

**Rohan Nyati**  
**500075940**  
**R177219148**  
**B-5 AI&ML SEM-5**

## **Assignment-1**

Q.1)What is Object Oriented Analysis and Design Software Development Life Cycle? Briefly explain that.

The object-oriented life cycle model considers 'objects' as the basis of the software engineering process. The development team starts by observing and analyzing the system they intend to develop before defining the requirements. Once the process is over, they focus on identifying the objects of the system. Now, an object could be anything; it can have a physical existence like a customer, car, etc. An object also constitutes intangible elements like a process or a project.

It's a structured method for analyzing, designing a system by applying the object-orientated concepts, and developing a set of graphical system models during the development life cycle of the software.

### **OOAD In The SDLC :**

The software life cycle is typically divided up into stages going from abstract descriptions of the problem to designs then to code and testing and finally to deployment.

The earliest stages of this process are analysis (requirements) and design.

In analysis developers work with users and domain experts to define what the system is supposed to do. Implementation details are supposed to be mostly or totally ignored at this phase.

The goal of the analysis phase is to create a model of the system regardless of constraints such as appropriate technology. This is typically done via use cases and *abstract* definition of the most important objects using conceptual models.

The design phase refines the analysis model and applies the needed technology and other implementation constraints.

It focuses on describing the objects, their attributes, behavior, and interactions. The design model should have all the details required so that programmers can implement the design in code.

Q.2)What is UML? Describe the building blocks of it.

**Unified Modeling Language (UML)** is a general purpose modelling language. The main aim of UML is to define a standard way to visualize the way a system has been designed. It is quite similar to blueprints used in other fields of engineering.

UML is not a programming language, it is rather a visual language. We use UML diagrams to portray the behavior and structure of a system. UML helps software engineers, businessmen and system architects with modelling, design and analysis.

UML is composed of three main building blocks, i.e., things, relationships, and diagrams. Building blocks generate one complete UML model diagram by rotating around several different blocks. It plays an essential role in developing UML diagrams. The basic UML building blocks are:

1.)Things: Anything that is a real world entity or object is termed as things. It can be divided into several different categories:

- Structural things
- Behavioral things
- Grouping things
- Annotational things

2.)Relationships : It illustrates the meaningful connections between things. It shows the association between the entities and defines the functionality of an application. There are four types of relationships given below:

- Dependency
- Association
- Generalization
- Realization

3.)Diagrams : The diagrams are the graphical implementation of the models that incorporate symbols and text. Each symbol has a different meaning in the context of the UML diagram. There are thirteen different types of UML diagrams that are available in UML 2.0, such that

each diagram has its own set of symbols. And each diagram manifests a different dimension, perspective, and view of the system.

UML diagrams are classified into three categories that are given below:

- Structural Diagram
- Behavioral Diagram
- Interaction Diagram

Q.3) Describe the following terms with examples

A) Inception : The first phase of the Rational Unified Process that deals with the original conceptualization and beginning of the project.

B) Elaboration : The second phase of the Rational Unified Process that allows for additional project planning including the iterations of the construction phase.

C) Composition : Composition is a specialized form of aggregation. In composition, if the parent object is destroyed, then the child objects also cease to exist. Composition is actually a strong type of aggregation and is sometimes referred to as a “death” relationship. As an example, a house may be composed of one or more rooms. If the house is destroyed, then all of the rooms that are part of the house are also destroyed.

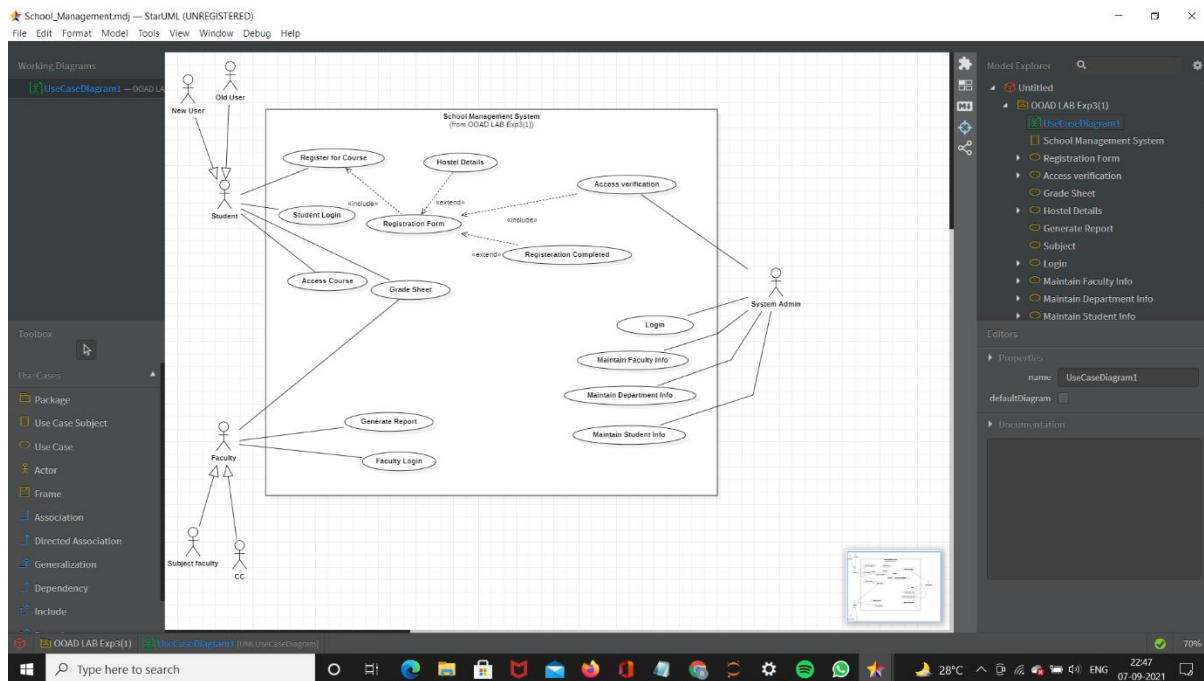
D) Aggregation : Aggregation is a specialized form of association between two or more objects in which each object has its own life cycle but there exists an ownership as well. Aggregation is a typical whole/part or parent/child relationship but it may or may not denote physical containment. An essential property of an aggregation relationship is that the whole or parent (i.e. the owner) can exist without the part or child and vice versa.

## EXPERIMENT-2

Q.1 Consider any given case study and identify different entities and relationships among them.

A relationship between two entities signifies that the two entities are associated with each other somehow. For example, a student might enroll in a course. The entity Student is therefore related to Course, and a relationship is presented as a connector connecting between them.

Example Uml Diagram Shown depicting different entities and relationships



Q.2 Provide the examples of generalization, aggregation and realization .

## Generalization

Generalization uses a “is-a” relationship from a specialization to the generalization class. Common structure and behaviour are used from the specialization to the generalized class. At a very broader level you can understand this as inheritance. Why I take the term inheritance is, you can relate this term very well. Generalization is also called a “Is-a” relationship.



*Example:* Consider there exists a class named Person. A student is a person. A faculty is a person. Therefore here the relationship between student and person, similarly faculty and person is generalization.

## Aggregation

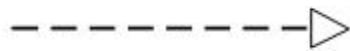
Aggregation is a special case of association. A directional association between objects. When an object ‘has-a’ another object, then you have got an aggregation between them. Direction between them specified which object contains the other object. Aggregation is also called a “Has-a” relationship.



*Example:* A Library contains students and books. Relationship between library and student is aggregation. A student can exist without a library and therefore it is aggregation.

## **Realization**

Realization is a relationship between the blueprint class and the object containing its respective implementation level details. This object is said to realize the blueprint class. In other words, you can understand this as the relationship between the interface and the implementing class.



