - May 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

Synopsis:

We plan to build an Hospital management system that is a computerized system used by Hospitals to manage the booking and reservation of seats on their flights, as well as other aspects of their operations, such as inventory, pricing, and ticketing.

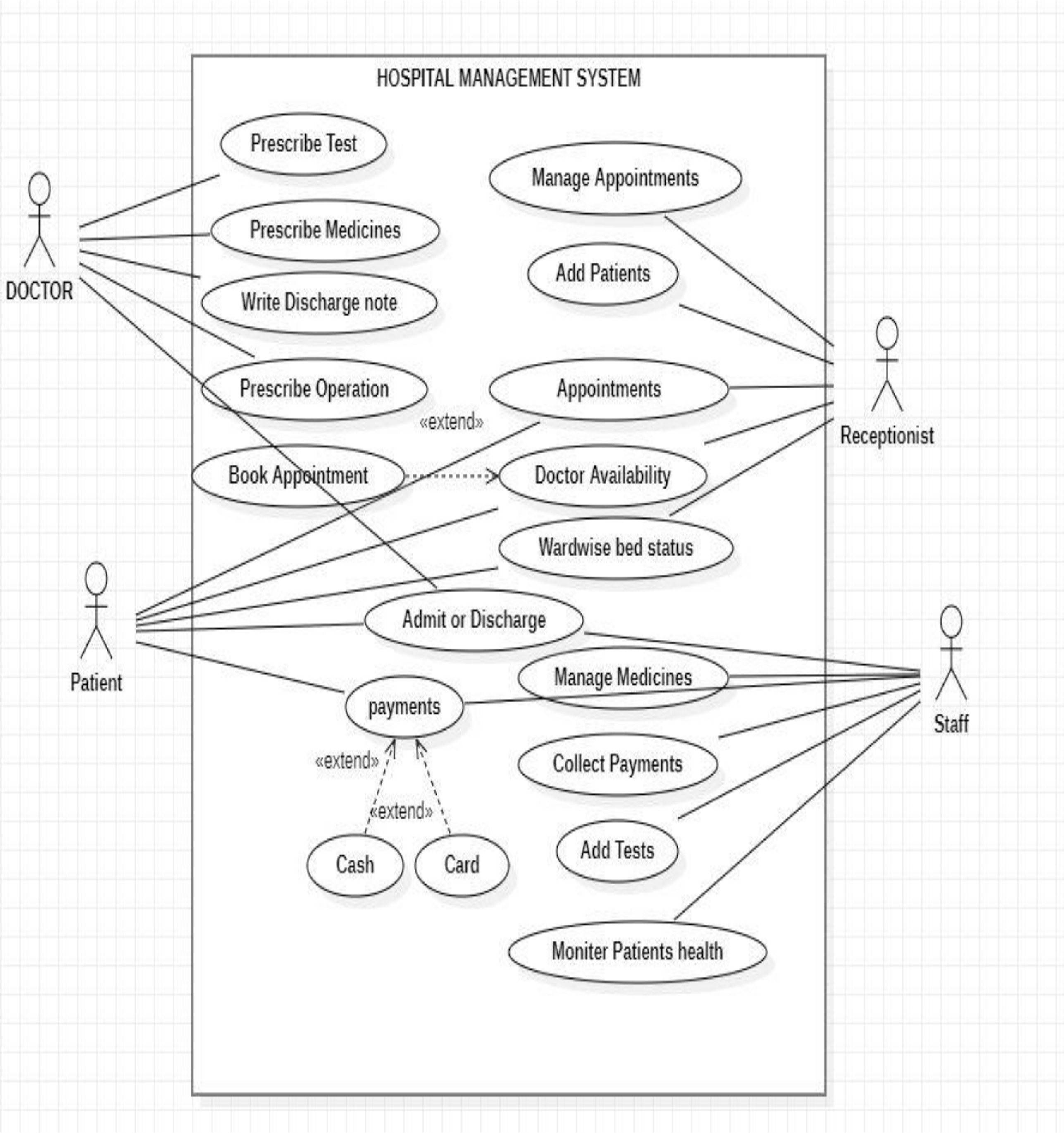
For this project we plan to use **Java and SQL** to build the various frontend and backend functionalities.

The system typically includes multiple modules, such as inventory management, booking management, and fare management, which allow airlines to manage their operations efficiently and provide a seamless booking experience for customers.

The inventory management module tracks the availability of seats on each flight. The booking management module enables customers to search for flights, and book their tickets online, making the booking process faster, more convenient, and more accessible. The fare management module allows airlines to adjust their pricing strategies based on various factors, such as demand, seasonality, and competition, allowing them to maximize their revenue while remaining competitive in the market.

This airline reservation system aims to help in enabling airlines to manage their operations efficiently and provide a seamless booking experience for customers.

