UE20CS352
Object Oriented Analysis and Design With Java

Project Synopsis

Project Title :
Hospital Management System

Team Members:
PES2UG20CS306 – SAMPREET A PATIL
PES2UG20CS307 – SAMYAK P SHETTY
PES2UG20CS321 – SHAIK MD. JEELAN

Synopsis :

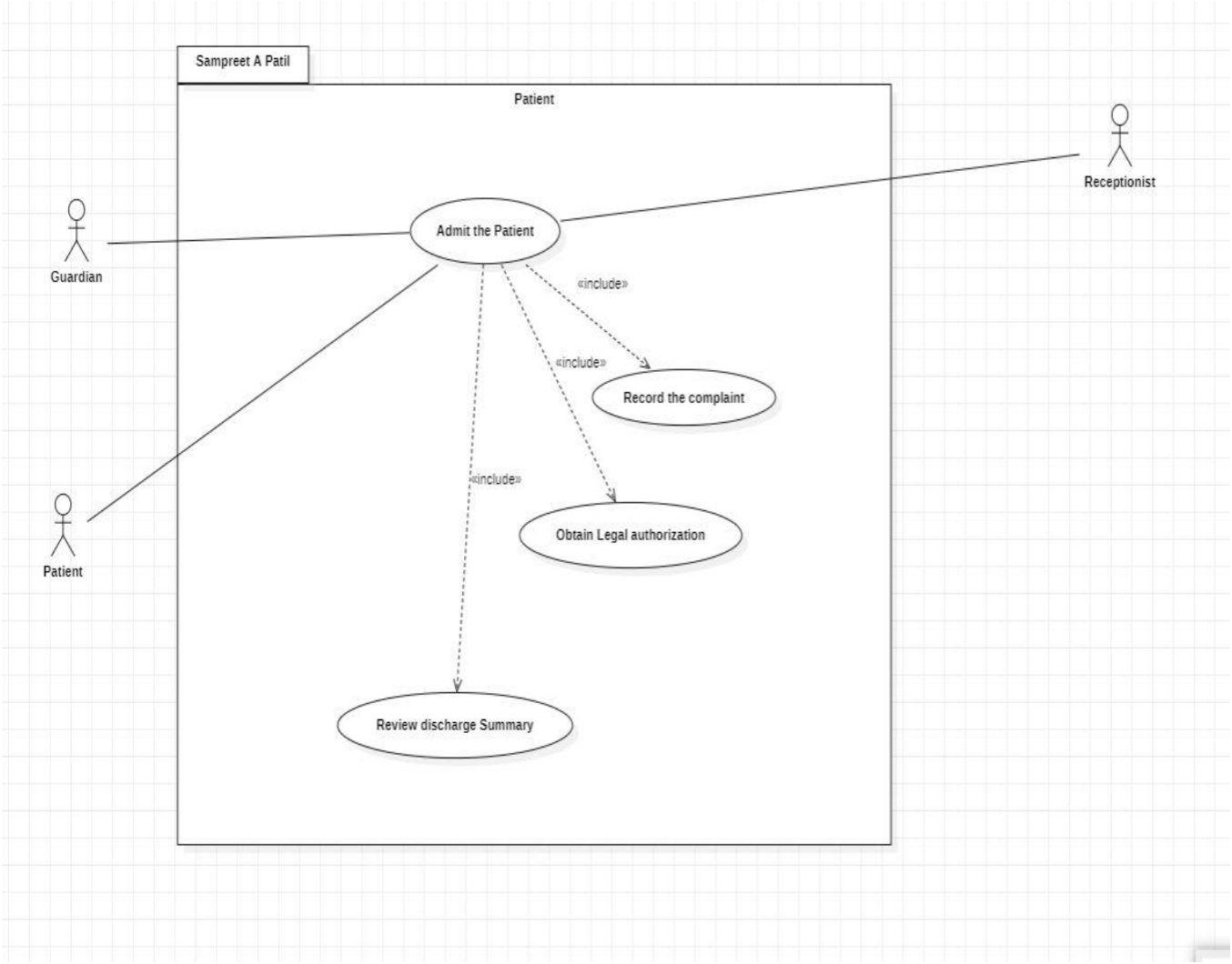
The Hospital Database System is a Java-based project that aims to provide an efficient and reliable solution for managing hospital records. The system provides functionalities for managing patient records, scheduling appointments, managing doctor information, and generating reports. It allows authorized personnel to access the system and perform operations based on their roles, ensuring data privacy and security.

