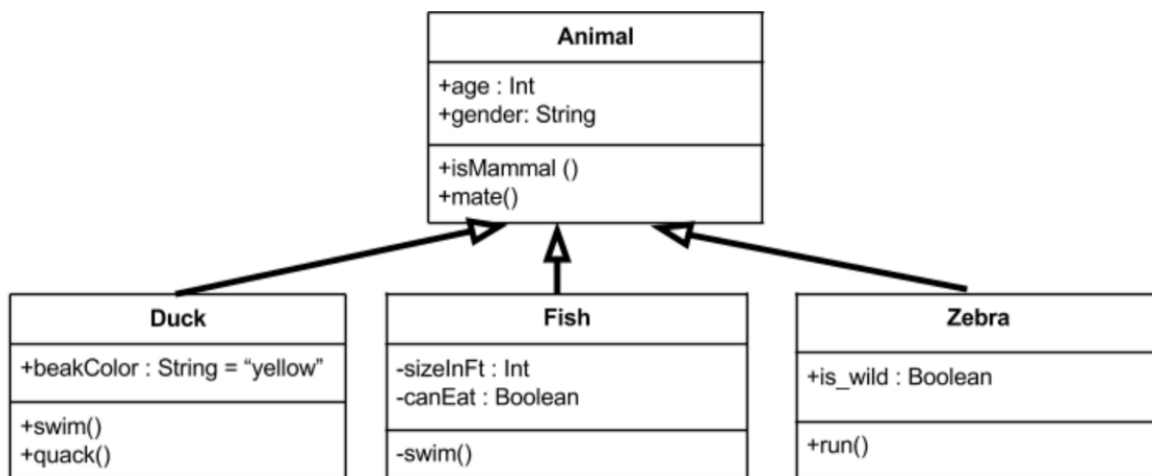
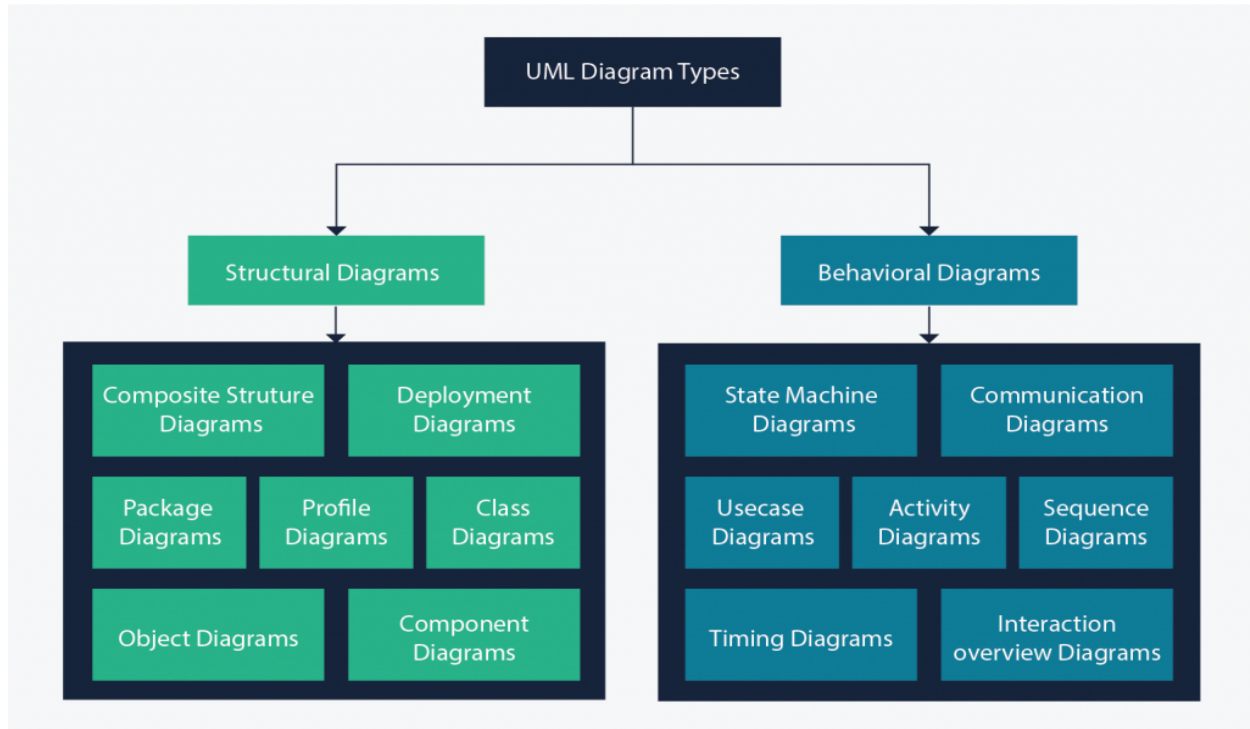

