PES University, Bengaluru

(Established under Karnataka Act 16 of 2013)

Department of Computer Science & Engineering

Session: Jan - May 2022

UE19CS353 – Object Oriented Analysis and Design with Java

Theory ISA (Mini Project)

Report on

Hospital Management System

By:

Kartika Nair – PES1UG19CS213

Krithika Ragothaman – PES1UG19CS231

Maitreyi P – PES1UG19CS254

6th Semester Section D

1. Project Description

a. Link to Github repository

https://github.com/kartika-nair/OOAD-MiniProject

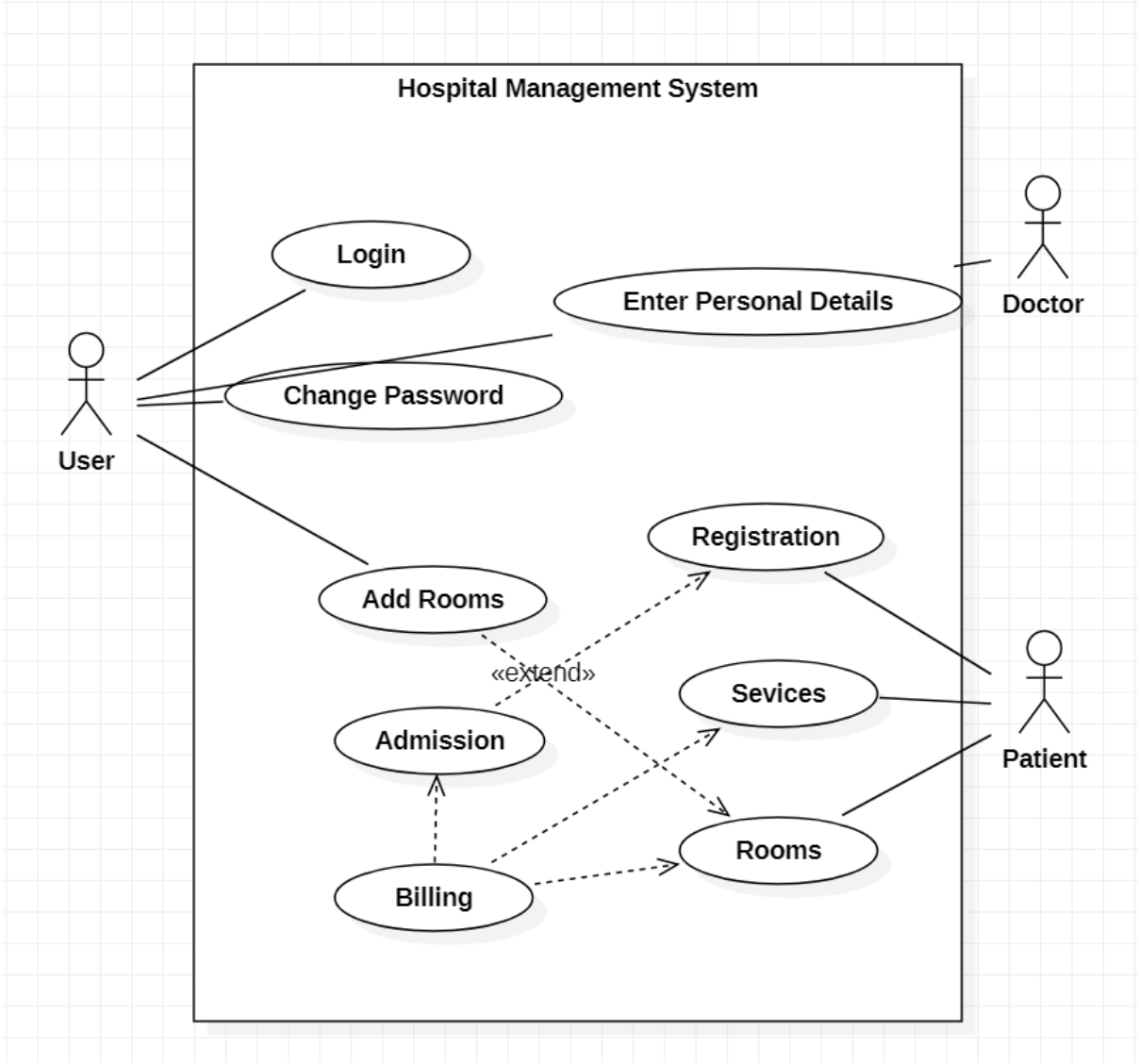
b. Synopsis

In order to simplify the management of processes within a hospital, a Hospital Management System has been created. The Patient begins by going through the Registration process, guided by the Receptionist. In some cases, the Patient may arrive via an Ambulance, driven by Ambulance Drivers employed by the Hospital. Following this, a Doctor is assigned to the Patient, and the Doctor conducts a Consultation/Diagnosis with the help of one or more Nurses. In order to assign a Doctor to the Patient, the Administrative/General Staff, who are responsible for Management of Doctors, assist the Receptionist. Based on the results of the Consultation/Diagnosis, which may involve Scans/Tests, and the Doctor going through the Patient's Medical History, a decision is made on the Patient's next steps. This may involve simply visiting the Pharmacist and Purchasing Medicine, or may require Admission to a Hospital Room. The Room is allotted by checking the Bed Status to see whether it is free or not, and whether the Cleaning Staff have ensured that the Room and Bed are clean. In some cases, the Diagnosis may also lead to Surgery. Payment from the Patient to the Hospital is managed by the Administrative/General Staff.