The system is designed to be user-friendly, with an intuitive user interface that facilitates easy navigation and operation. It is built using object-oriented design principles and follows the MVC architecture pattern to ensure flexibility, scalability, and maintainability.

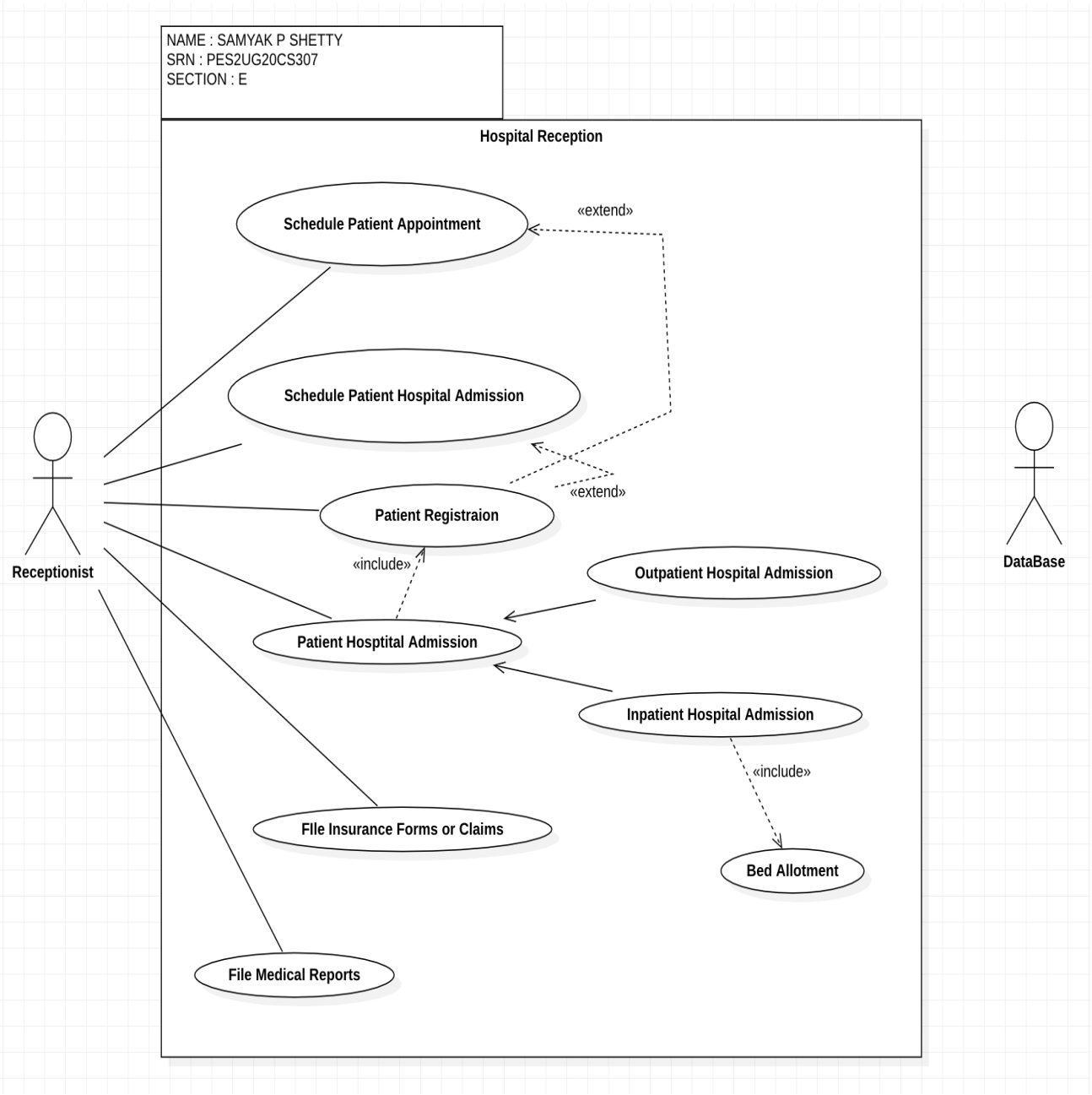
The Hospital Database System is a comprehensive solution that addresses the challenges faced by hospitals in managing patient records and appointment scheduling. By leveraging the power of technology, the system enables hospitals to optimize their operations, reduce manual errors, and improve the quality of patient care.