Use Case Diagram:



DISCRIPTION

PRESCIPTION

TITLE OR REFERENCE NAME	Prescription
AUTHOR/DATA	Doctor/24-2-2023
MODIFICATION/DATE	24-2-2023
PURPOSE	To advice patients what treatment to be taken by doctor
OVERVIEW	Doctor can write prescription like what treatment to take etc. for the patients. He/she can also advice patients to admit too.
ACTOR	Doctor
PRE-CONDITION	Doctor should prescribe for patients
POST CONDITION	The patients should be satisfied with the prescription
NORMAL FLOW OF EVENTS	Doctor prescribes for respective patients
ALTERNATIVE FLOW OF EVENTS	if doctor prescribed to admit the patient can check for room availability
EXCEPTION FLOW OF EVENTS	If patients are not satisfied, he/she can ask for refund

PRESCRIBE MEDICINES

TITLE OR REFERENCE NAME	Prescribe Medicines
AUTHOR/DATA	Doctor/24-2-2023
MODIFICATION/DATE	24-2-2023
PURPOSE	To advice patients what medicines to be taken by doctor
OVERVIEW	Doctor can write prescription like what medicines to take etc. for the patients.
ACTOR	Doctor
PRE-CONDITION	Doctor should prescribe medicines for patients
POST CONDITION	The patients should be satisfied with the prescription
NORMAL FLOW OF EVENTS	Doctor prescribes for respective patients
EXCEPTION FLOW OF EVENTS	If patients are not satisfied, he/she can ask for refund

PRESCRIBE OPERATION

TITLE OR REFERENCE NAME	Prescribe Operation
AUTHOR/DATA	Doctor/24-2-2023
MODIFICATION/DATE	24-2-2023
PURPOSE	Operation need for patience
OVERVIEW	To advice patients what operation to be taken by doctor
ACTOR	Doctor
PRE-CONDITION	Doctor should prescribe necessary operation for patients
POST CONDITION	The patients should be satisfied with the operation
NORMAL FLOW OF EVENTS	Doctor operates for respective patients
ALTERNATIVE FLOW OF EVENTS	If the operation does not go well residual another operation

WRITE DISCHARGE NOTE

TITLE OR REFERENCE NAME	Write discharge note
AUTHOR/DATA	Doctor/24-2-2023
MODIFICATION/DATE	24-2-2023
PURPOSE	To discharge patients
OVERVIEW	To discharge patients if there health condition is good.
ACTOR	Doctor
PRE-CONDITION	The patient should be admitted
POST CONDITION	The patients can discharge form the hospital.
NORMAL FLOW OF EVENTS	Doctor can prescribe for discharge
EXCEPTION FLOW OF EVENTS	The patients can discharge immediately if he/she wants to shift to another hospital.

PAYMENTS

TITLE OR REFERENCE NAME	Payments
AUTHOR/DATA	Patient/25-02-2023
MODIFICATION/DATE	25-02-2023
PURPOSE	To make payments if the consultation is done.

OVERVIEW	After the consultation is done the patients can make the payments.
CROSS REFERENCE	-
ACTOR	Patient
PRE-CONDITION	The patient should have consulted the doctor.
POST CONDITION	The payment transition should be done.
NORMAL FLOW OF EVENTS	The payment transition should be done successfully
ALTERNATIVE FLOW OF EVENTS	The payments can be made by card or by cash.
EXCEPTION FLOW OF EVENTS	the payment not made successfully
IMPLEMENTATION ISSUES	-

MANAGE APPOINTMENTS

TITLE OR REFERENCE NAME	Manage Appointment
AUTHOR/DATA	Receptionist/ 24-02-2023
MODIFICATION/DATE	24-02-2023
PURPOSE	To manage patient appointment
OVERVIEW	To handle and manage patient appointment with the respective doctor
ACTOR	Receptionist
PRE-CONDITION	Patient should be registered
POST CONDITION	Appointment with the respective doctor is made and managed by receptionist.
NORMAL FLOW OF EVENTS	Handle patient appointment
ALTERNATIVE FLOW OF EVENTS	If the patient does not take appointment he/she can take walk-in appointment
EXCEPTION FLOW OF EVENTS	Appointment is canceled scene the doctor is not available

WARD WISE BED STATUS

TITLE OR REFERENCE NAME	Ward Wise bed status
AUTHOR/DATA	Receptionist/24-02-2023
MODIFICATION/DATE	24-02-2023
PURPOSE	To check the availability of rooms to get admitted
OVERVIEW	The doctor can advise to admit the patient for treatment so, patient can enquire the receptionist about the availability of rooms
ACTOR	Receptionist
PRE-CONDITION	Patient should have the doctor prescription to get admitted

POST CONDITION	Id the rooms are free the patient can make a booking
NORMAL FLOW OF EVENTS	The patient can enquire and book the room
ALTERNATIVE FLOW OF EVENTS	
EXCEPTION FLOW OF EVENTS	The rooms are full hence the patient can not make the booking