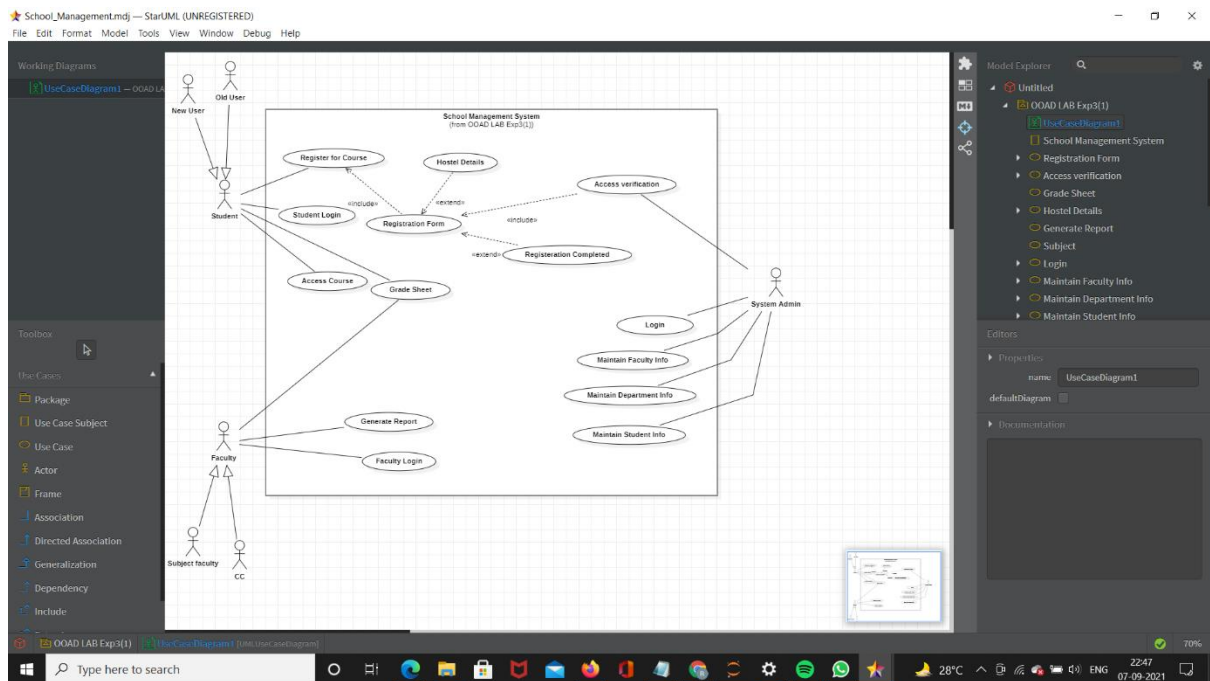
*Example:* A particular model of a car 'Nano' that implements the blueprint of a car realizes the abstraction.

## **Assignment-3**

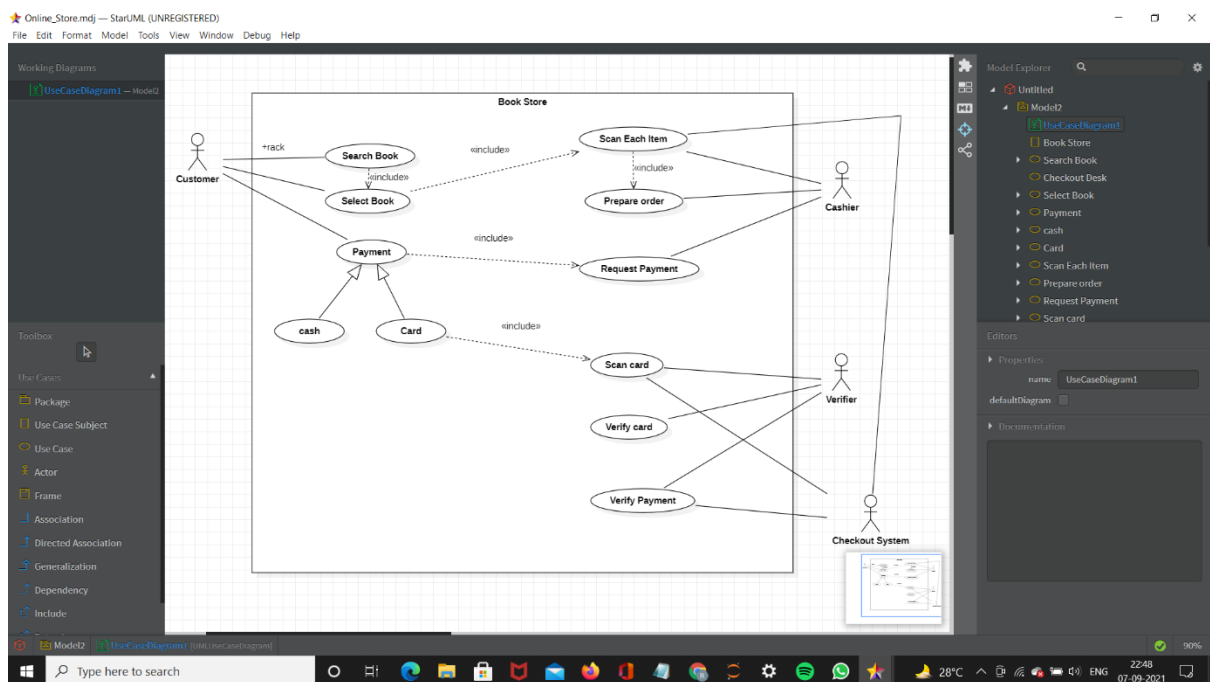
Draw Use Case Diagram for following:

a) An organization wants to develop the system based on following Requirements:

1. Students have to register first through fill up the registration form.
2. Faculty can prepare the grade sheet of concern subjects.
3. Student and faculty can see the result of each student.
4. Faculty can generate the report about registered students.
5. System can verify the students Registration Details First.
6. Student may have to fill up the hostel details during registration process



**b)** Consider a bookstore in a shopping mall. The customer selects the books from racks to purchase. The customer brings selected books to cashier. The cashier scans each item with checkout system to prepare an order. The cashier requests to customer for payment. The customer gives credit card to cashier. The verifier and checkout system scans the card. The verifier accepts the card and payment is accepted. Customer signs the credit card slip. The purchased books are handed over to customer.