Hello 🙌 my readers in this article I will tell you about UML and its further classifications. After reading this article you guys have a better understanding of UML & its classification.

UML CLASS DIAGRAM.

UML is abbreviated as Unified Modeling Language. And is used to define the visualization of the whole software system. UML class diagram consists of two things A-class and the attributes of that class as you can see in the below figure.



UML is future classified into different types as you can see in the figure below.



STRUCTURAL DIAGRAM.

Structural diagrams are conceptual diagram that is used to document different structure that makes up a system.

E.g

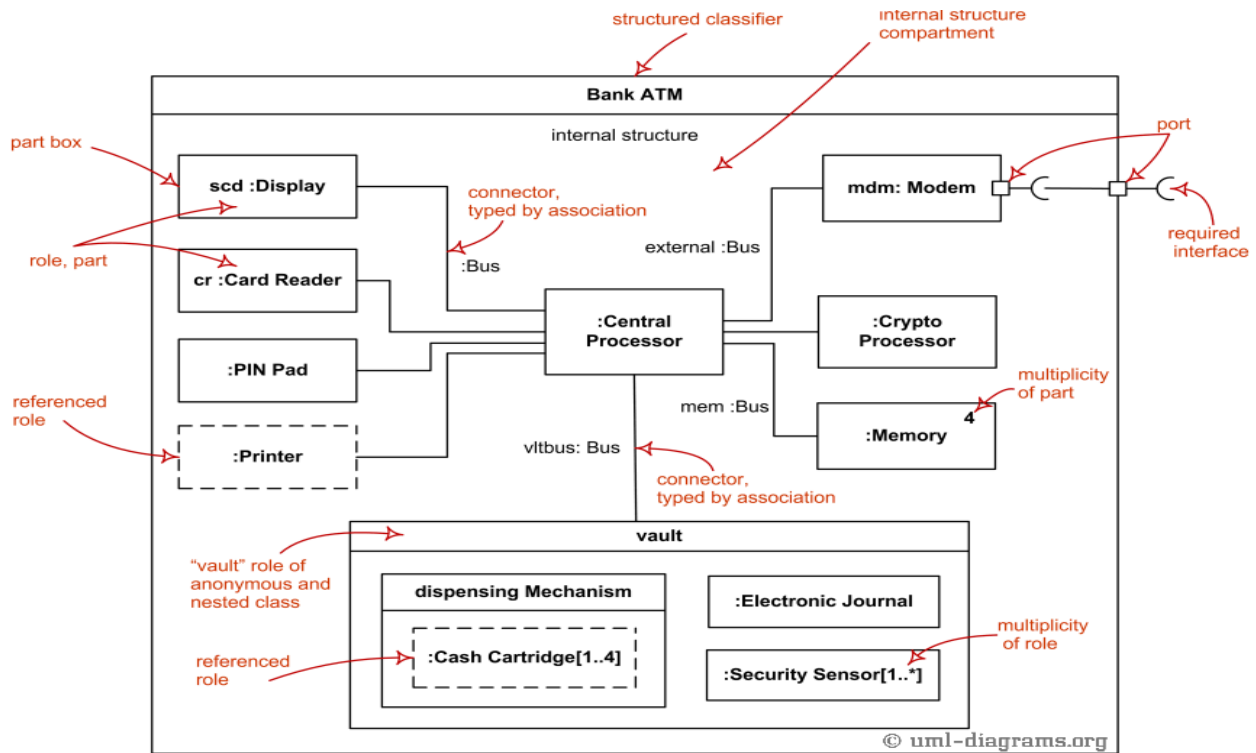
1. Database
2. Applications

Structural diagrams are further classified into the following types.