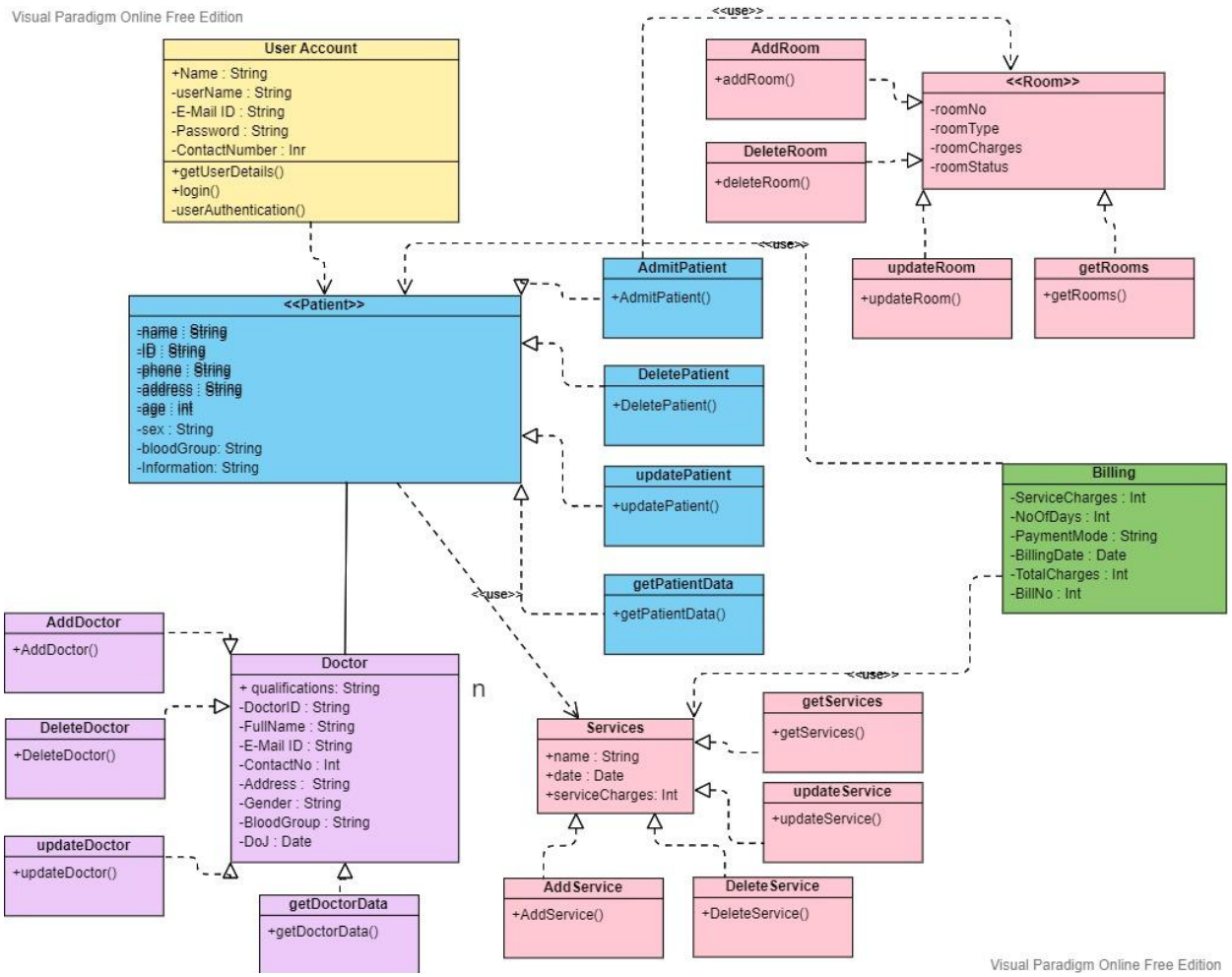
2. Analysis and Design Models

a. Static Models

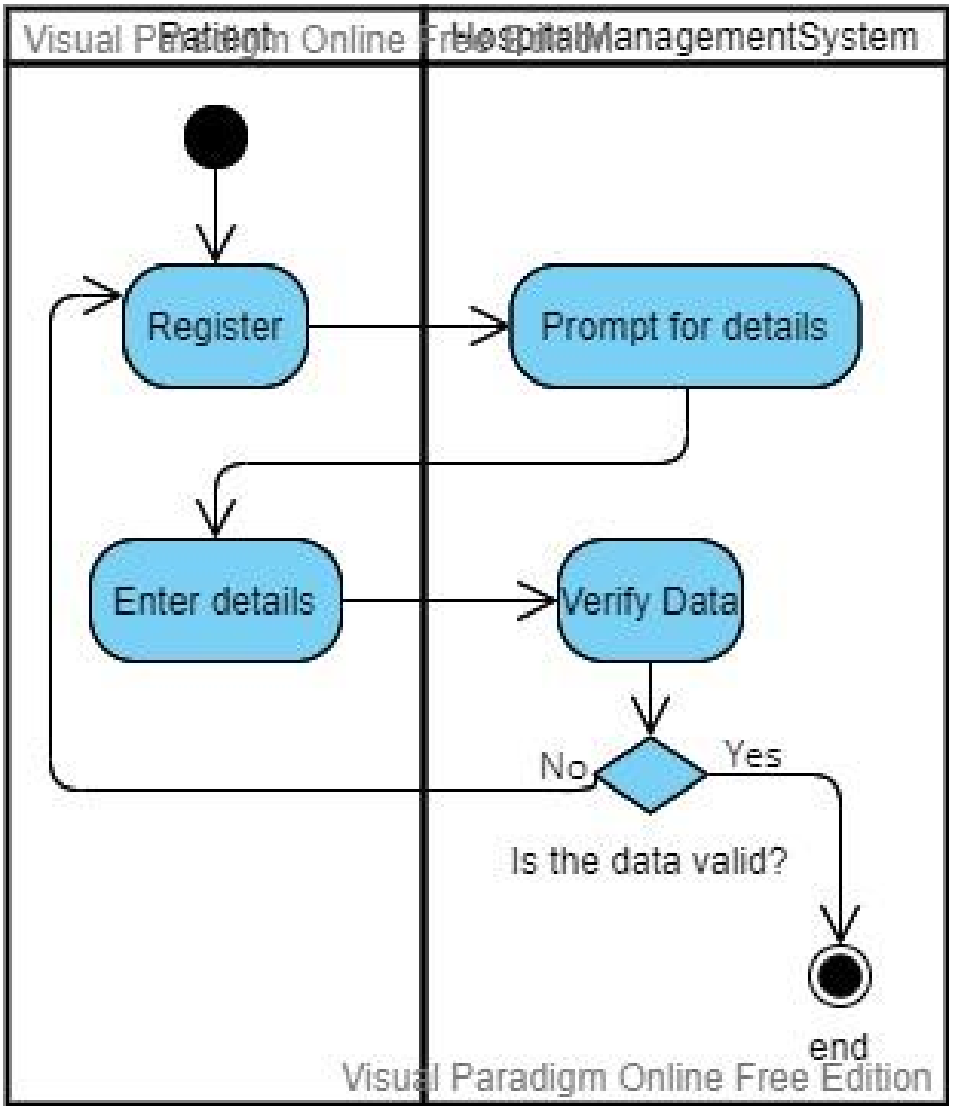
Use Case Diagram

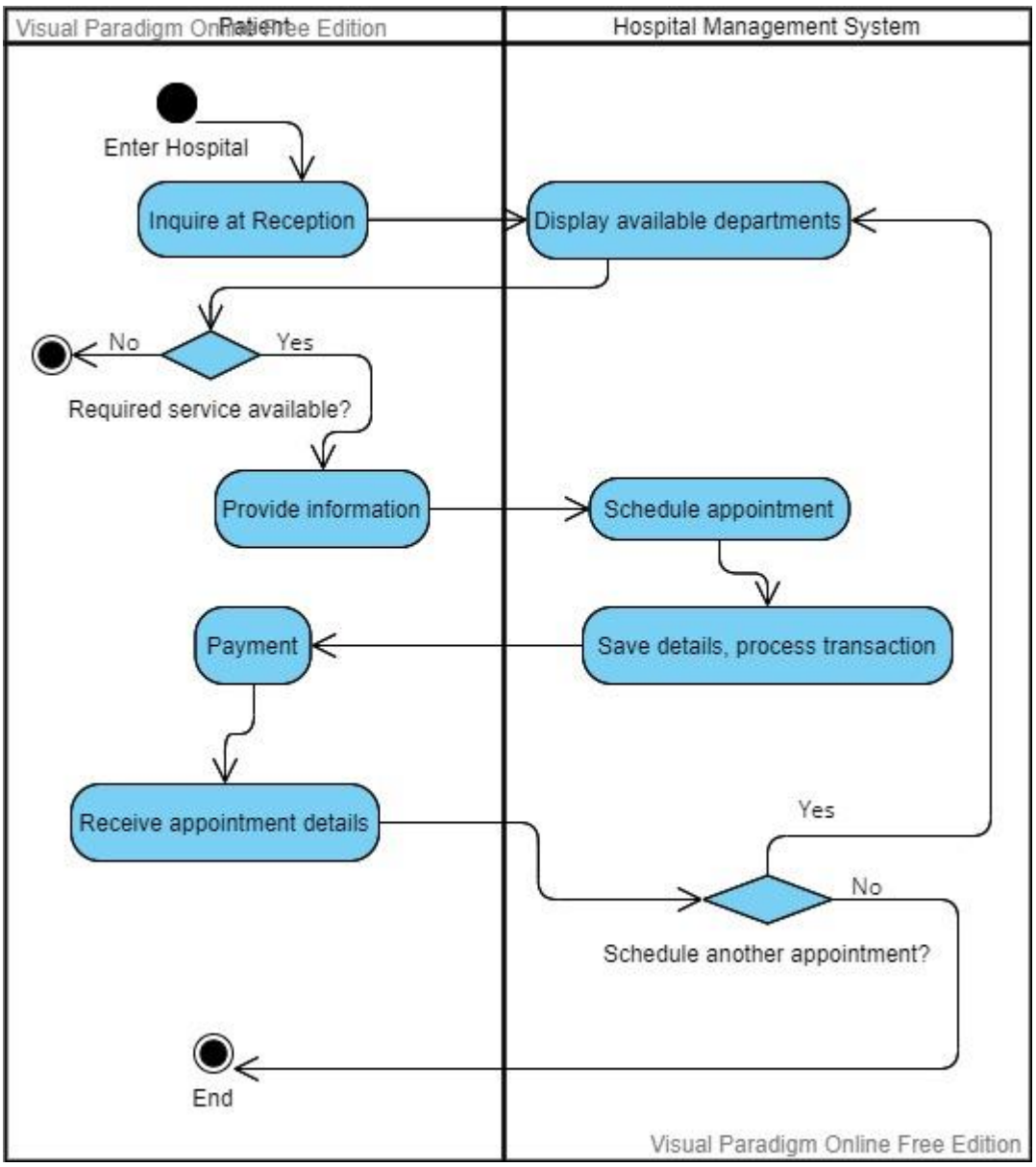


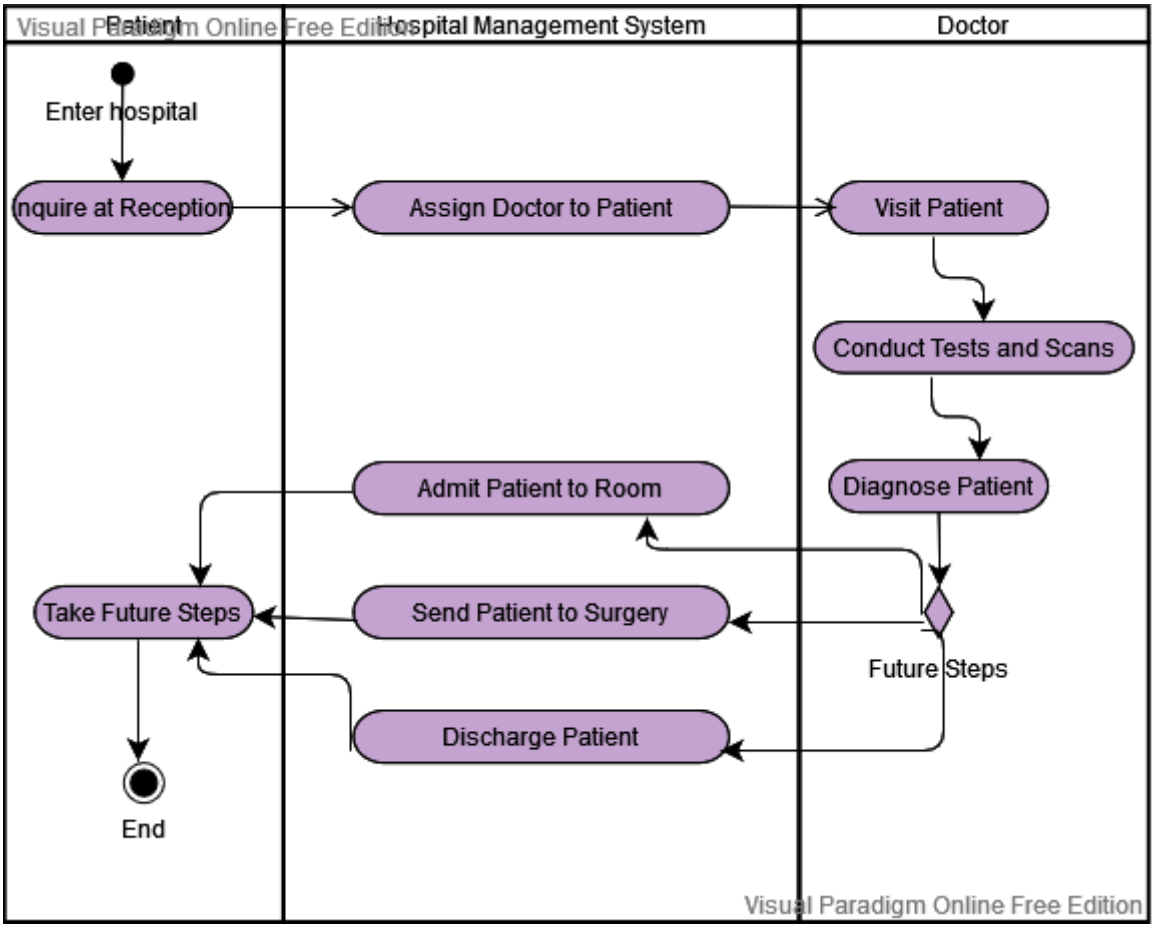
Final Class Model



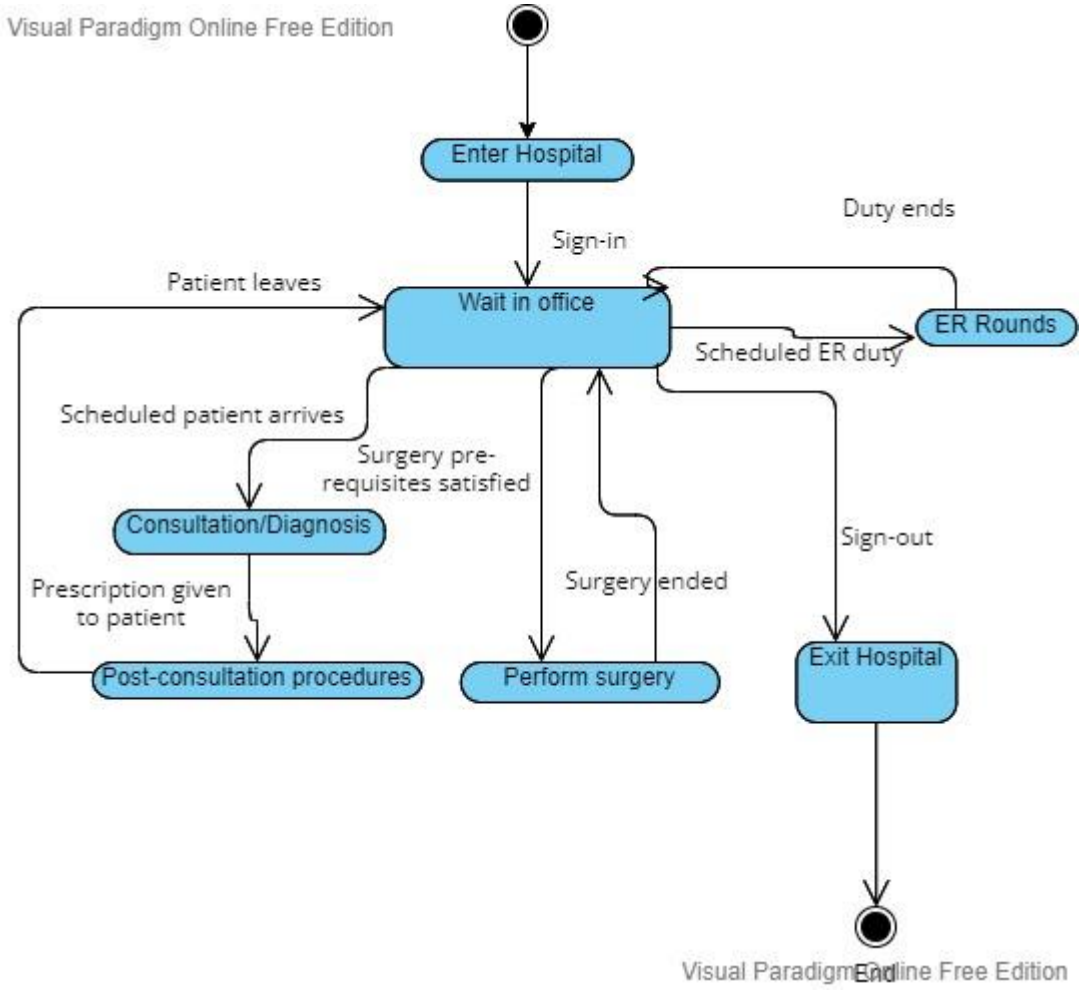
3. Activity Diagrams

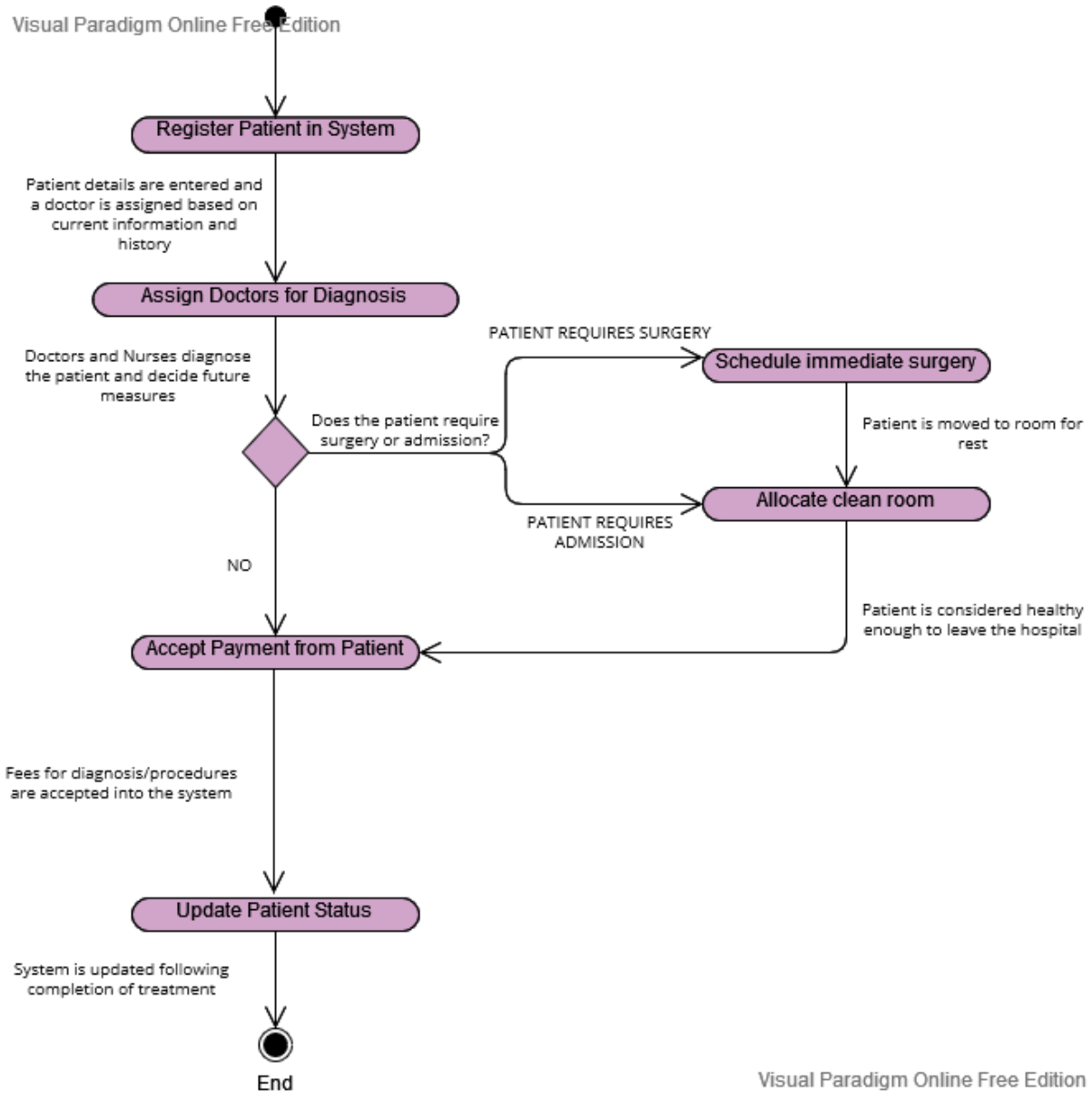


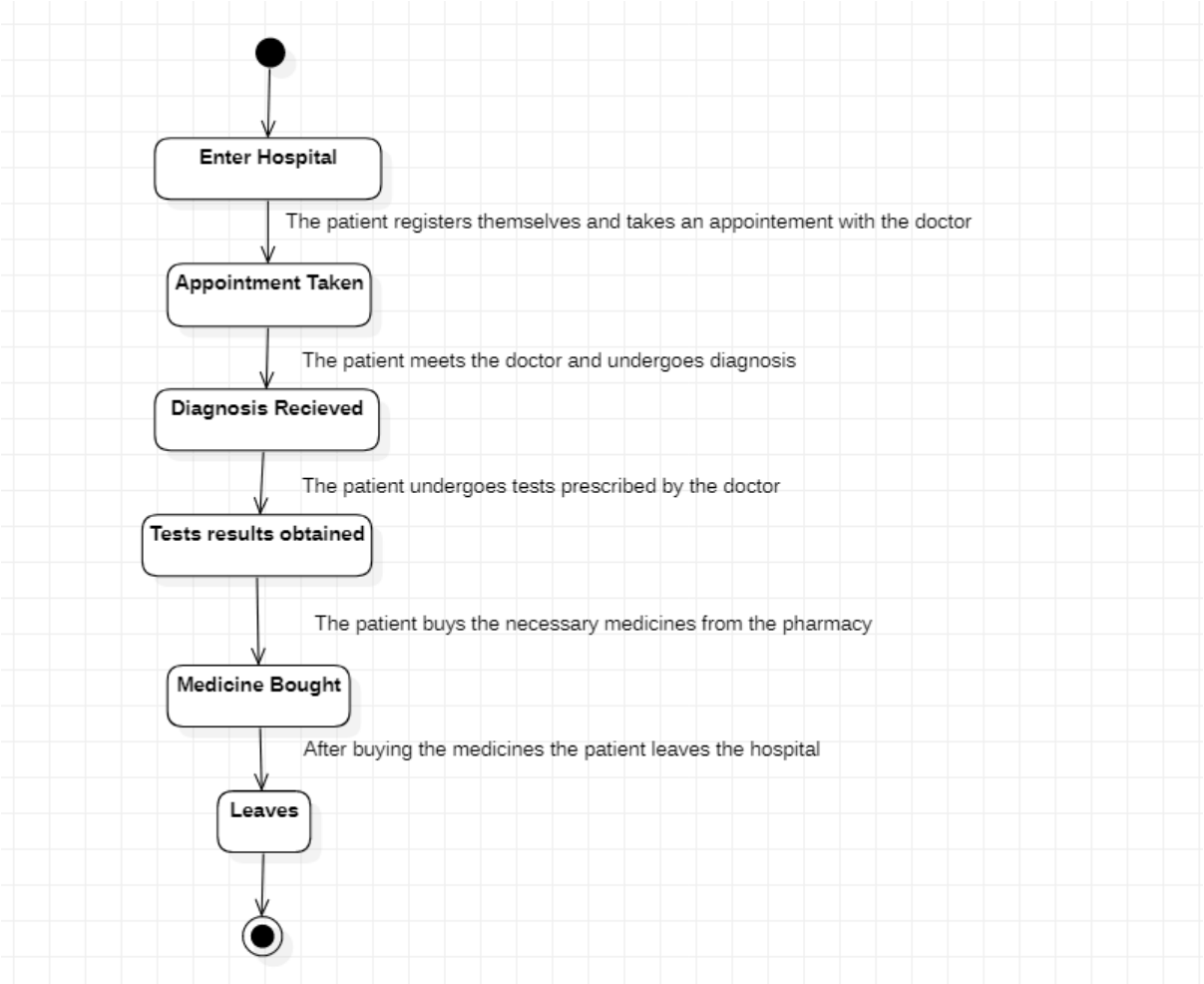

Patient
Hospital Management System
Enter Hospital
Inquire at Reception
Display available departments
No
Yes