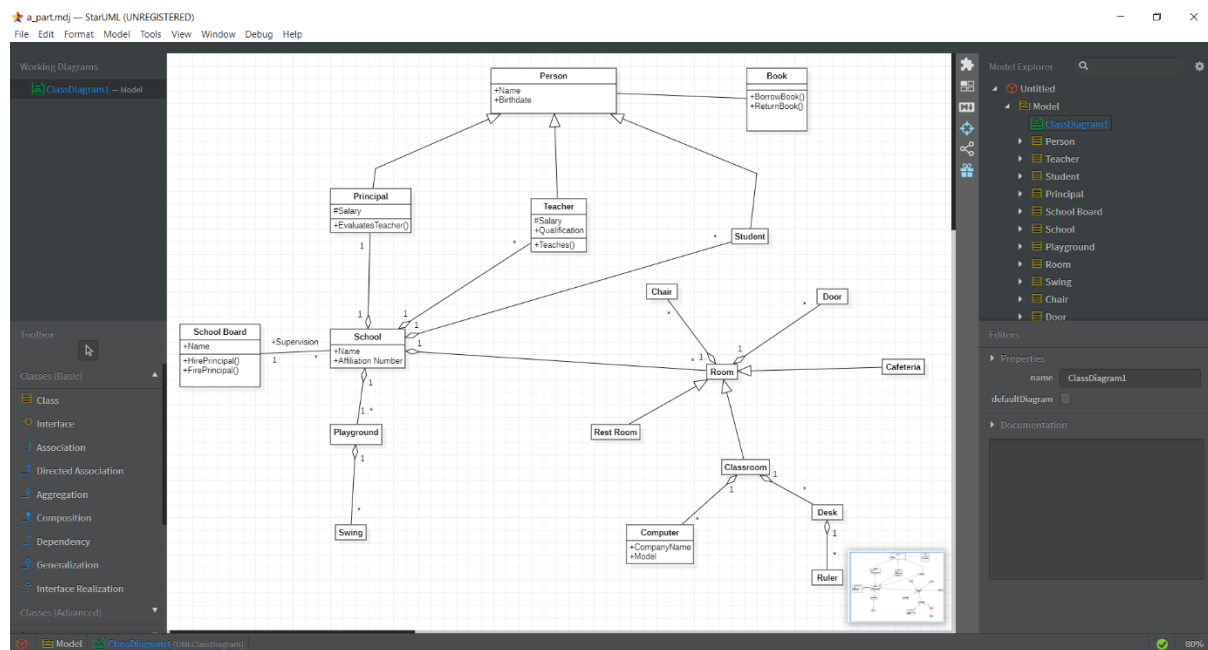


# Experiment-4

## Experiment No: 04 Class Diagram and Object Diagram

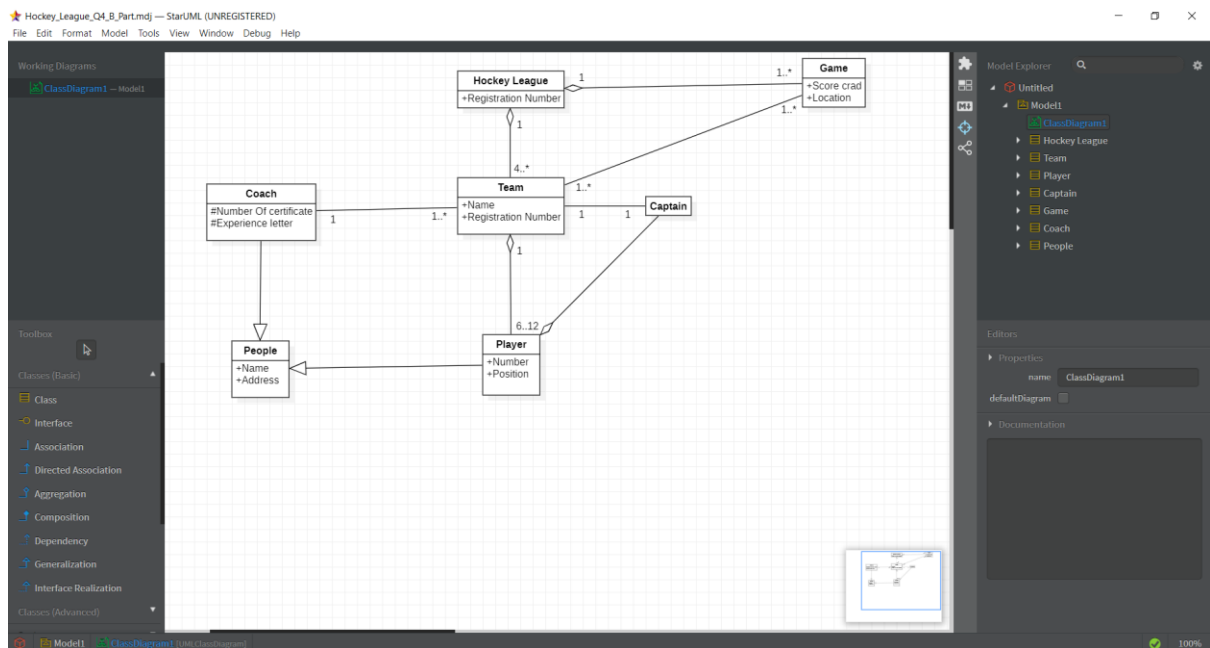
A) Draw the class diagram for the following:

A school has a principal, many students, and many teachers. Each of these persons has a name, birthdate, and may borrow and return books. Teachers and the principal are both paid a salary; the principal evaluates the teachers. A school board supervises multiple schools and can hire and fire the principal for each school. A school has many playgrounds and rooms. A playground has many swings. Each room has many chairs and doors. Rooms include restrooms, classrooms, and the cafeteria. Each classroom has many computers and desks. Each desk has many rulers.



B) Draw class Diagram for a Hockey League:

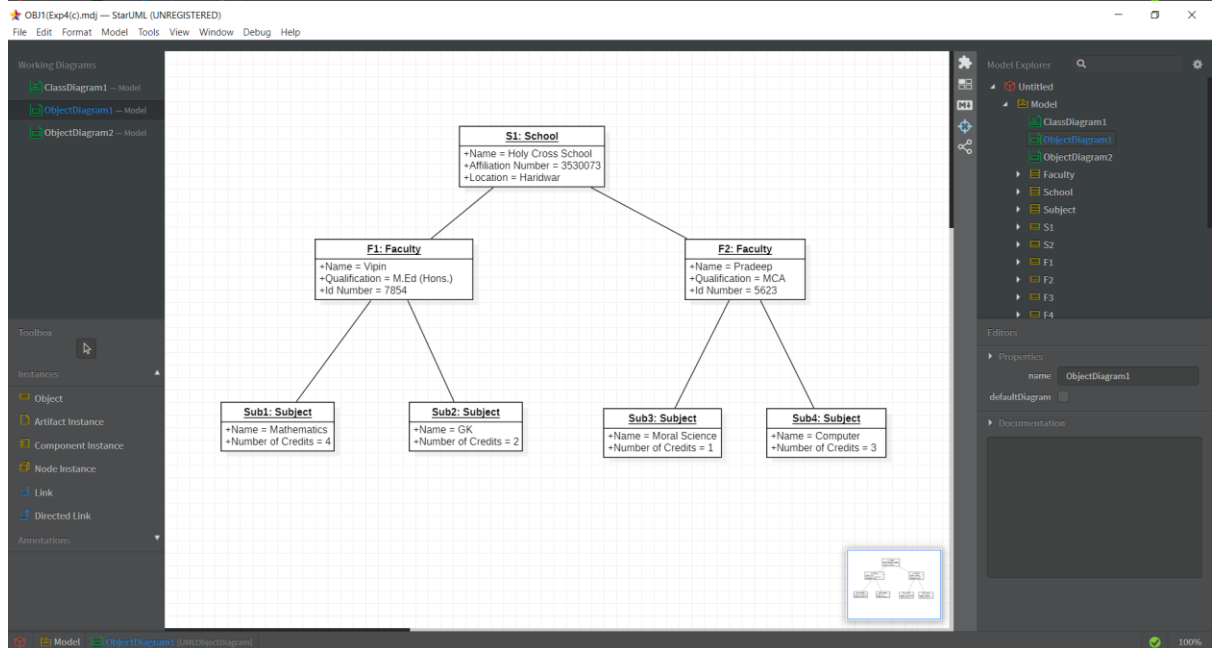
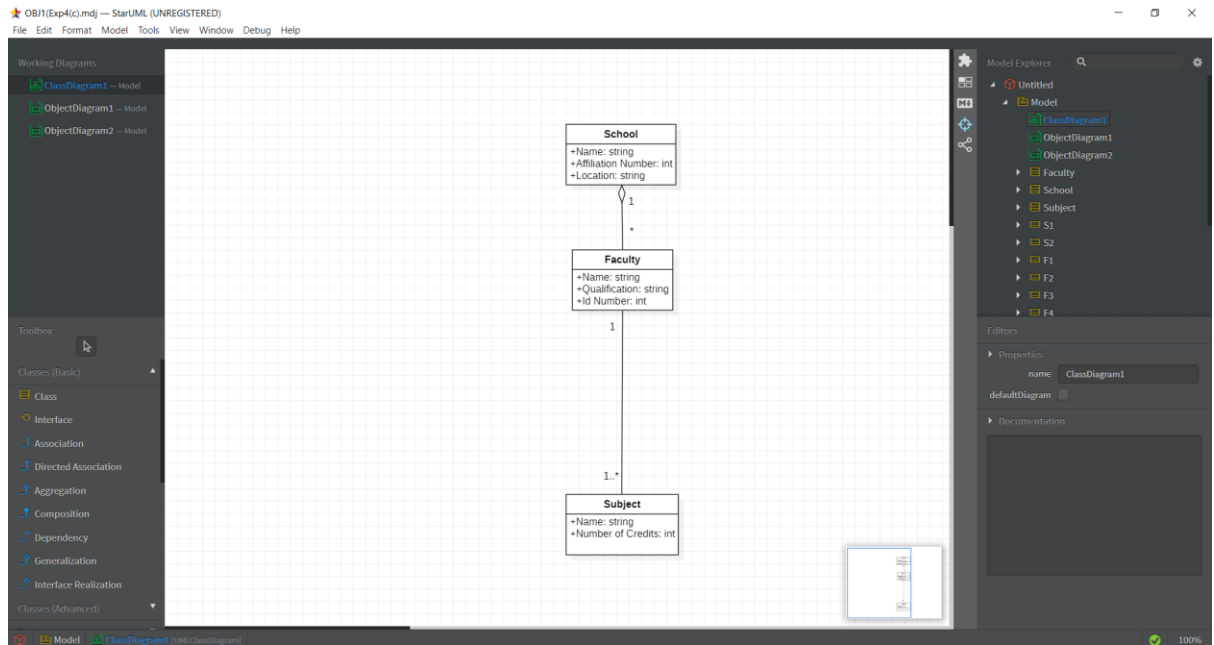
Draw a UML class diagram representing the following elements from the problem domain for a Hockey League. A Hockey League is made up of at least 4 hockey teams. Each hockey team is composed of 6-12 players and 1 player is the captain of the team. A team has a name and registration number. Player has a name, a number and a position. Hockey teams play games against each other. Each game has a score card and a location. Each team also has a coach. A coach has a number of certificates and experience letter. A coach can provide coaching to multiple teams. Coaches and players are people having name and address. A league also has registration number and teams dynamically participate in the league. Draw a class diagram for this information with appropriate relationships and multiplicities.