1. Composite structure diagram
2. Deployment diagram
3. Package diagram
4. Profile diagram
5. Class diagram
6. Object diagram
7. Component diagram

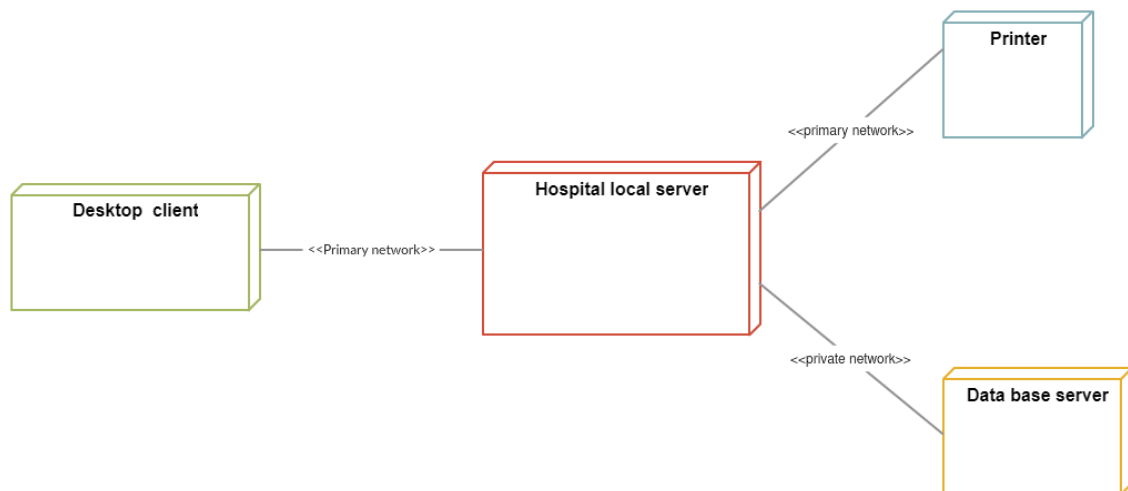
COMPOSITE STRUCTURE DIAGRAM.

It is a type of structure diagram that shows or defines the internal structure of a class diagram with the help of ports and connectors.