APPOINTMENT

TITLE OR REFERENCE NAME	Appointment
AUTHOR/DATA	Receptionist/24-02-2023
MODIFICATION/DATE	24-02-2023
PURPOSE	Patient can make the appointment with respective doctor
OVERVIEW	The patient can call or ask the receptionist to book an appointment with the following doctor
ACTOR	Receptionist
PRE-CONDITION	The patient should be registered with the hospital
POST CONDITION	The patient appointment is successful booked
NORMAL FLOW OF EVENTS	Patient book or enquire about the appointment with the doctor
ALTERNATIVE FLOW OF EVENTS	The patient can also do walk-in appointment with the respective doctor
EXCEPTION FLOW OF EVENTS	The doctor may not be available

DOCTOR AVALABILITY

TITLE OR REFERENCE NAME	Doctor availability
AUTHOR/DATA	Receptionist/24-02-2023
MODIFICATION/DATE	24-02-2023
PURPOSE	Patient can check the availability of doctor
OVERVIEW	Patient book or enquire about the appointment with the doctor
ACTOR	Receptionist
PRE-CONDITION	Checks for available appointments
POST CONDITION	Appointment is booked
NORMAL FLOW OF EVENTS	Patient book or enquire about the appointment with the doctor
ALTERNATIVE FLOW OF EVENTS	The patient can also do walk-in appointment with the respective doctor

MANAGE MEDICINES

TITLE OR REFERENCE NAME	Manage Medicines
AUTHOR/DATA	Staff/24-02-2023
MODIFICATION/DATE	24-02-2023
PURPOSE	manage patients medicines
OVERVIEW	manage patients medicines with doctor prescribe it
CROSS REFERNCE	
ACTOR	Staff
PRE-CONDITION	Patients should consult doctor then buy medicines
POST CONDITION	Correct medicines should be given to the respective patients
NORMAL FLOW OF EVENTS	Patients buy medicine with doctor prescription
ALTERNATIVE FLOW OF EVENTS	Patients buy medicine with no doctor prescription
EXCEPTION FLOW OF EVENTS	Medicine not available
IMPLEMENTATION ISSUES	

ADD TESTS

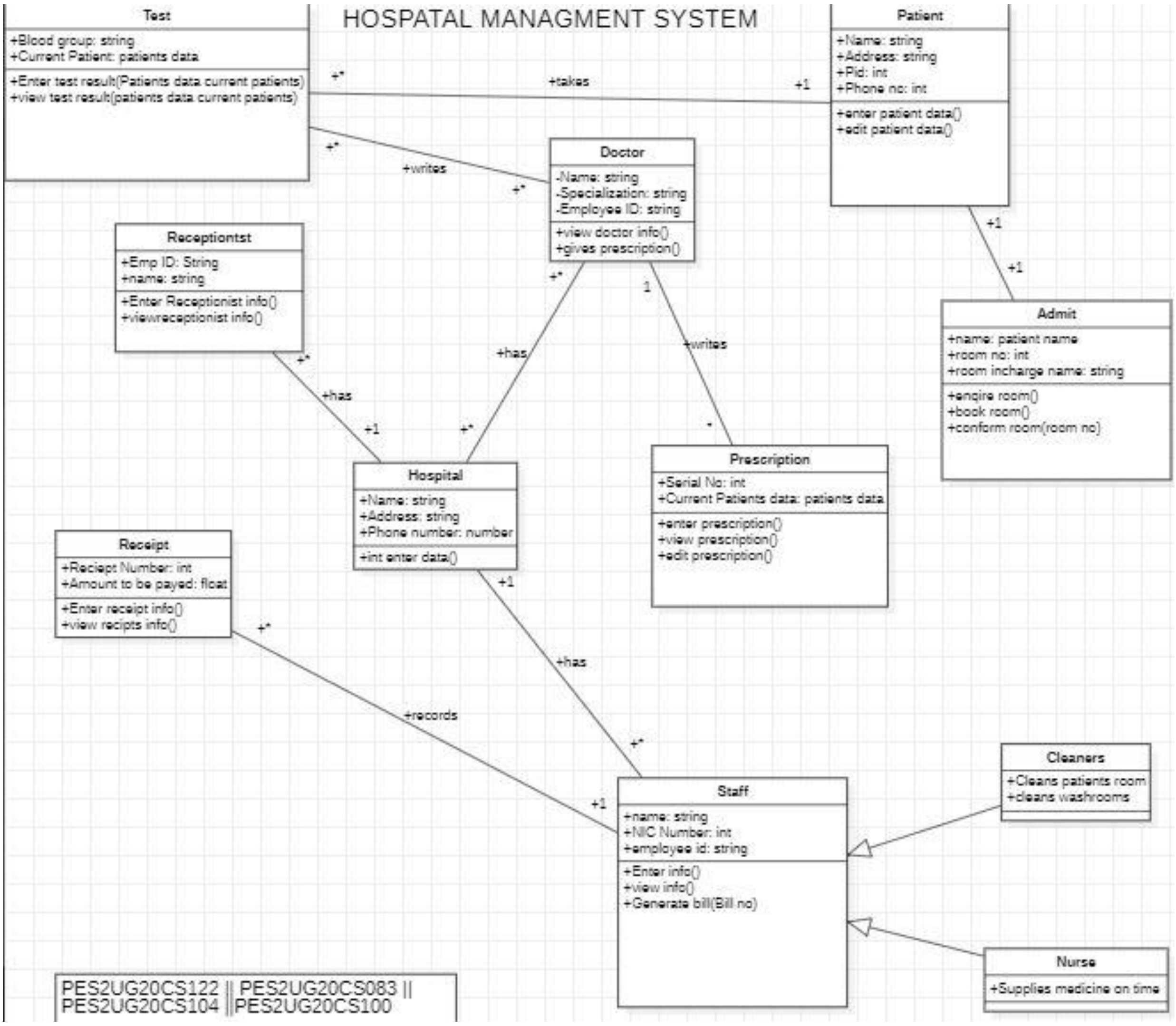
TITLE OR REFERENCE NAME	Add tests
AUTHOR/DATA	Staff/24-02-2023
MODIFICATION/DATE	24-02-2023
PURPOSE	To add the test what the doctor has prescripted
OVERVIEW	Staff add the test and sends it to the laboratory so the patients can make the necessary test
ACTOR	Staff
PRE-CONDITION	The patients should be diagnosed by doctor
POST CONDITION	The necessary test should be done to the patients
NORMAL FLOW OF EVENTS	No changes in the test told by doctor
ALTERNATIVE FLOW OF EVENTS	Change in the test told by doctor
EXCEPTION FLOW OF EVENTS	Test not available in the hospital