Required service available?
Provide information
Schedule appointment
Payment
Save details, process transaction
Receive appointment details
Yes
No
Schedule another appointment?
End
Visual Paradigm Online Free Edition


Visual Paradigm Online Free Edition
Patient
Hospital Management System
Doctor
Enter hospital
Inquire at Reception
Assign Doctor to Patient
Visit Patient
Conduct Tests and Scans
Diagnose Patient
Admit Patient to Room
Take Future Steps
Send Patient to Surgery
Future Steps
Discharge Patient
End
Visual Paradigm Online Free Edition

4. State Diagrams




Visual Paradigm Online Free Edition
Register Patient in System
Patient details are entered and a doctor is assigned based on current information and history
Assign Doctors for Diagnosis
Doctors and Nurses diagnose the patient and decide future measures
Does the patient require surgery or admission?
PATIENT REQUIRES SURGERY
Schedule immediate surgery
Patient is moved to room for rest
Allocate clean room
PATIENT REQUIRES ADMISSION
NO
Patient is considered healthy enough to leave the hospital
Accept Payment from Patient
Fees for diagnosis/procedures are accepted into the system
Update Patient Status
System is updated following completion of treatment
End
Visual Paradigm Online Free Edition



5. Tools and Frameworks Used

a. Frameworks:
- MVC (Model View Controller)
  - Java Swing

b. Tools:
- NetBeans (Java IDE)
- MySQL

6. Design Principles and Design Patterns Applied

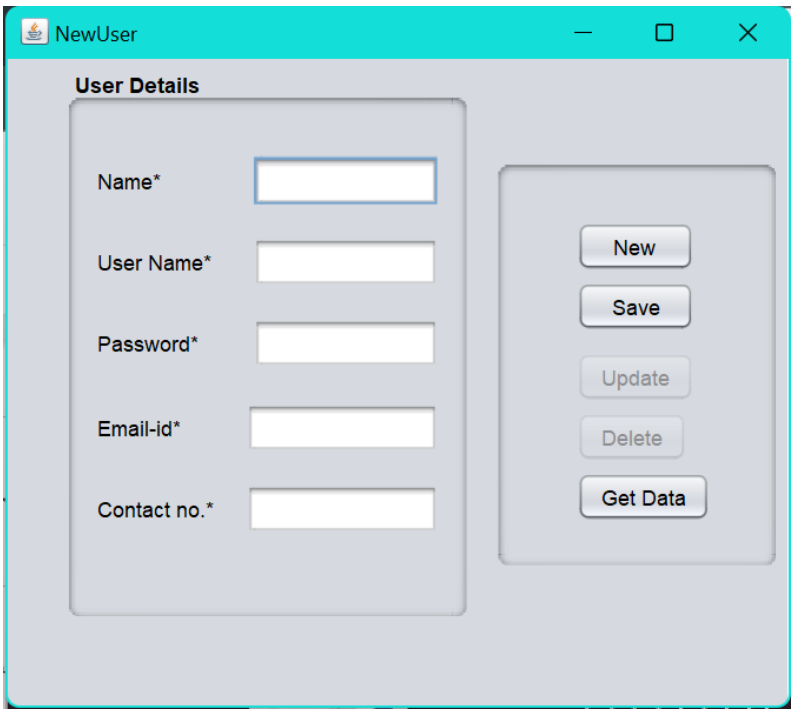
The design principles applied to this project include:

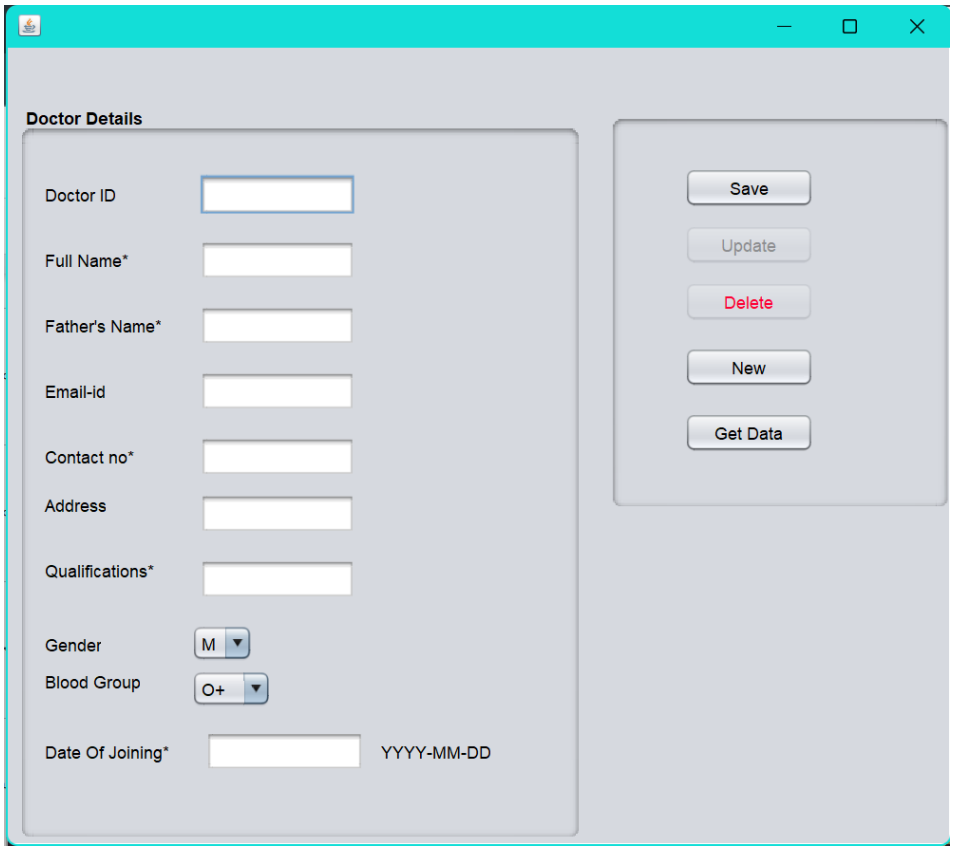
- Single Responsibility Principle (SRP) - every class performs a different functionality
    - Different files for different features
    - Some components of features also split into different files to ensure SRP
- Integration Segregation Principle (ISP)
    - Only 4 interfaces - Users, Doctors, Patients, Rooms
    - Client interacts only with what is necessary
- Dependency Inversion Principle (DIP)
    - Classes were added for increment, count, etc.
    - Above details depend on abstractions

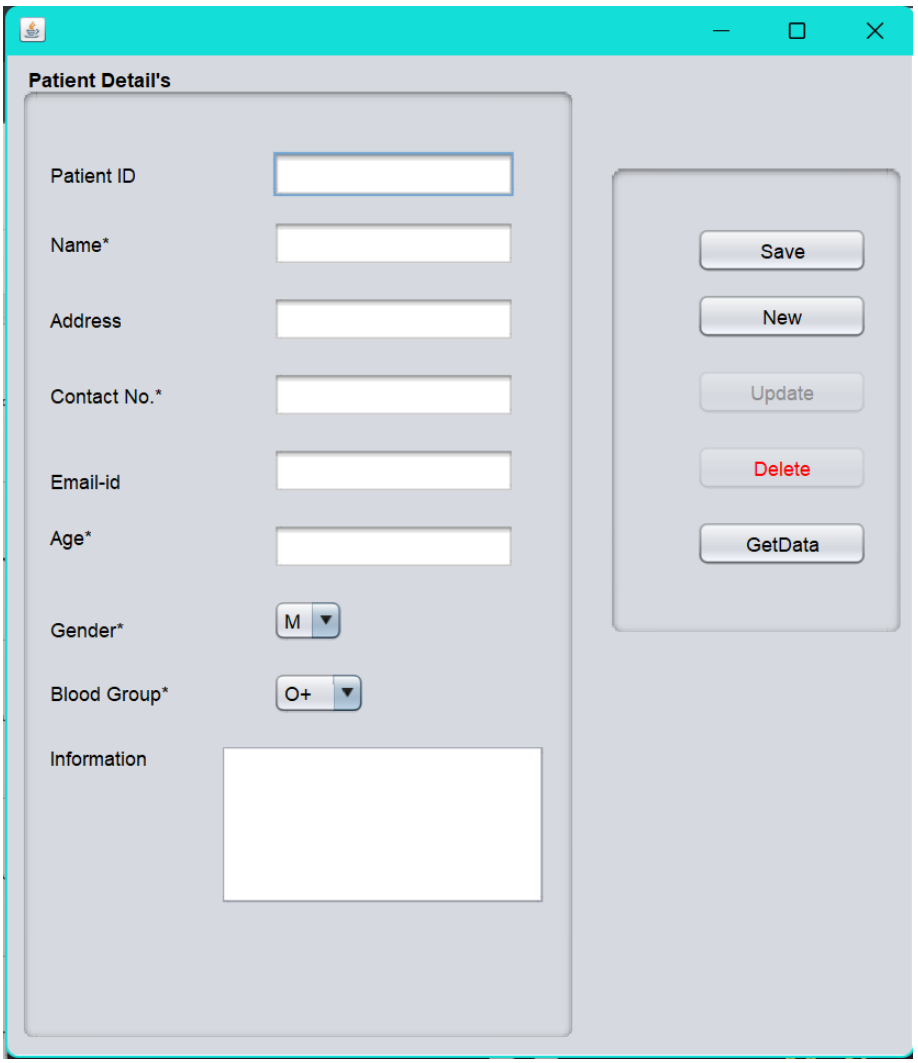
The design patterns used in the project include:

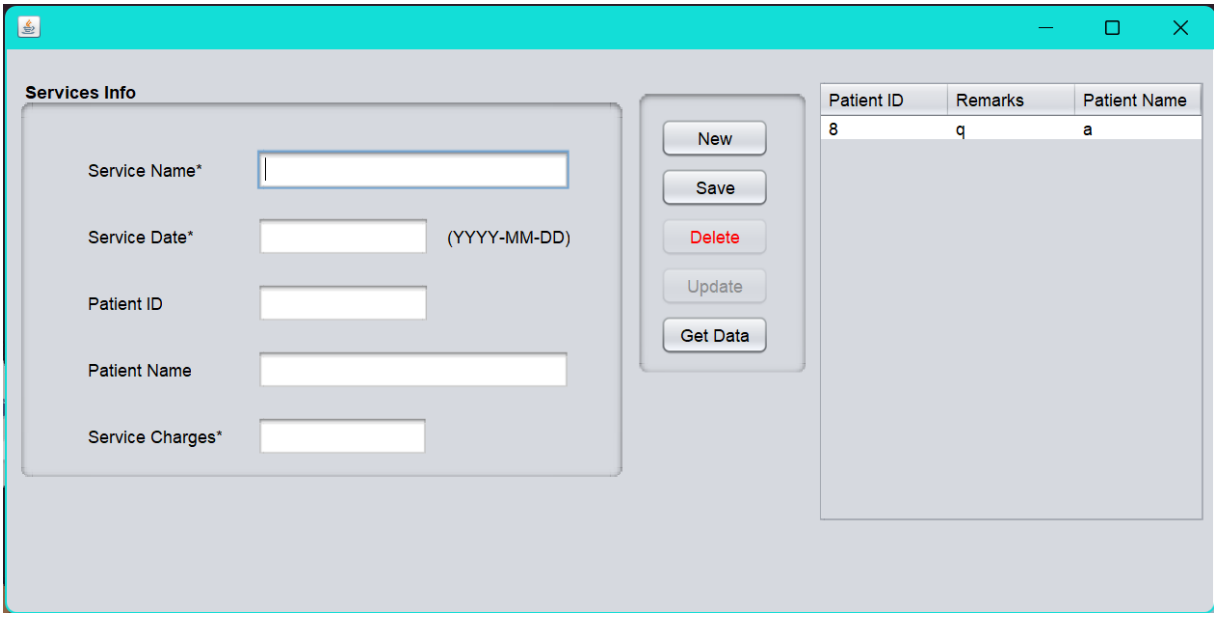
- Creational Design Pattern - Builder Pattern
    - Separates the construction of a complex object from its representation
    - Database contains details
    - Querying and calculations are done on database to create complex objects
    - Final complex object is returned to user
- Structural Design Patterns
    - Bridge - separated object interface and implementation
    - Facade - single class to represent entire subsystem (ex: Users, Doctors, Patients, Rooms)
- Behavioural Design Patterns
    - Command - command requests encapsulated as objects (ex: SQL queries)
    - Interpreter - language elements are included
    - Mediator - communication between classes is simple
    - State - object behaviour altered when state is changed (ex: entry of details)
    - Template - exact steps of algorithm deferred to subclass (ex: calculations)