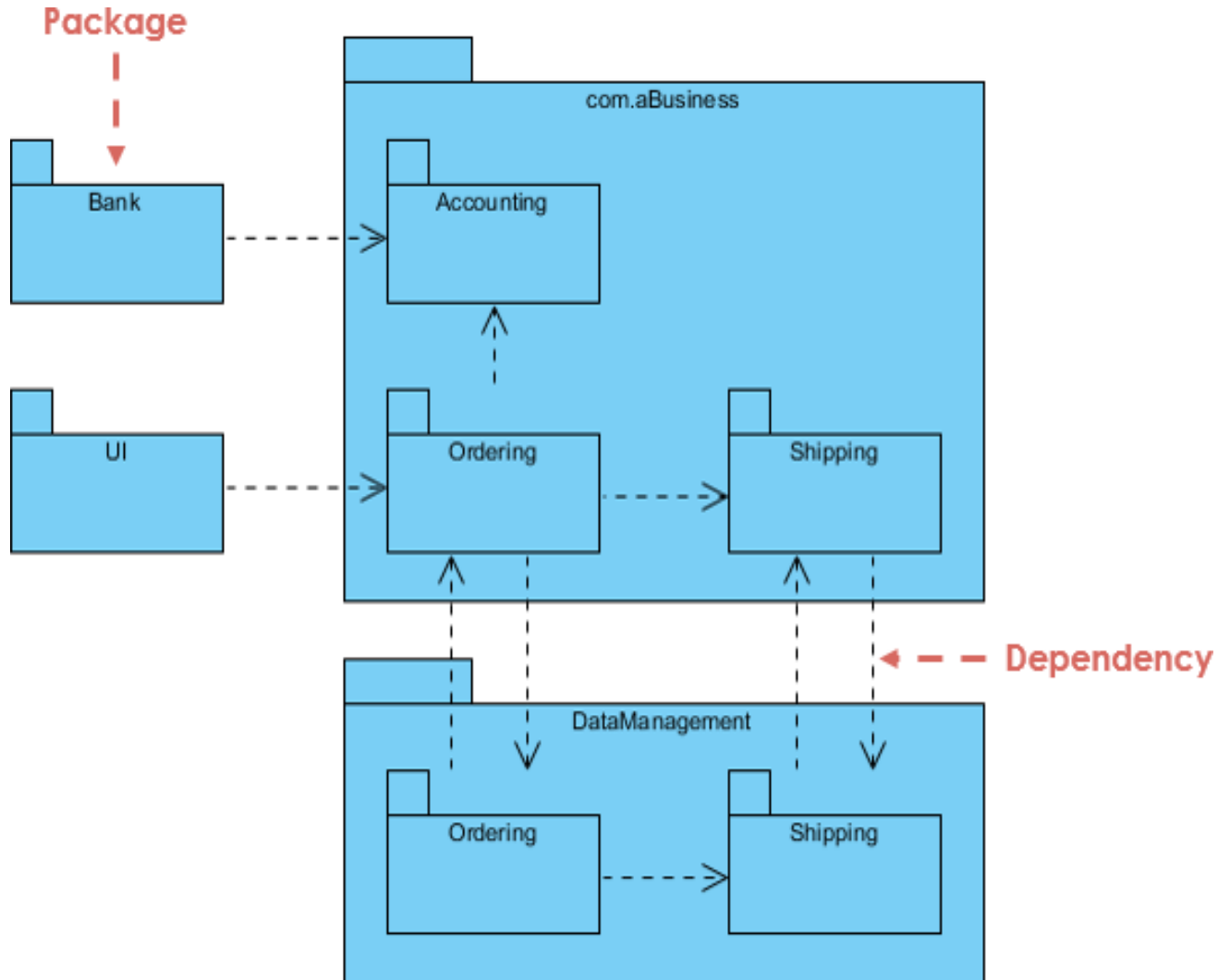
DEPLOYMENT DIAGRAMS

The deployment diagram shows the execution architecture of a system that is how the hardware and software of a system execute also including nodes.