MONITOR PATIENTS HEALTH

TITLE OR REFERENCE NAME	Monitor Patients Health
AUTHOR/DATA	Staff/24-02-2023
MODIFICATION/DATE	24-02-2023
PURPOSE	Staff will monitor patients health

OVERVIEW	The nurse will monitor the patients health as told by doctor
ACTOR	Staff
PRE-CONDITION	The patients should be diagnosed by doctor
POST CONDITION	The patients should be good to discharge
NORMAL FLOW OF EVENTS	The patients are monitored
ALTERNATIVE FLOW OF EVENTS	Make necessary changes so the patient's health is better
EXCEPTION FLOW OF EVENTS	Necessary monitoring tools are not available

Class Diagram :



Architecture Patterns:

The Model-View-Controller (MVC) pattern is a popular architectural design pattern used in software development to separate concerns and make software applications more modular, scalable, and maintainable.

In the context of the project, the Login module can be considered as a View which is responsible for presenting the user interface and gathering user input. The Login module interacts with the user to gather the login credentials and then passes them to the next module.

The module can be considered as the Controller which is responsible for managing the flow of data and control between the View and Model components of the system. The module receives the user input from the Login module, validates the credentials and then decides which Model component to call next based on the user's actions. The Model components are responsible for managing data and business logic.

The Model components in the project can be considered as the Model of the MVC pattern, which are responsible for managing data and business logic. The Dashboard module calls the appropriate Model component based on the user's actions and passes the data back to the View for presentation.

By following the MVC pattern, the project ensures a clear separation of concerns and allows for easier maintenance, scalability, and extensibility of the software application. The View, Controller, and Model components can be developed and tested independently, and changes in one component do not affect the others as long as the interfaces between the components remain unchanged.

Design Principles:

The following GRASP design principles were adopted for the project:

- ➔ **Information Expert**: This principle suggests that a responsibility should be assigned to the class that has the most information required to fulfill it. The login module acts as the information expert in this project.

- ➔ **Creator**: GRASP Pattern which helps to decide which class should be responsible for creating a new instance of a class

- ➔ **Controller**: The login module is responsible for managing the overall flow of the system and coordinating the interactions between the other modules.

- **Low coupling**: Each module is designed to have minimal dependencies on the other modules. ‘Adddoctor’ module does not depend on the ‘addpatient’ module and so on.

- **High Cohesion**: Each module was designed to have a clear and well-defined responsibility. Example the addDoctor will add a doctor , editpatient will edit the patient

Design Patterns:

The following design patterns were used to build the project:

- **Observer**: This behavioral pattern was used to implement the communication between different modules, such as the login module and the add patient module. The Observer pattern defines a one-to-many dependency between objects, so that when one object changes state, all its dependents are notified and updated automatically.