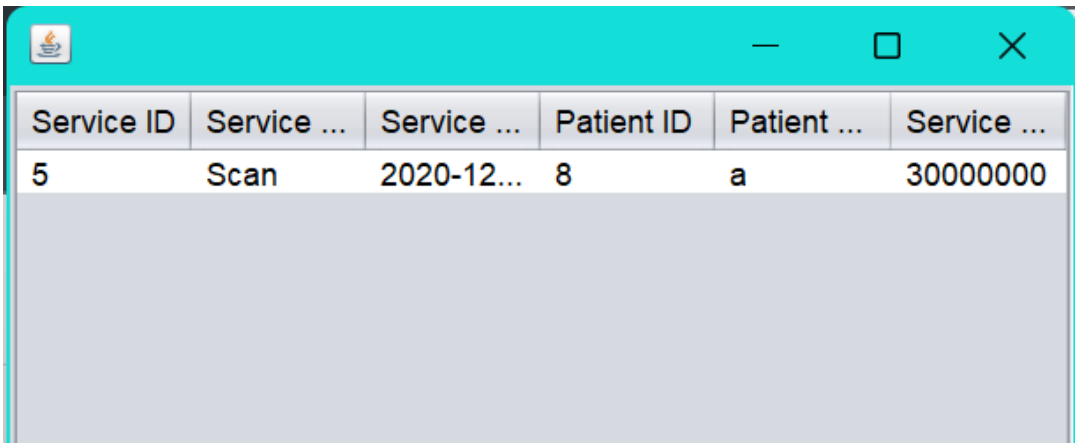
7. Application Screenshots (3-4 important pages)


NewUser
User Details
Name*
User Name*
Password*
Email-id*
Contact no.*
New
Save
Update
Delete
Get Data

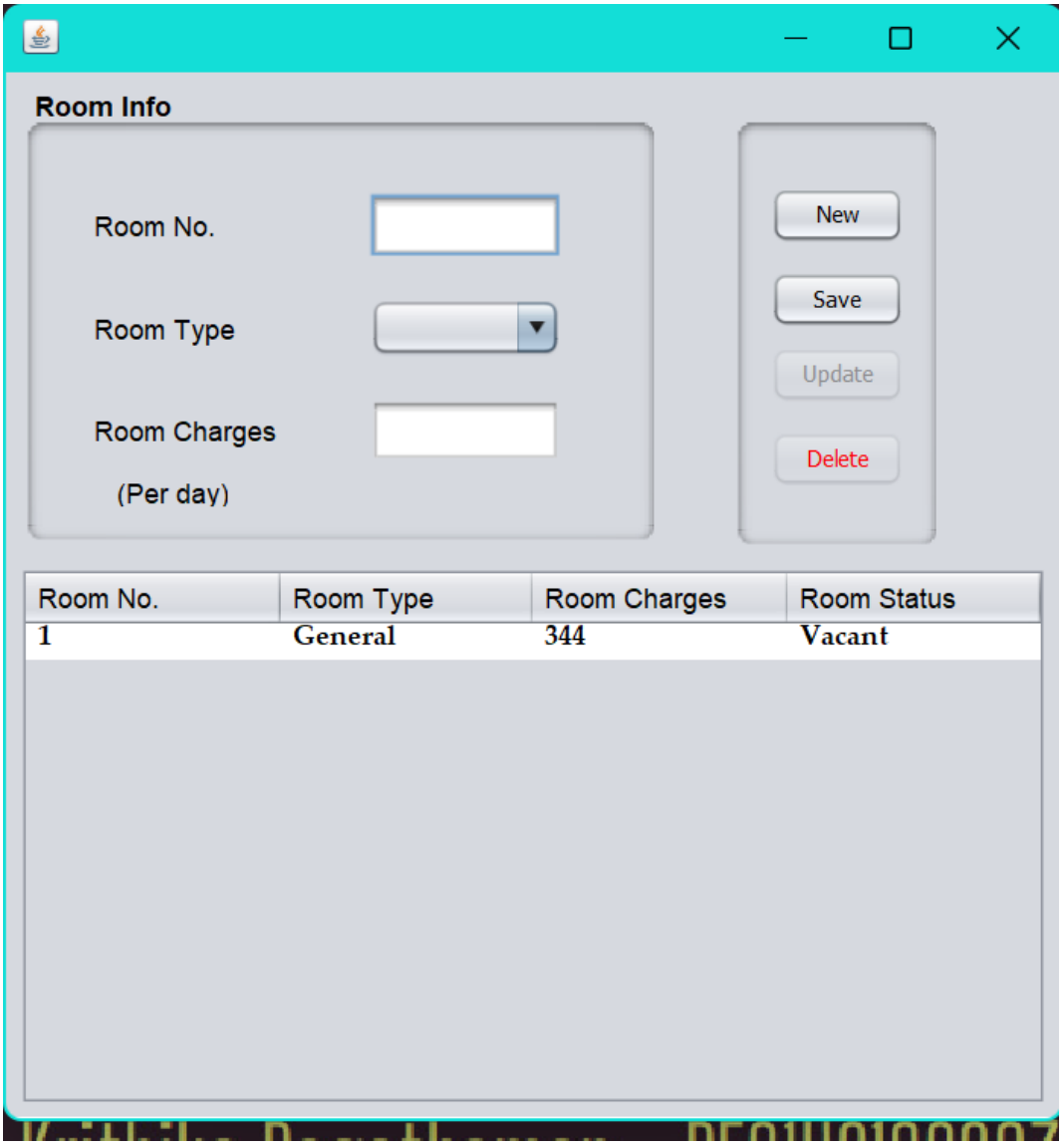

Doctor Details
Doctor ID
Full Name*
Father's Name*
Email-id
Contact no*
Address
Qualifications*
Gender
M
Blood Group
O+
Date Of Joining*
YYYY-MM-DD
Save
Update
Delete
New
Get Data


Patient Detail's
Patient ID
Name*
Address
Contact No.*
Email-id
Age*
Gender*
M
Blood Group*
O+
Information
Save
New
Update
Delete
GetData