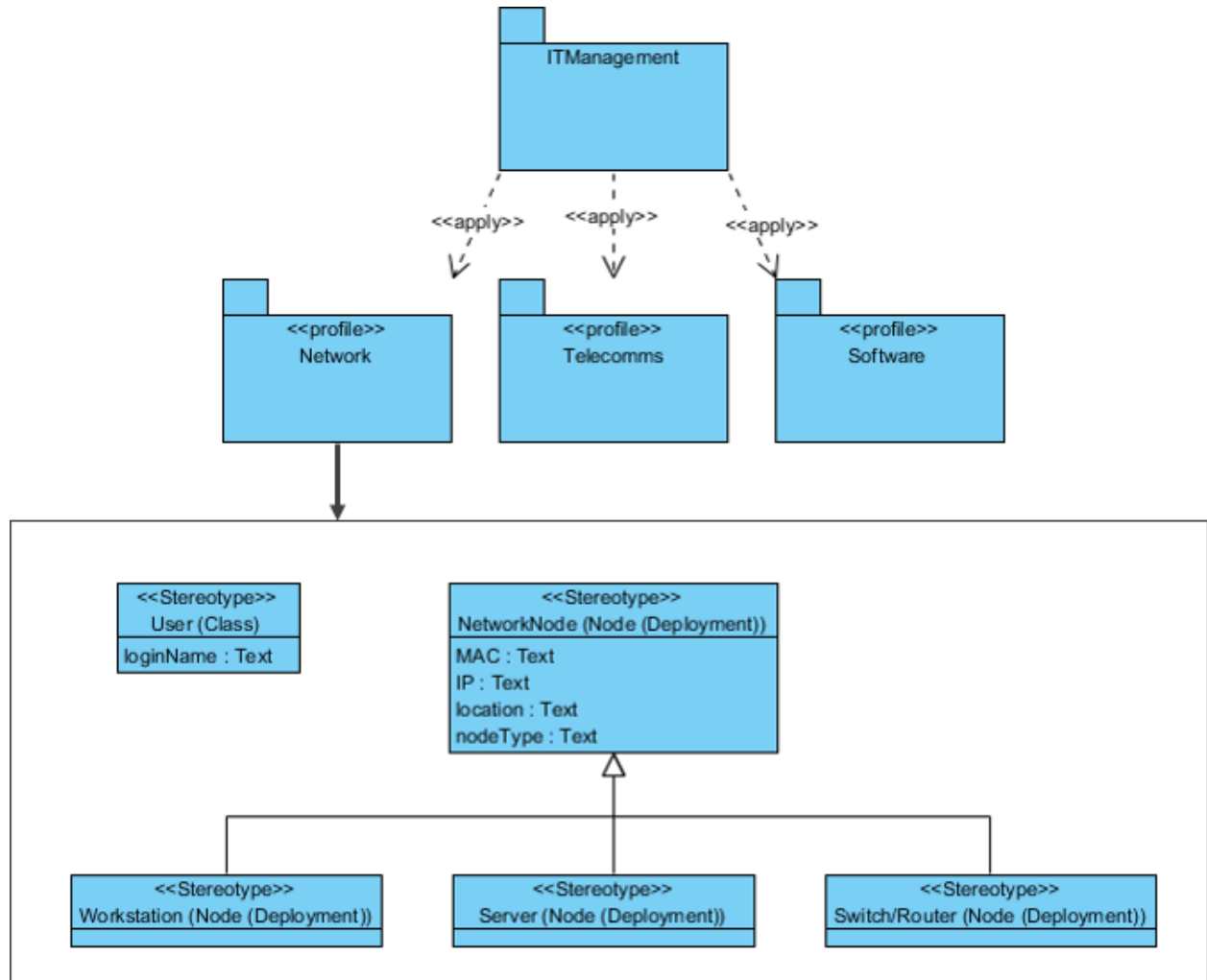
PACKAGE DIAGRAM

A package diagram shows how various models of software are arranged in the form of a package. Also, it shows the structure and dependencies of a subsystem.



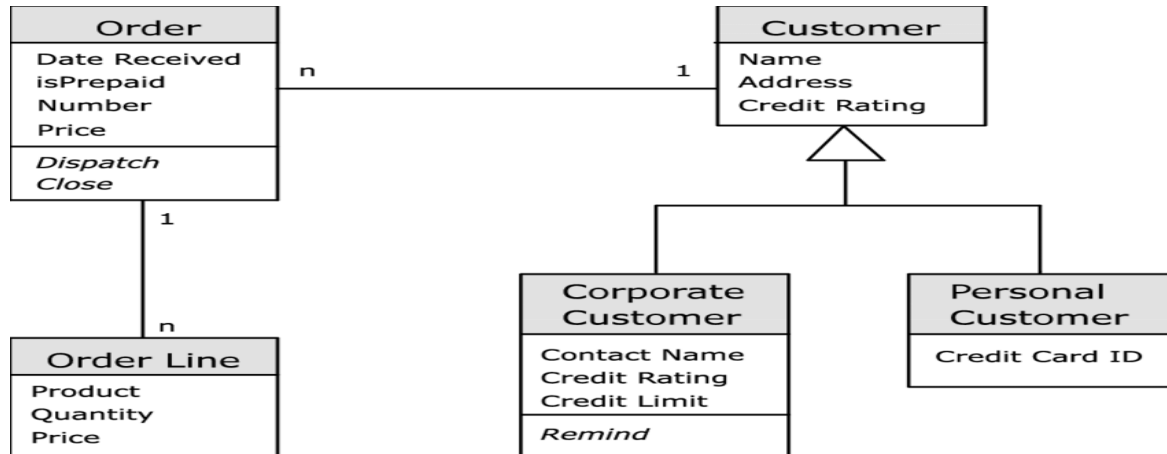
PROFILE DIAGRAM

A profile diagram is a diagram that is created in a profile package and it helps in providing the mean of extending UML.