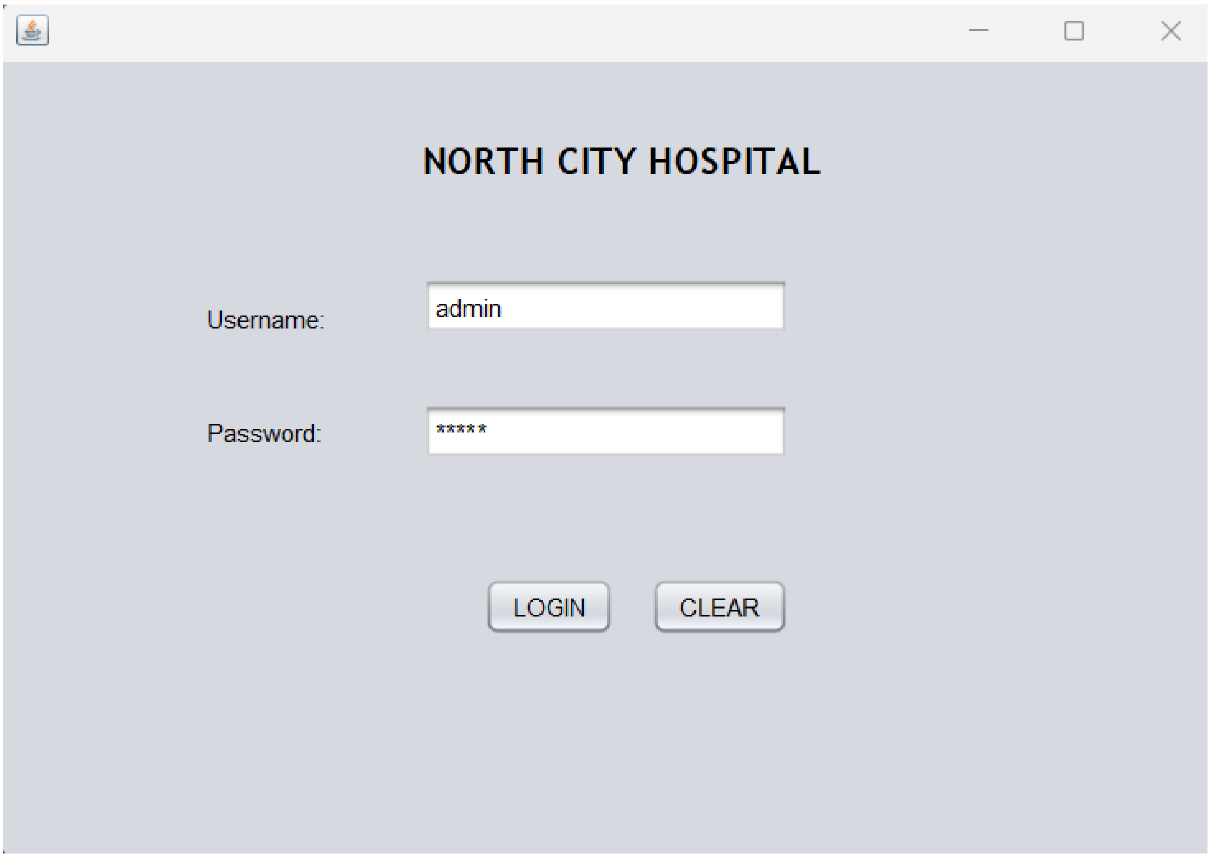
- **Factory Method**: This creational pattern was used to create different types of objects, such as flights, passengers and tickets in a flexible and extensible way. The factory method pattern allows for the creation of objects without specifying the exact class of object that will be created.

- **Singleton Pattern** - This pattern can be used to ensure that only one instance of a critical class, such as a database connection manager or a scheduling service, is created and used throughout the system

- **Facade**: This structural pattern is used to provide a simplified interface to a complex subsystem, such as the and cancellation modules. The Facade pattern provides a unified interface to a set of interfaces in a subsystem, making it easier to use and reducing coupling between the subsystem and its clients.

Screenshots with input values populated and output shown:

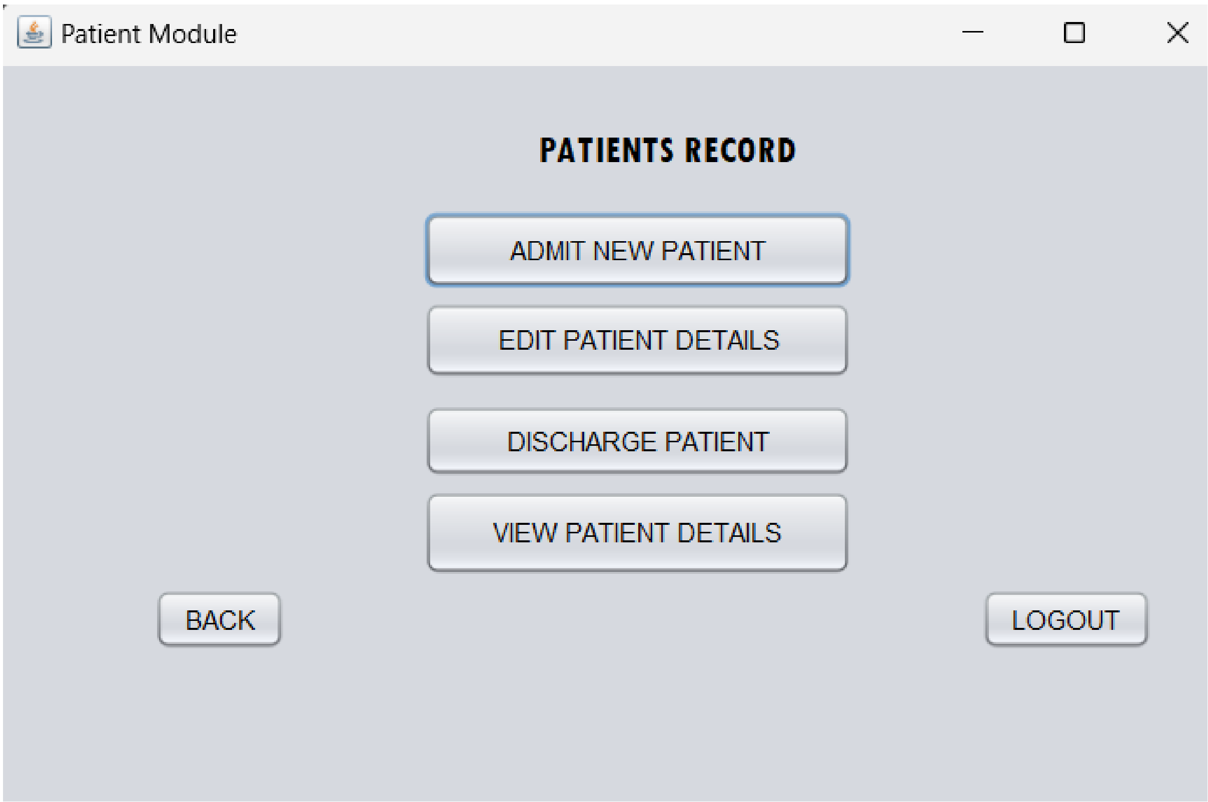
Login page:

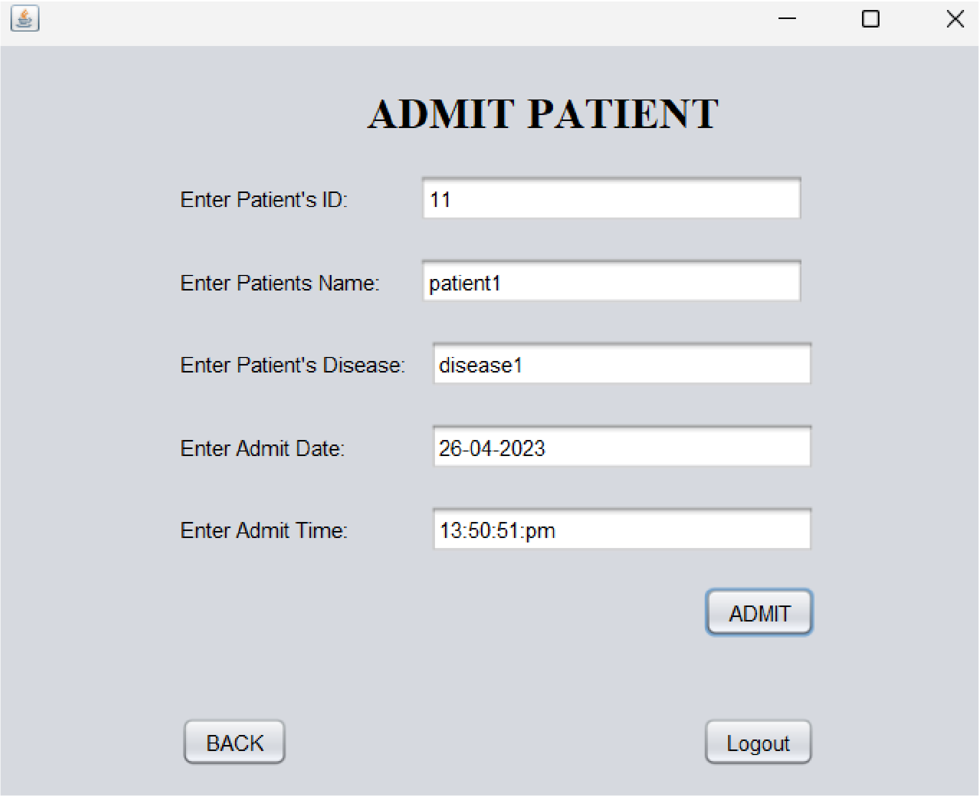


NORTH CITY HOSPITAL

Username: admin

Password: *****





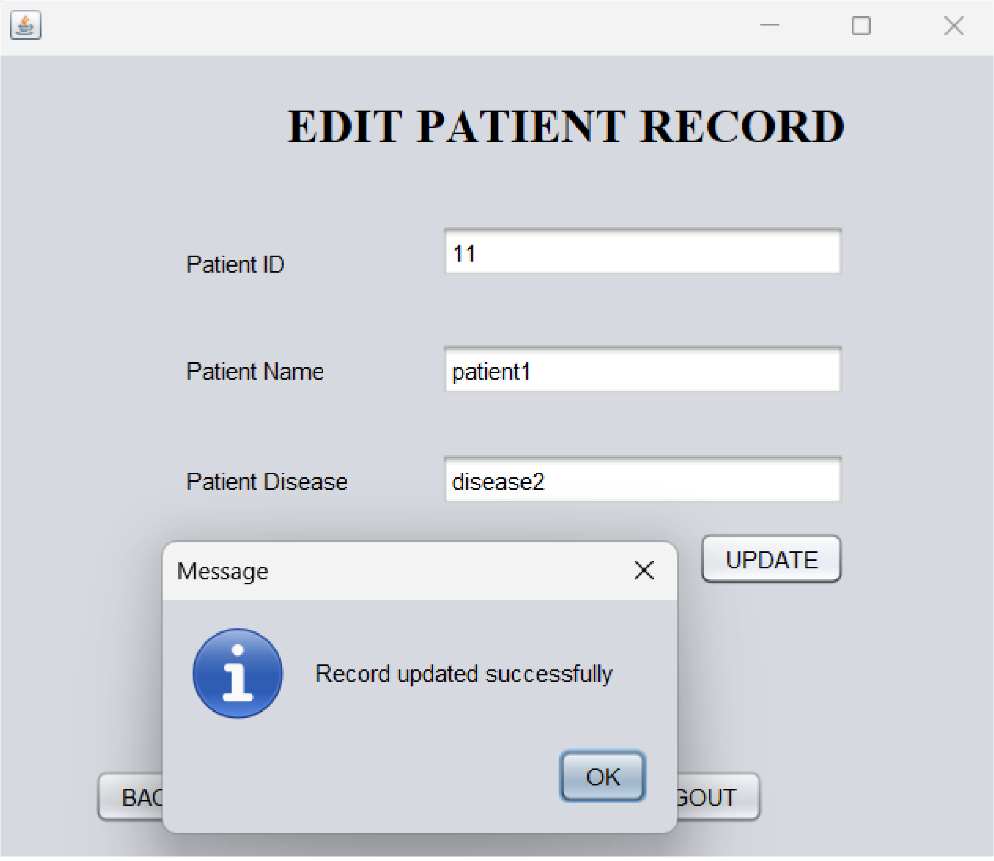
Enter Patient's ID: 11

Enter Patients Name: patient1

Enter Patient's Disease: disease1

Enter Admit Date: 26-04-2023

Enter Admit Time: 13:50:51:pm



EDIT PATIENT RECORD

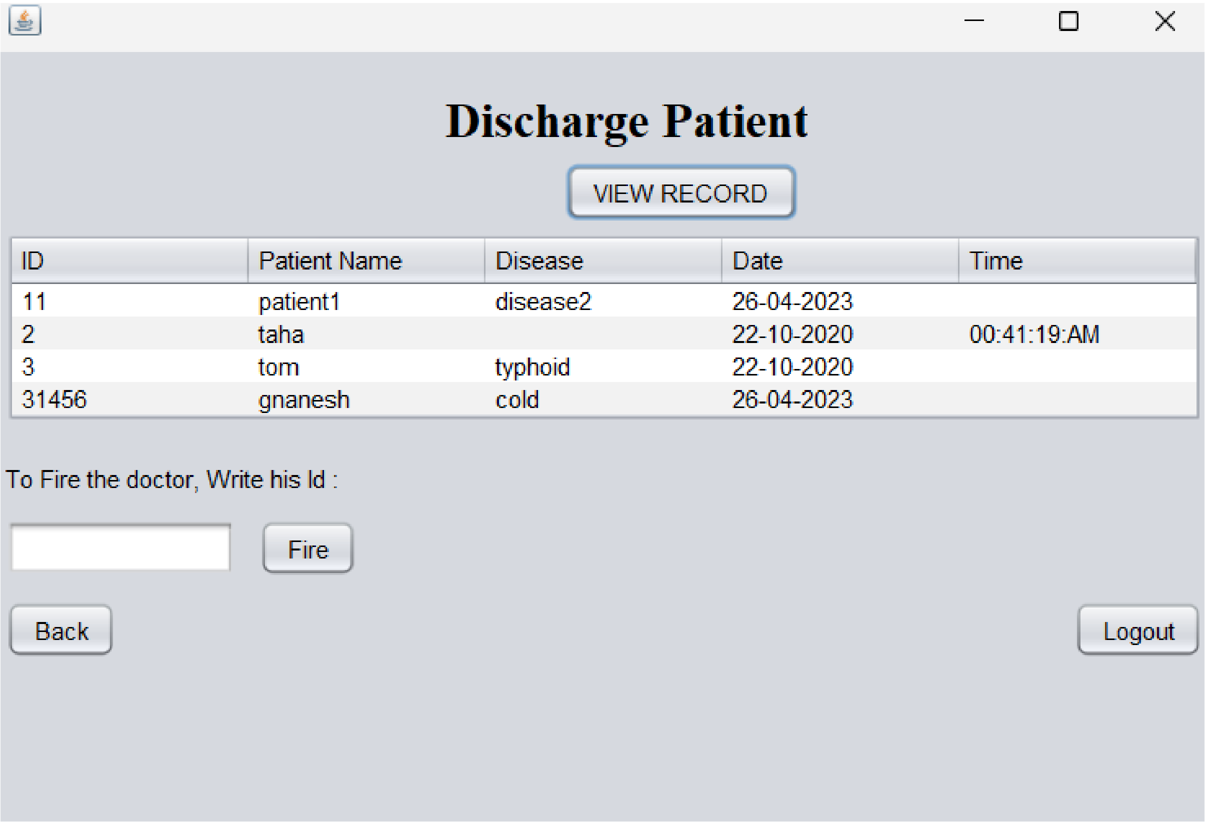
Patient ID: 11

Patient Name: patient1

Patient Disease: disease2

Message