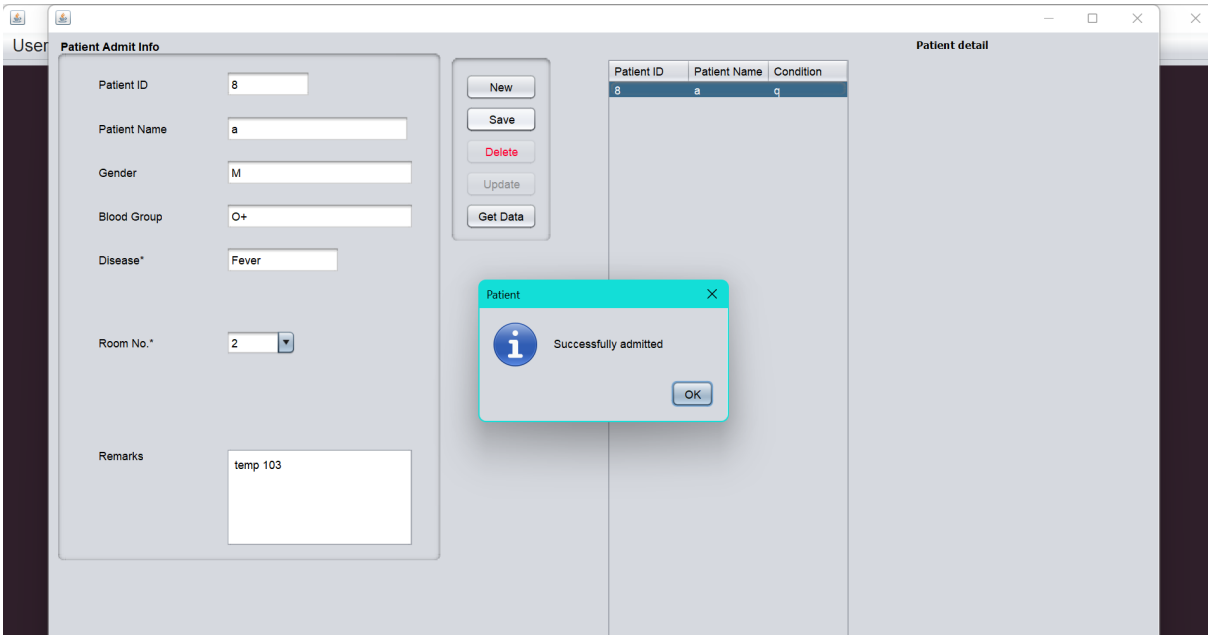

Services Info
Service Name*
Service Date*
(YYYY-MM-DD)
Patient ID
Patient Name
Service Charges*
New
Save
Delete
Update
Get Data
Patient ID
Remarks
Patient Name
8
q
a

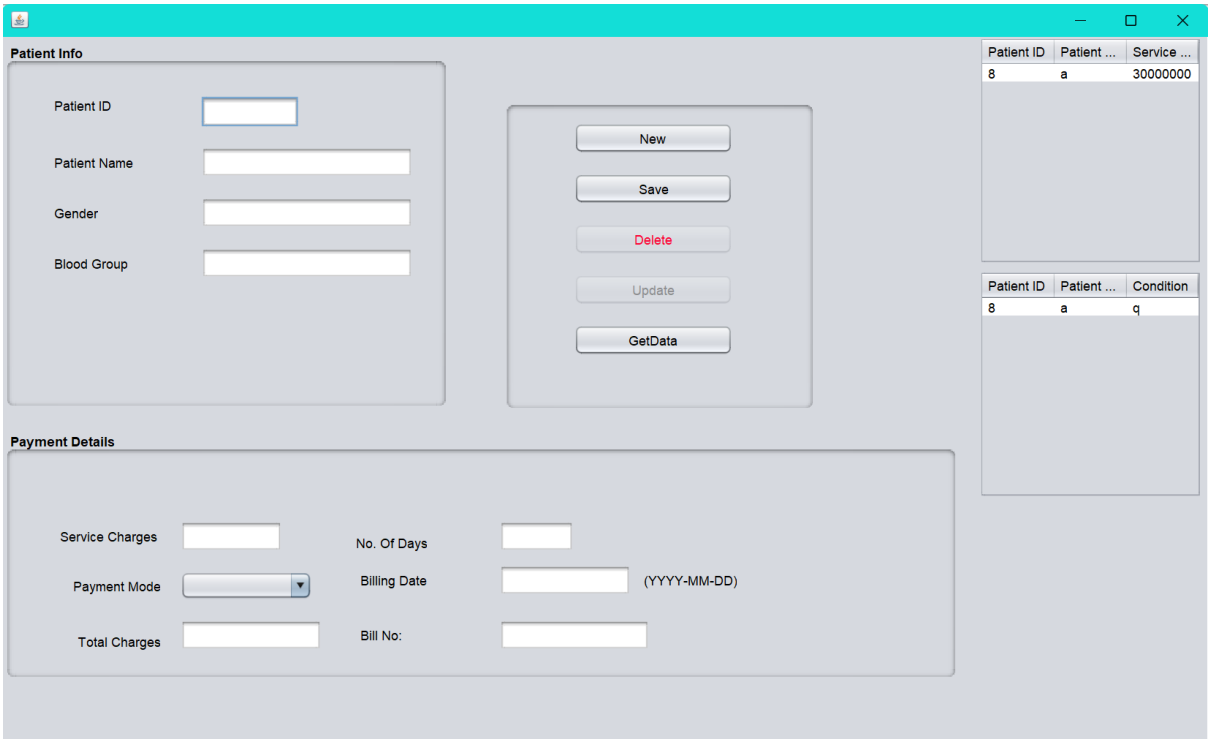


Service ID	Service ...	Service ...	Patient ID	Patient ...	Service ...
5	Scan	2020-12...	8	a	30000000



Room No.	Room Type	Room Charges	Room Status
1	General	344	Vacant


Patient Admit Info
Patient ID
8
Patient Name
a
Gender
M
Blood Group
O+
Disease*
Fever
Room No.*
2
Remarks
temp 103
New
Save
Delete
Update
Get Data
Patient detail
Patient ID
Patient Name
Condition
8
a
q
Patient
Successfully admitted
OK


Patient Info
Patient ID
Patient Name
Gender
Blood Group
New
Save
Delete
Update
GetData
Patient ID
Patient ...
Service ...
8
a
30000000
Patient ID
Patient ...
Condition
8
a
q
Payment Details
Service Charges
No. Of Days
Payment Mode
Billing Date
(YYYY-MM-DD)
Total Charges
Bill No:

8. Team member contributions

a. Use Cases

	Kartika	Krithika	Maitreyi
Ambulance		✔	
Admission to Hospital Room			✔
Bed Status	✔		
Clean Bed/Room	✔		
Consultation/ Diagnosis		✔	
Doctor Management			✔
Patient History			✔
Payment	✔		
Purchase of Medicine			✔
Registration	✔		
Surgery		✔	
Tests/Scans		✔	

b. Class Model

	Kartika	Krithika	Maitreyi
User Account		✔	
Admin	✔		
Doctor	✔		
Patient		✔	
Pharmacist			✔
Procedure			✔
Ambulance	✔		
Payment		✔	

c. Activity and State Diagrams

	Kartika	Krithika	Maitreyi
Activity 1		✔	
State 3			✔
Activity 3	✔		
Activity 4	✔		
Activity 2		✔	
State 2	✔		
State 1		✔	