CLASS DIAGRAM

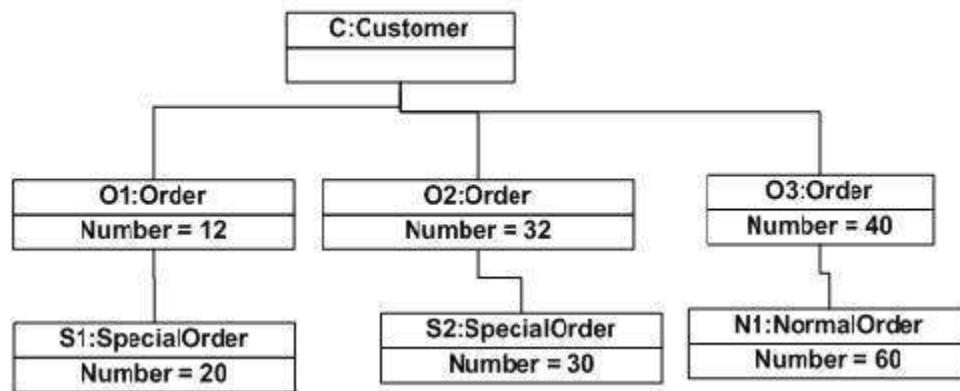
The class diagram is the main building model of OOP. it describes the structure of a system by showing its classes, attributes, and operations and their relationship among objects.



OBJECT DIAGRAM

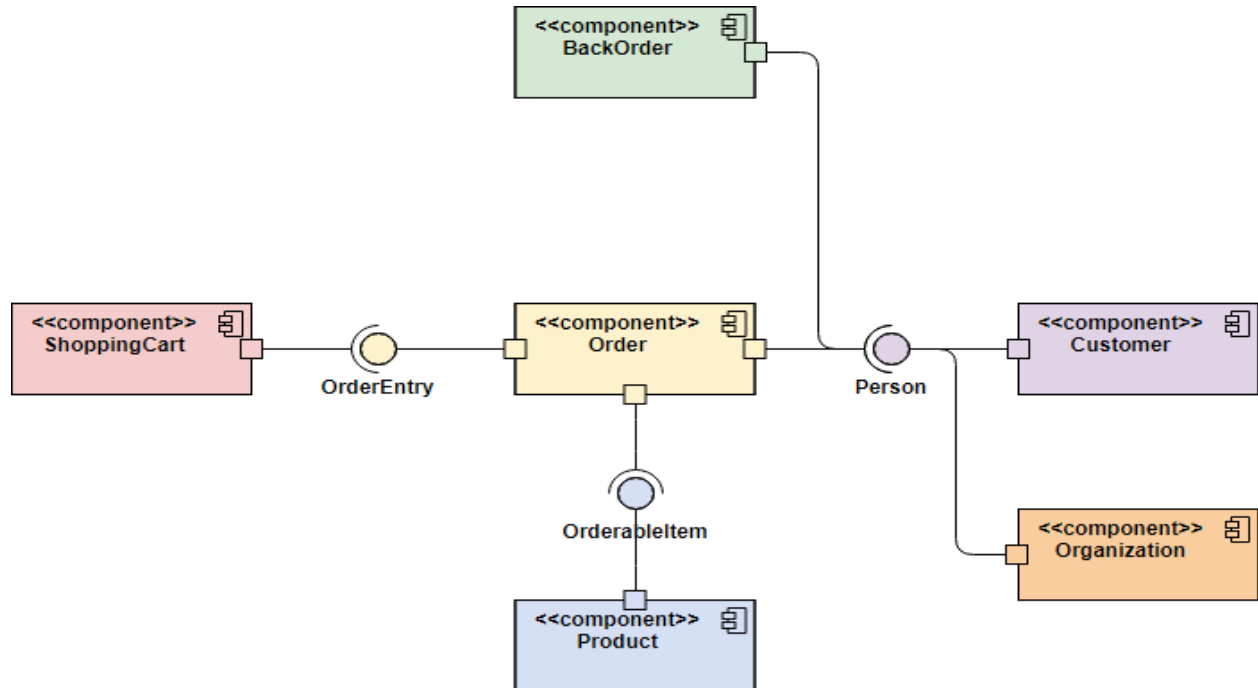
The basic components are the same for the class diagram and object diagram. Object diagrams are used to render a set of objects and their relationships as an instance.