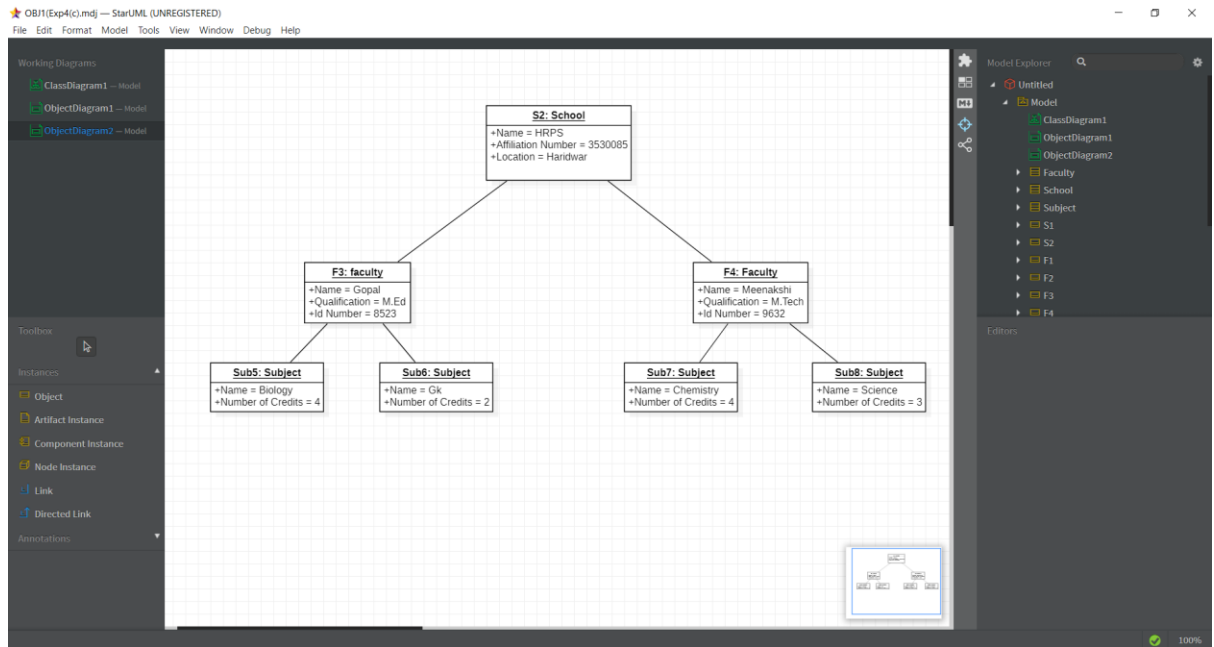


C) Design a class diagram and any two **object diagrams** for each statement:

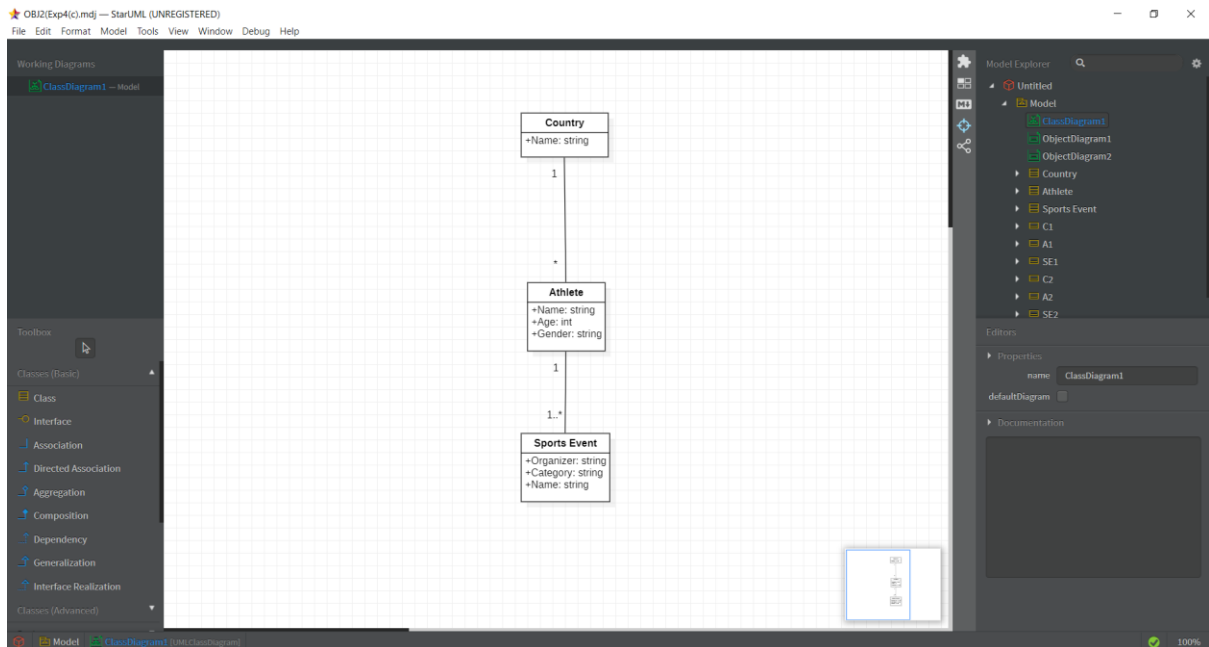
1. A faculty takes many subjects in a school.