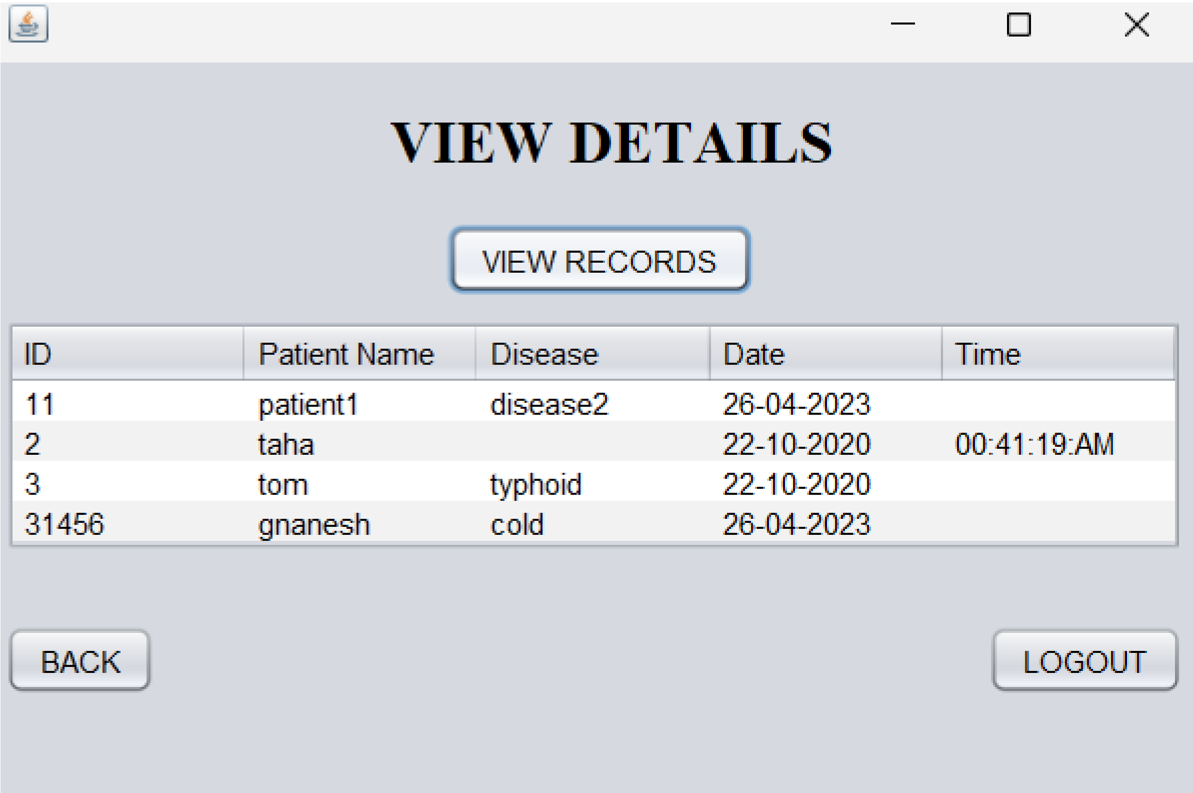
Record updated successfully



Discharge Patient

ID	Patient Name	Disease	Date	Time
11	patient1	disease2	26-04-2023	
2	taha		22-10-2020	00:41:19:AM
3	tom	typhoid	22-10-2020	
31456	gnanesh	cold	26-04-2023	

To Fire the doctor, Write his Id :



VIEW DETAILS

ID	Patient Name	Disease	Date	Time
11	patient1	disease2	26-04-2023	
2	taha		22-10-2020	00:41:19:AM
3	tom	typhoid	22-10-2020	
31456	gnanesh	cold	26-04-2023	