Object diagram of an order management system



COMPONENT DIAGRAM

A component diagram is used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components that are often used to model the static implementation view of a system.



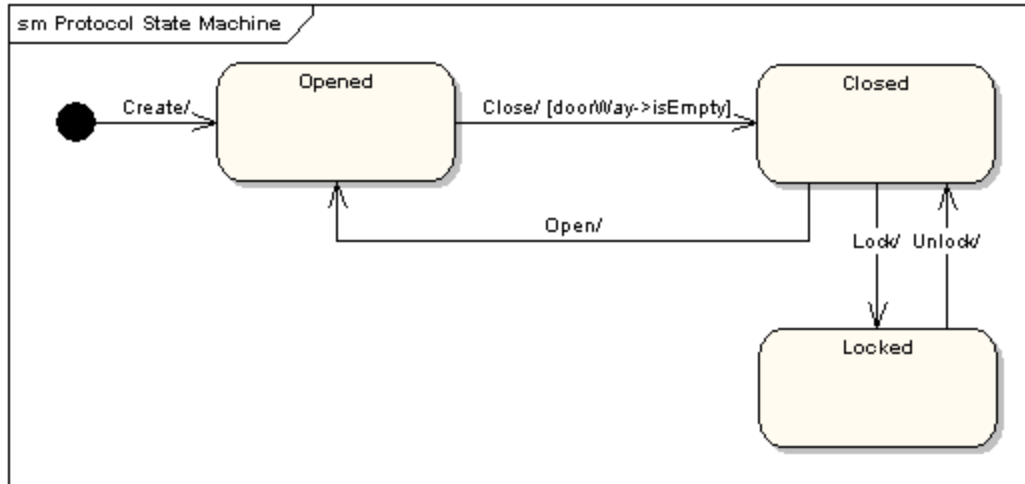
BEHAVIORAL DIAGRAM

Behavioral diagrams are used to provide clarity about the system such as external processes and business processes. Behavioral diagrams are further classified into different types.

1. STATE MACHINE DIAGRAM
2. ACTIVITY DIAGRAM
3. USE CASE DIAGRAM
4. COMMUNICATION DIAGRAM
5. SEQUENCE DIAGRAM
6. INTERACTION OVERVIEW DIAGRAM
7. TIMING DIAGRAM

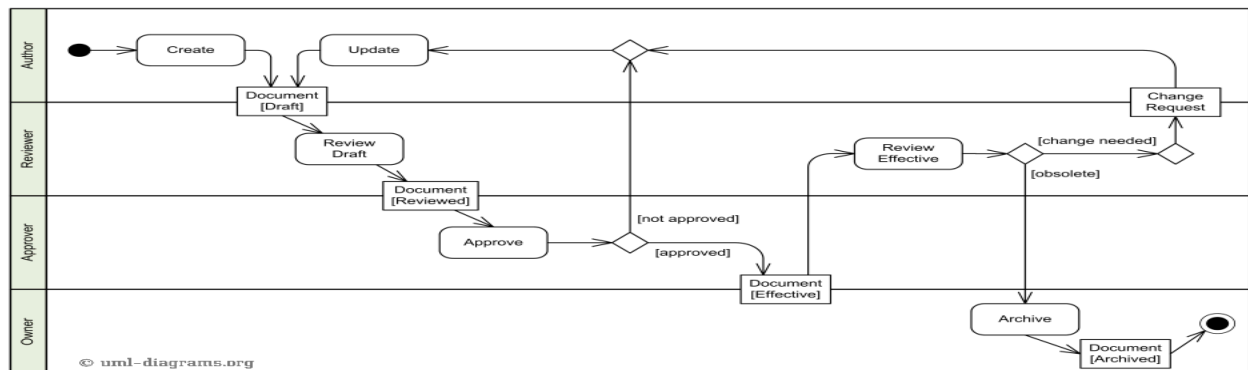
STATE MACHINE DIAGRAM.