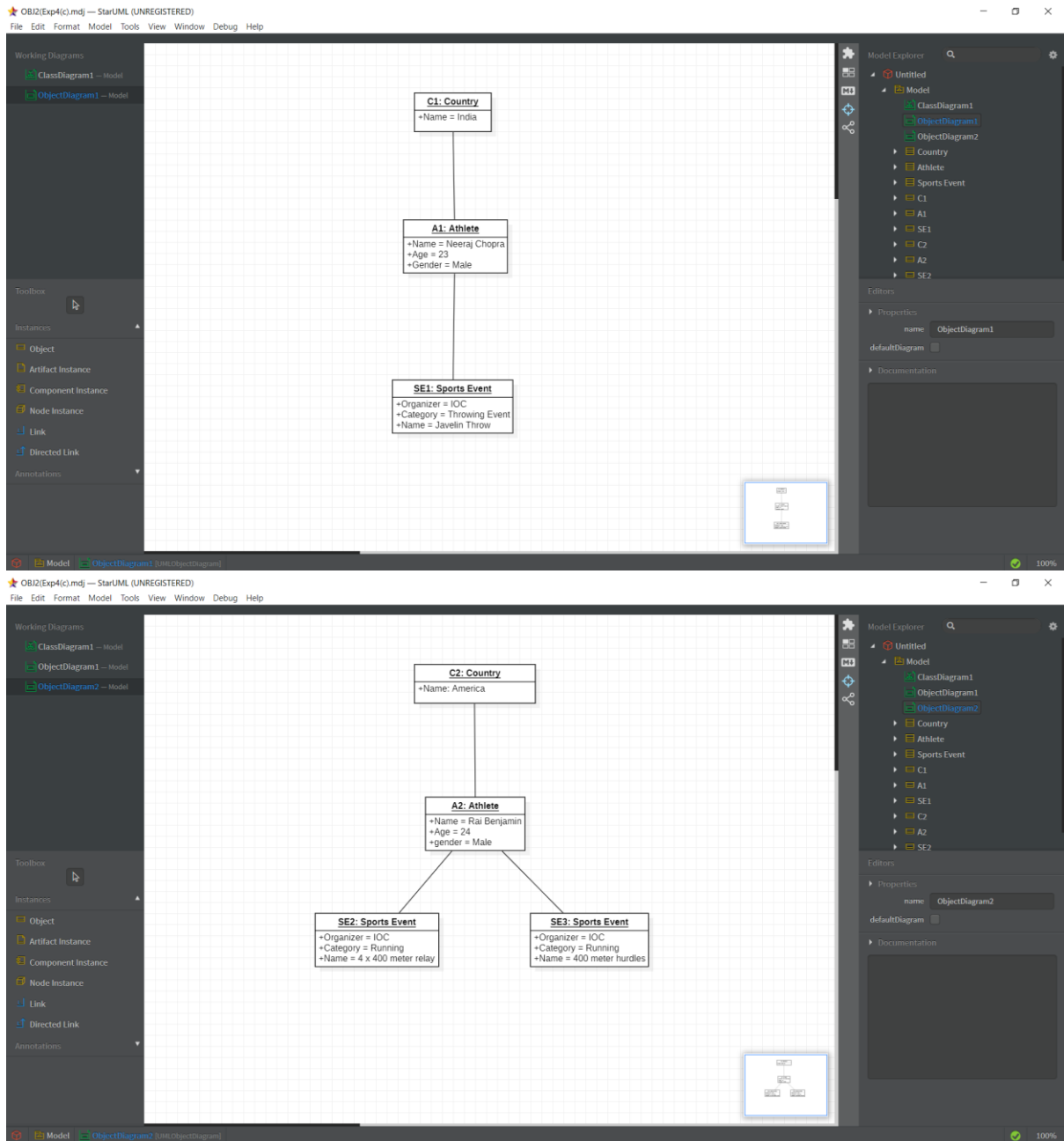






2. Athlete may represent country in various sports events.





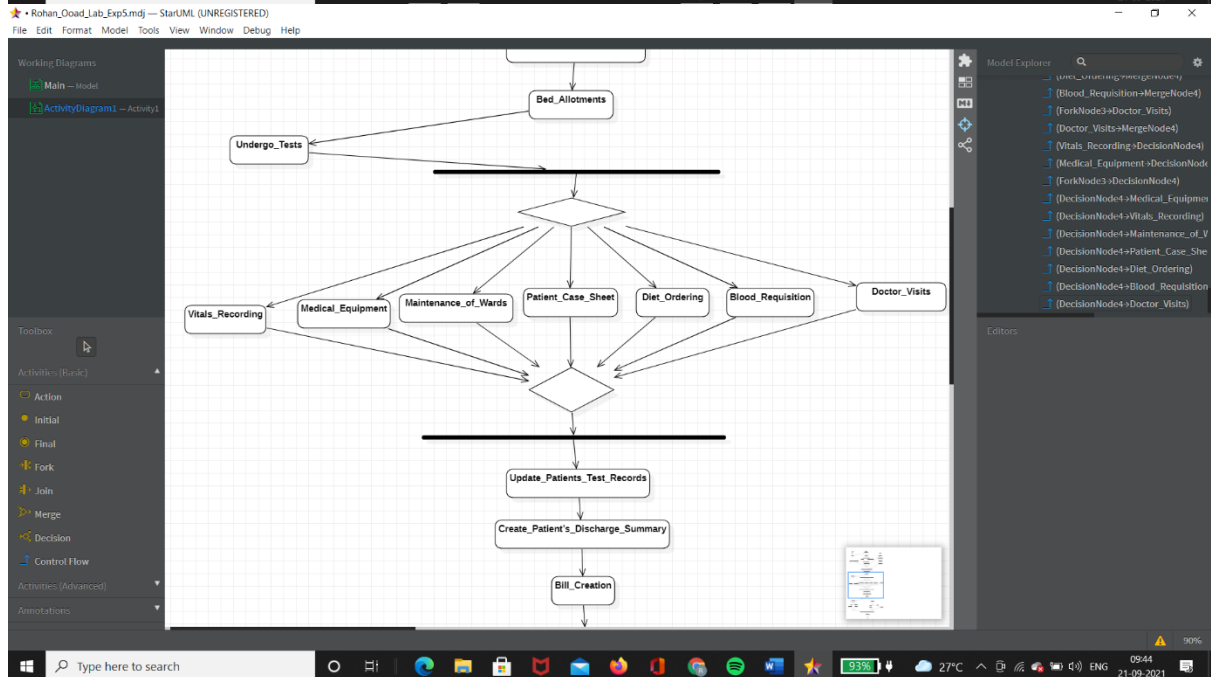
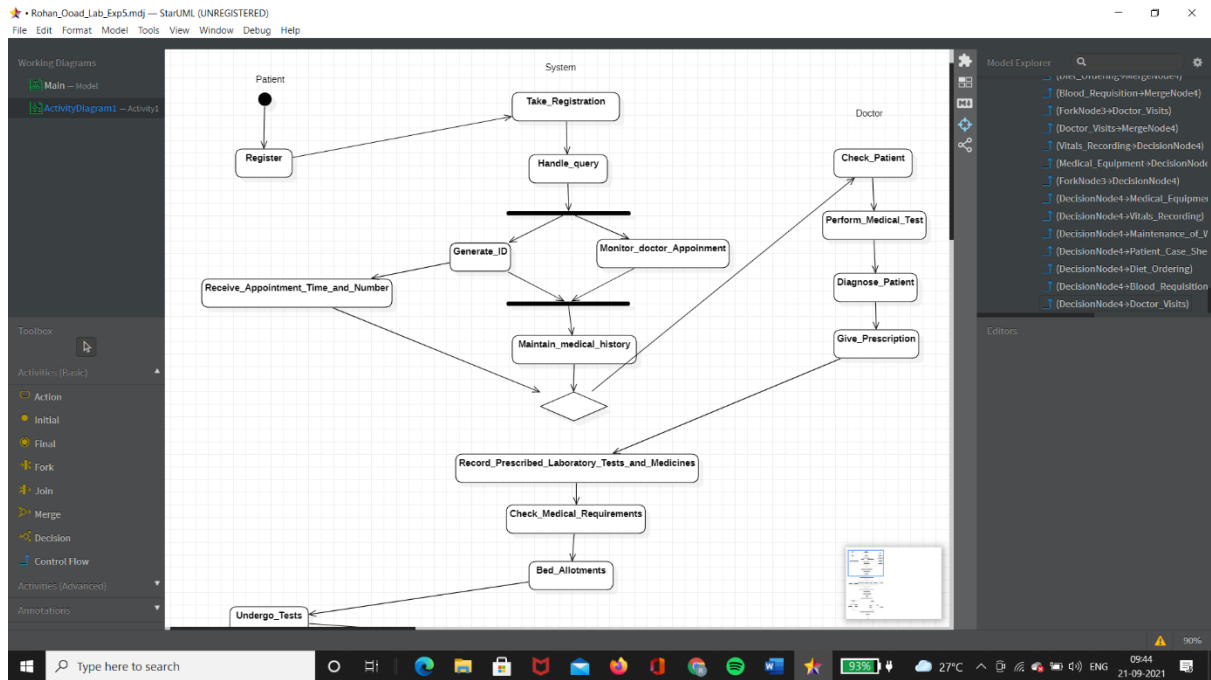
## Experiment-5