Use Case Diagrams :

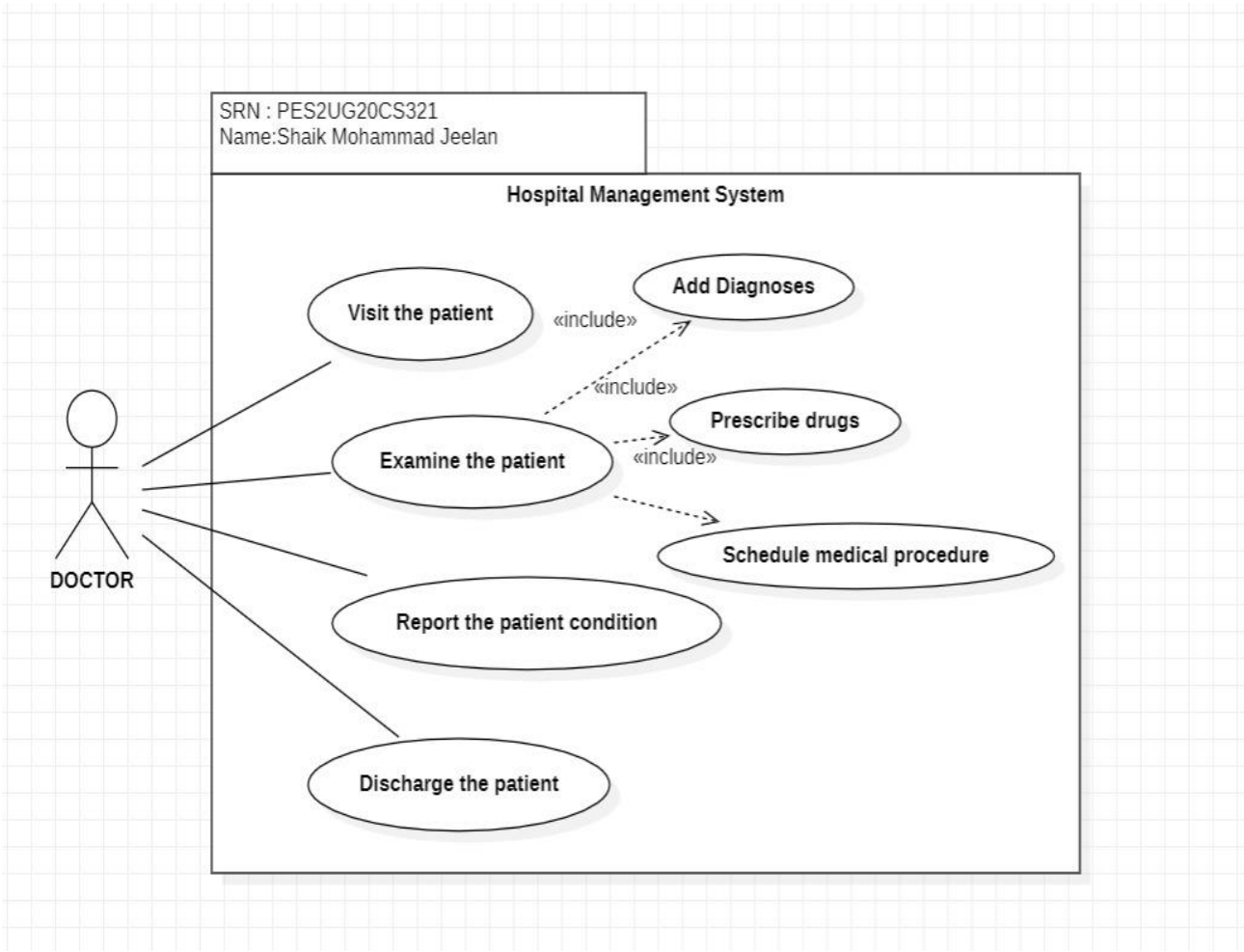
Sampreet A Patil (Patient Use Case) :



Samyak P Shetty (Receptionist Use Case) :



Shaik Mohammad Jeelan (Doctor Use Case) :



Use Case Specifications:

Sampreet A Patil (Patient Use Case):

Use Case Name: Book Appointment

Actors: Patient

Preconditions:

1. **The patient must have a valid login account.**
2. **The patient must have searched for the doctor or department they wish to book an appointment with.**
3. **The patient must have selected a date and time for the appointment.**

Basic Flow of Events:

1. **The patient logs in to the hospital management system.**
2. **The patient selects the "Book Appointment" option from the patient dashboard.**
3. **The system displays a search interface where the patient can search for the doctor or department they wish to book an appointment with.**
4. **The patient enters the name of the doctor or department and selects the relevant option from the list of search results.**
5. **The system displays the availability of the selected doctor or department for the selected date and time.**
6. **The patient selects a suitable appointment time from the available options.**
7. **The system confirms the selected appointment and displays the appointment details to the patient.**
8. **The patient can view the appointment details on their dashboard and can also choose to cancel the appointment if needed.**

Alternate Flows:

1. **If there are no available appointment slots for the selected doctor or department, the system displays a message indicating the unavailability of appointments and provides the patient with an option to search for another doctor or department.**
2. **If the patient cancels the appointment, the system updates the appointment status as cancelled and notifies the doctor and receptionist.**
3.

Postconditions:

1. **The patient successfully books an appointment with the selected doctor or department.**
2. **The appointment details are stored in the system and can be viewed by the patient, doctor, and receptionist.**

Samyak P Shetty (Receptionist Use Case) :

Use Case Name: Register Patient

Scope: Hospital Management System

Primary Actor: Receptionist

Level: User Goal

Preconditions:

1. **The Receptionist has access to the Hospital Management System**
2. **The Receptionist is logged in**

Main Success Scenario:

1. **The Receptionist selects the "Register Patient" option from the main menu.**
2. **The system prompts the Receptionist to enter the patient's personal details such as name, address, contact number, etc.**
3. **The Receptionist enters the patient's personal details.**
4. **The system validates the entered details and generates a unique patient ID for the patient.**
5. **The system displays the patient ID to the Receptionist and prompts to enter additional information such as insurance details.**
6. **The Receptionist enters the insurance details (if any).**
7. **The system saves the patient details to the database and confirms the registration to the Receptionist.**

Alternate Flows:

1. **If the system cannot validate the entered details, it displays an error message and prompts the Receptionist to re-enter the correct details.**

Postconditions:

1. **The patient details are saved to the database with a unique patient ID.**
2. **The Receptionist can view the registered patient details.**

Shaik Mohammad Jeelan (Doctor Use Case) :

Use Case Name: Prescribe Drugs

Actors: Doctor

Preconditions:

1. **Doctor must be logged in to the hospital system.**
2. **Patient information must be in the hospital system.**
3. **The doctor must have access to the patient's medical records.**
4. **The doctor must have determined that medication is necessary for the patient's treatment.**

Postconditions:

1. **The prescription details are saved in the patient's medical records.**
2. **The prescription details are sent to the Pharmacy.**

Basic Flow of Events:

1. **The doctor selects the patient's medical record.**
2. **The doctor reviews the patient's medical history and determines the appropriate medication.**
3. **The doctor enters the medication details into the system, including the name, dosage, and frequency of administration.**
4. **The system verifies that the medication is not contraindicated for the patient.**
5. **The system prompts the doctor to confirm the prescription details.**
6. **The doctor confirms the prescription details.**
7. **The system saves the prescription details in the patient's medical records.**
8. **The system sends the prescription details to the Pharmacy.**

Alternate Flows:

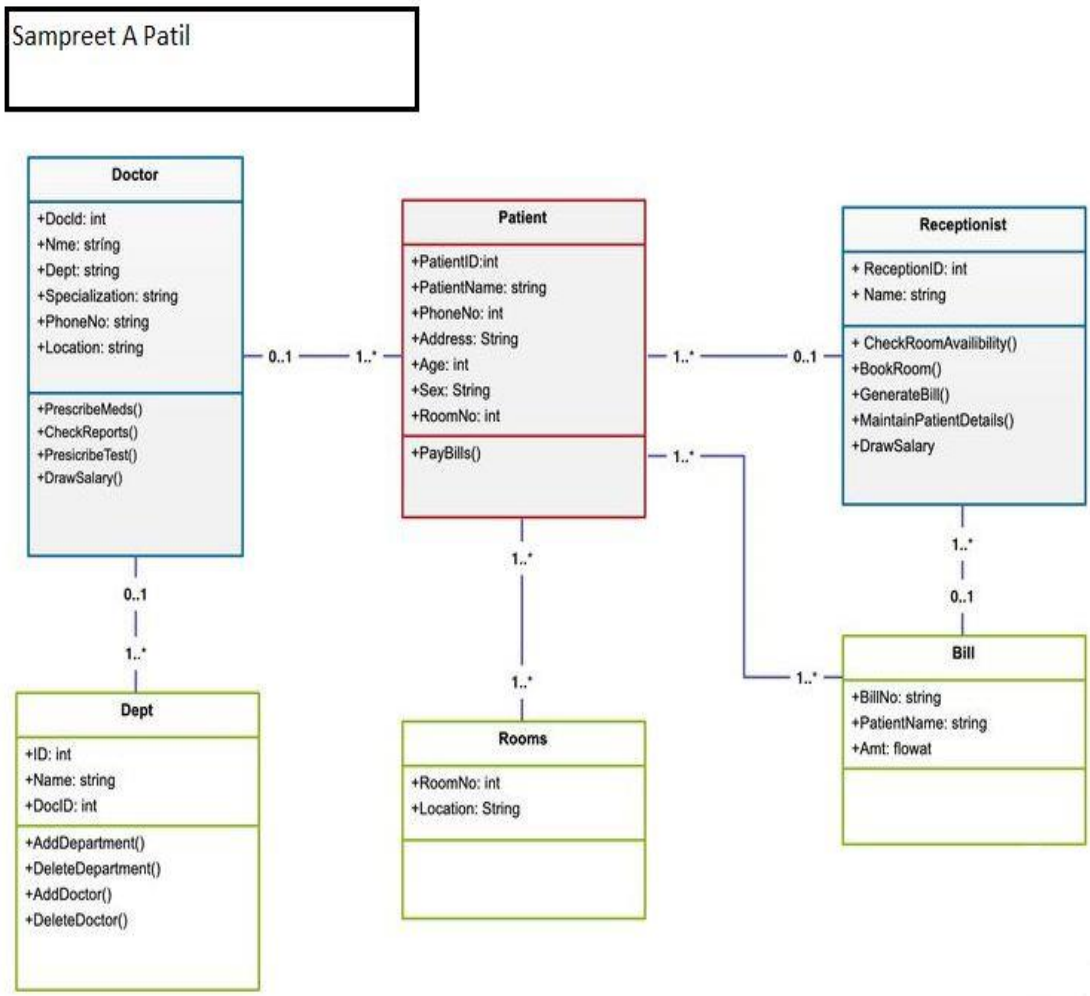
1. **If the medication is contraindicated for the patient, the system will prompt the doctor to select an alternative medication.**
2. **If the prescription details are incomplete or invalid, the system will prompt the doctor to correct them before saving the prescription details.**
3. **If the prescription details cannot be sent to the Pharmacy, the system will prompt the doctor to contact the Pharmacy directly.**