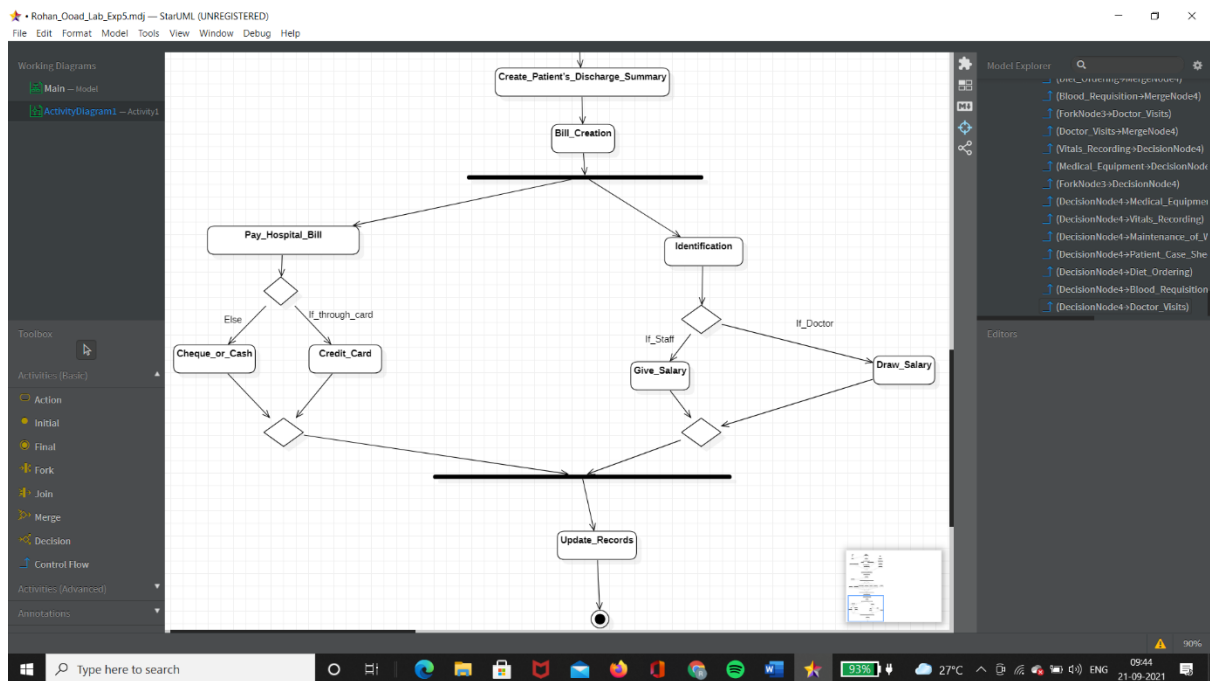
### Experiment No: 05 Activity and Swimlane Diagram

a) Design detailed Activity diagram of Hospital Management System for the following:

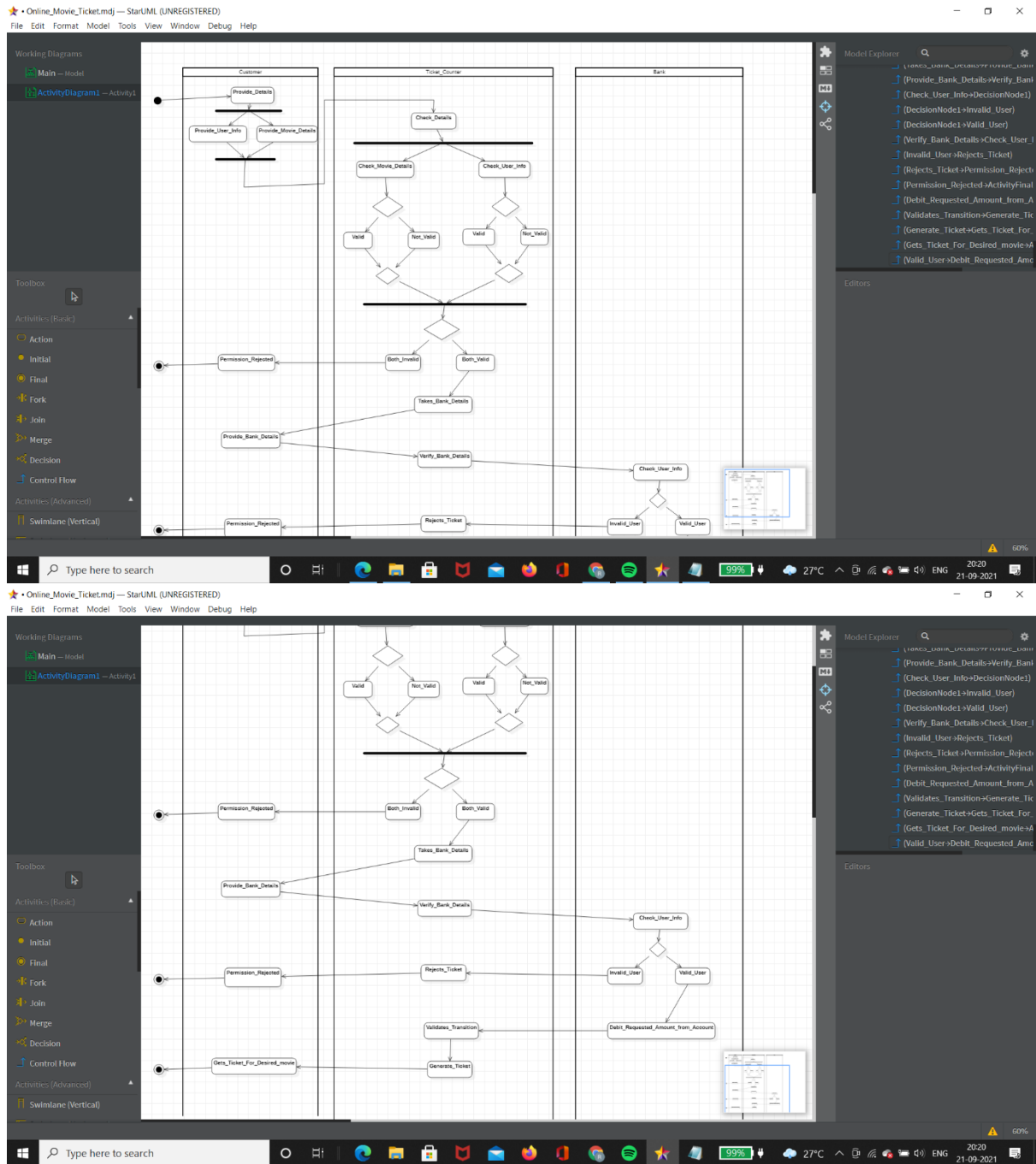
1. System helps in registering information about patients and handles patient's query.
2. A unique ID is generated for each patient after registration for maintains medical history of patient.
3. System also monitors the doctor appointments, when the ID is generated the patient receives the appointment time and number from the receptionist and accordingly visits the doctor.
4. This system also deals with testing appointments as and when ID is generated the

- patient receives the appointment time and number and accordingly undergoes the test.
5. It also deals with bed allotments to various patients by checking their ID. It also undergoes various operations by diagnosing the patients.
  6. The system identifies whether the person is a doctor or staff and handles various activities such as draw salary and gives salary, also it adds doctor/staff information into database.
  7. This system is responsible for handling various other activities like deleting, editing doctor/staff information into the database.
  8. As per doctor diagnoses the patient, gives treatment and gives suggestions to patients and prescribe laboratory tests and medicines.
  9. This system also takes care of medical equipment, doctor visit, vitals recording, patient case sheet, diet ordering, blood requisition, transfer information and discharge information, maintenance of wards, inter and intra wards transfers also it generates patient's discharge summary which includes patients health at the time of discharge, medical history, various diagnosis and drug prescriptions, history of patients illness and course in hospital.
  10. Patient can pay bill through credit card, cash or cheque whose information is maintained by this system.