Exceptions:

1. **The system is unable to save the prescription details due to a technical error. The doctor will be prompted to try again later or contact technical support if the problem persists.**

Use Class Diagrams :

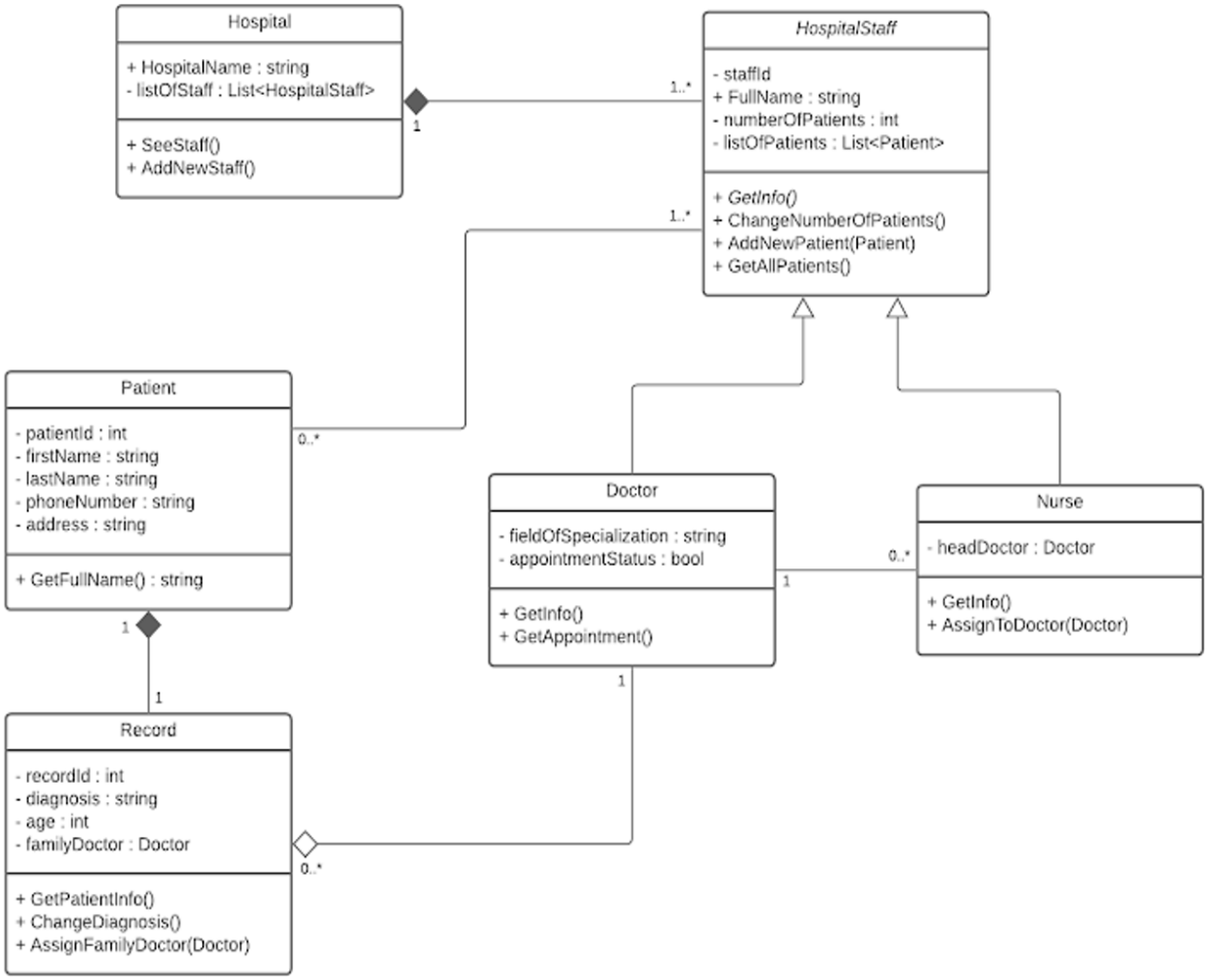
Sampreet A Patil :



Use Class Diagram

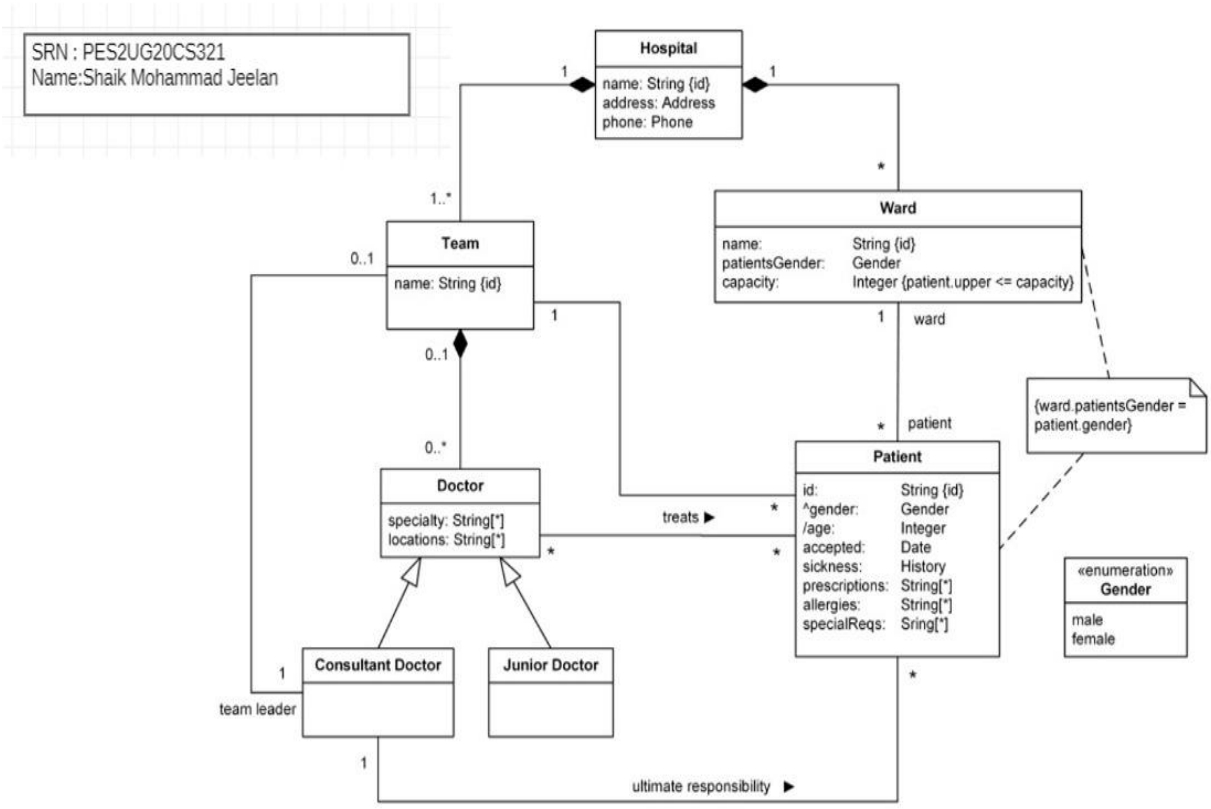
Samyak P Shetty :

NAME : SAMYAK P SHETTY
SRN : PES2UG20CS307
SECTION : E



Use Case Diagram :

Shaik Mohammad Jeelan :



Conclusion :

In conclusion, the Hospital Management System using Java is a comprehensive software system designed to simplify the process of managing a hospital's operations. It is an efficient system that enables doctors, nurses, and other staff members to easily access and manage patient records, appointments, diagnoses, and treatments. The system's user-friendly interface, automated features, and real-time data management make it a valuable tool for hospitals of any size.