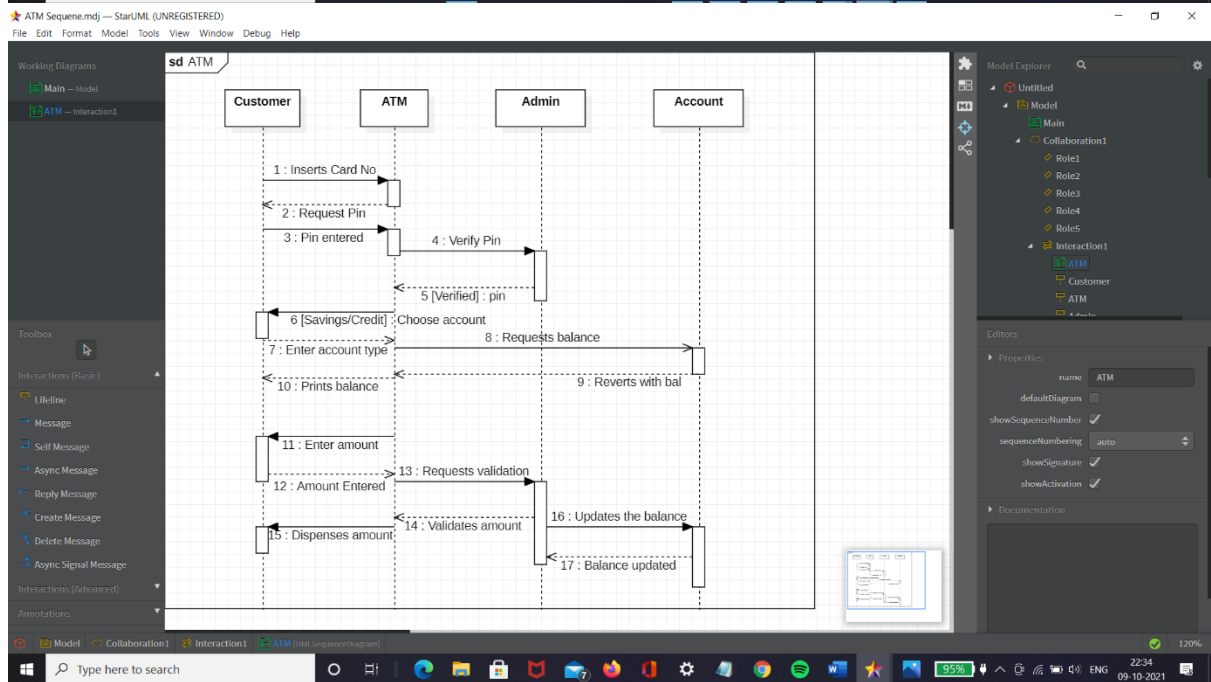
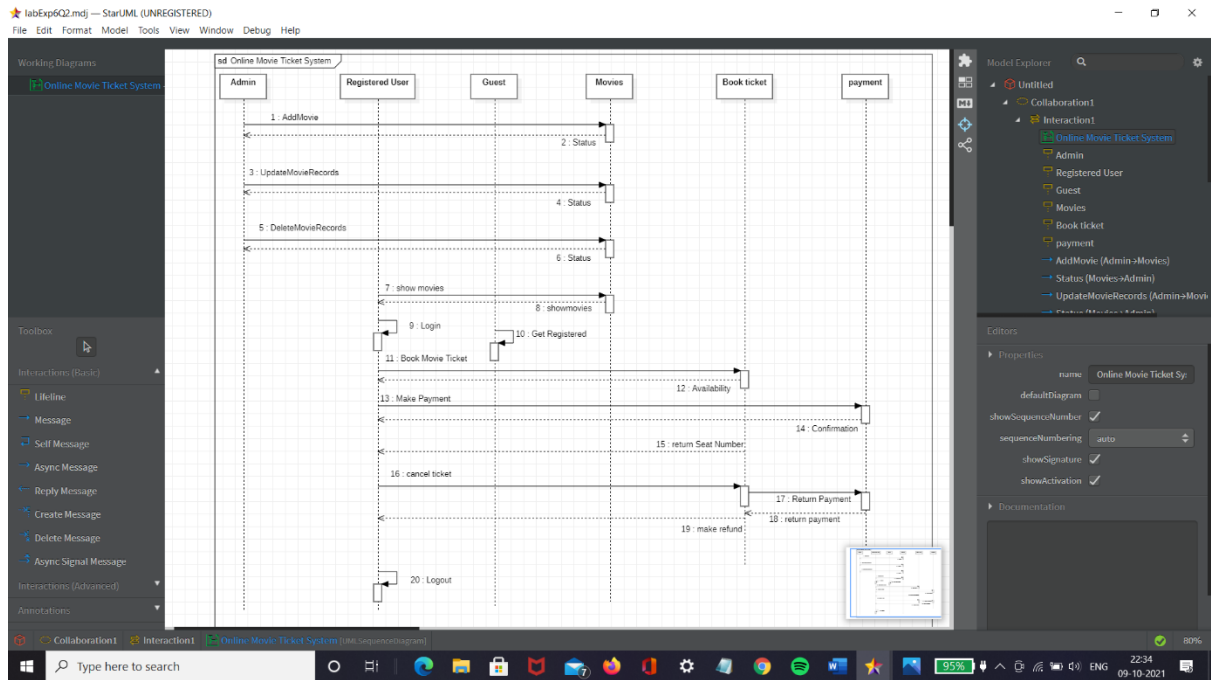




**b) Design Swimlane diagram for booking an “online movie ticket”.**

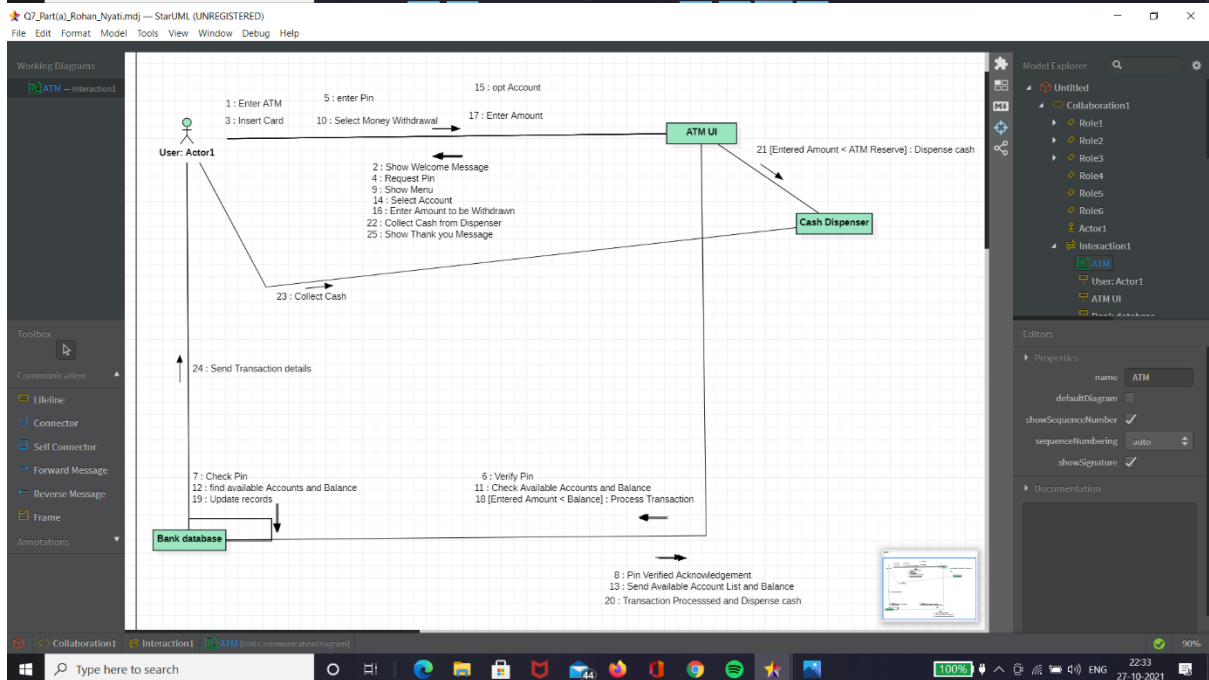
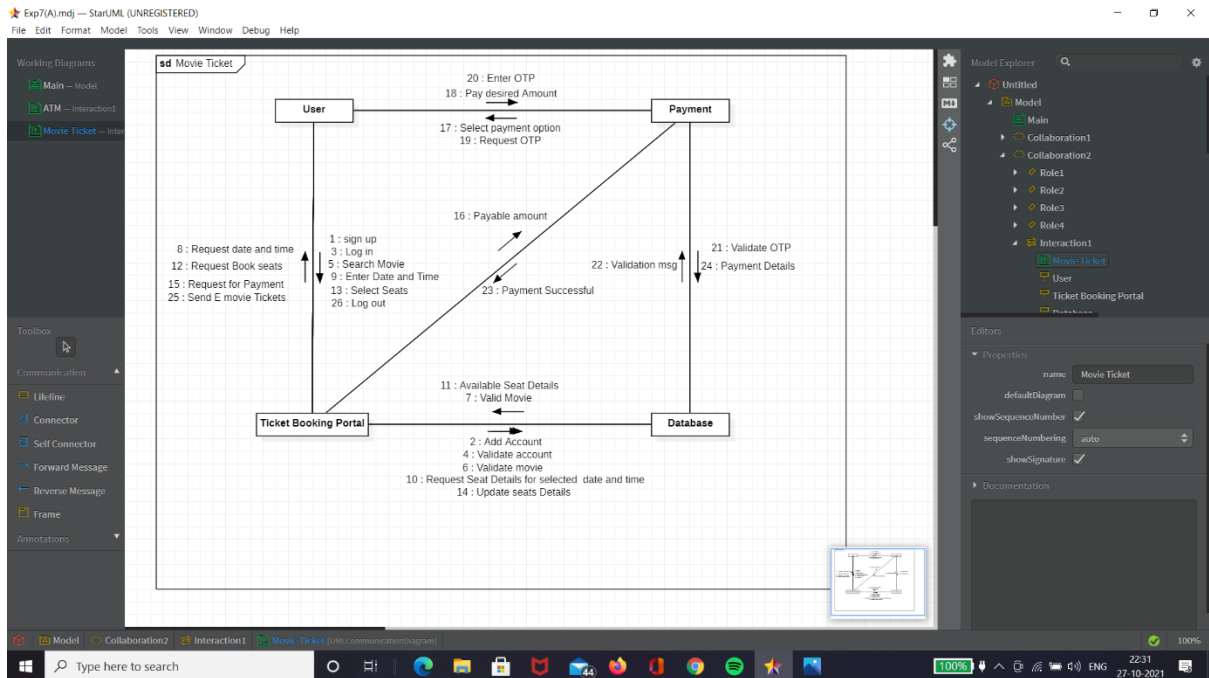


## Assignment-6



# Assignment-7

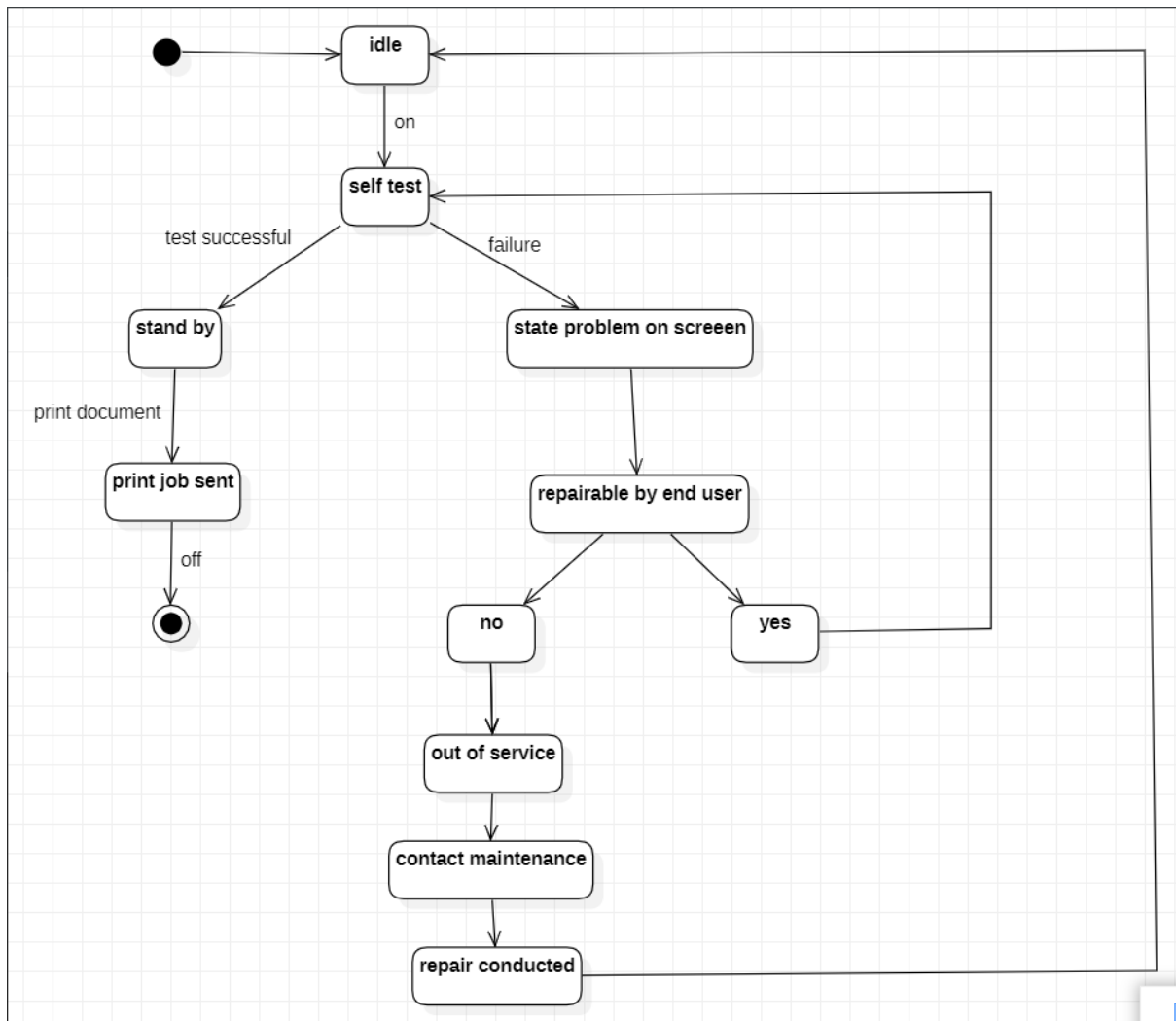




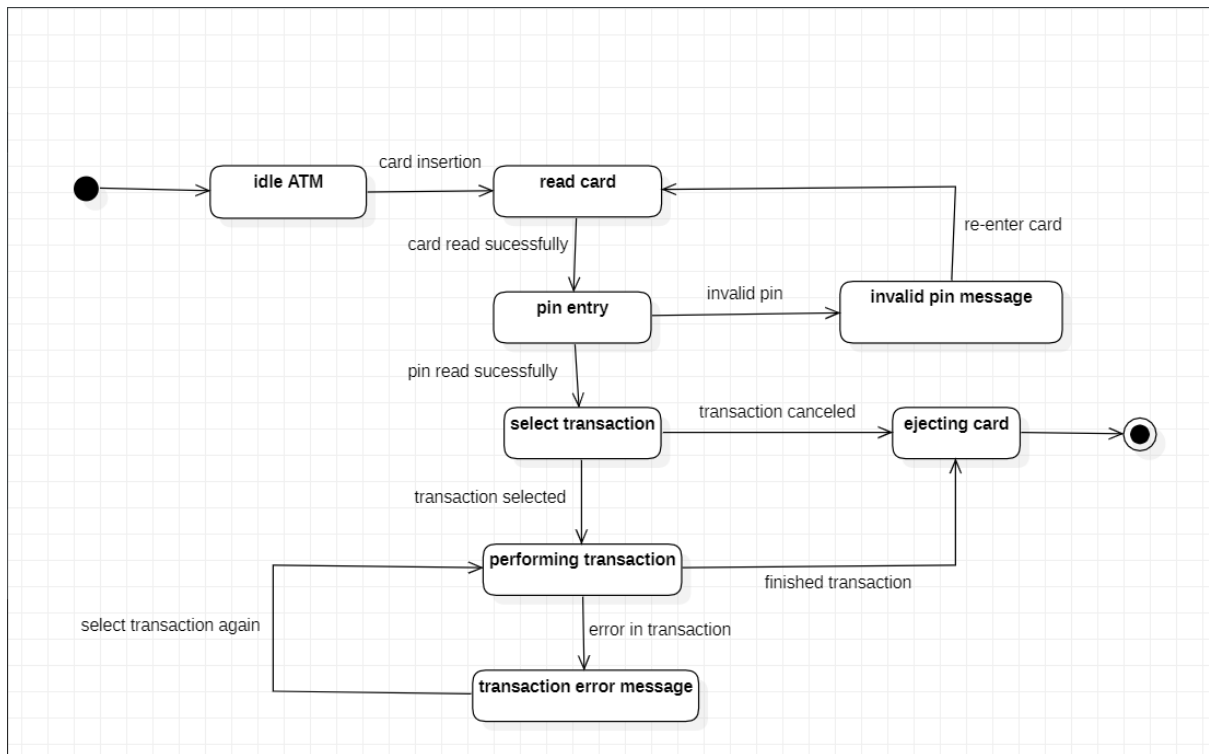
# Assignment-8

## STATE DIAGRAMS

### A) Printer



## B) ATM



### C) Telephone