State machine diagram is also known as state diagram which is used to show transitions between objects.



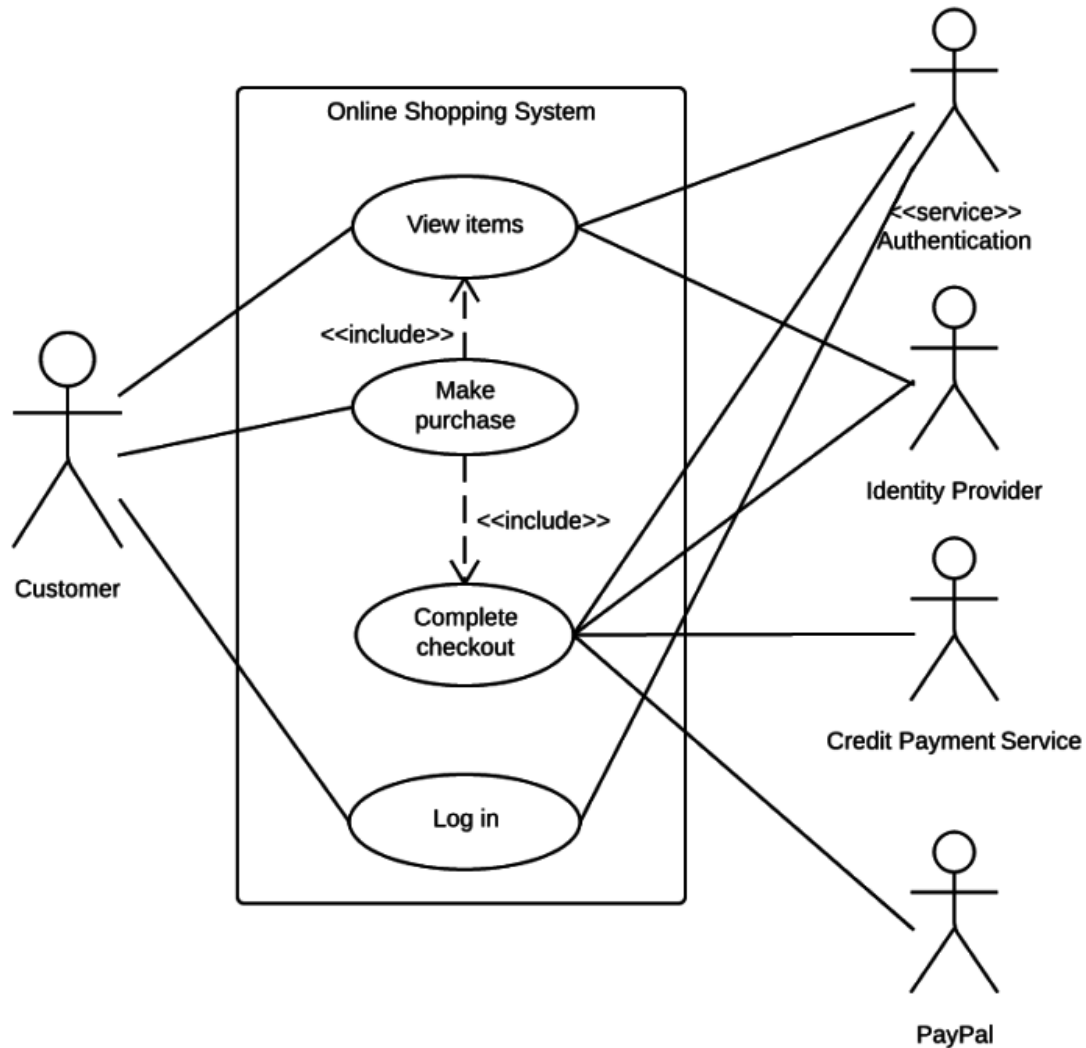
ACTIVITY DIAGRAM

An activity diagram is similar to a flow chart that represents all the action and flow of a system.



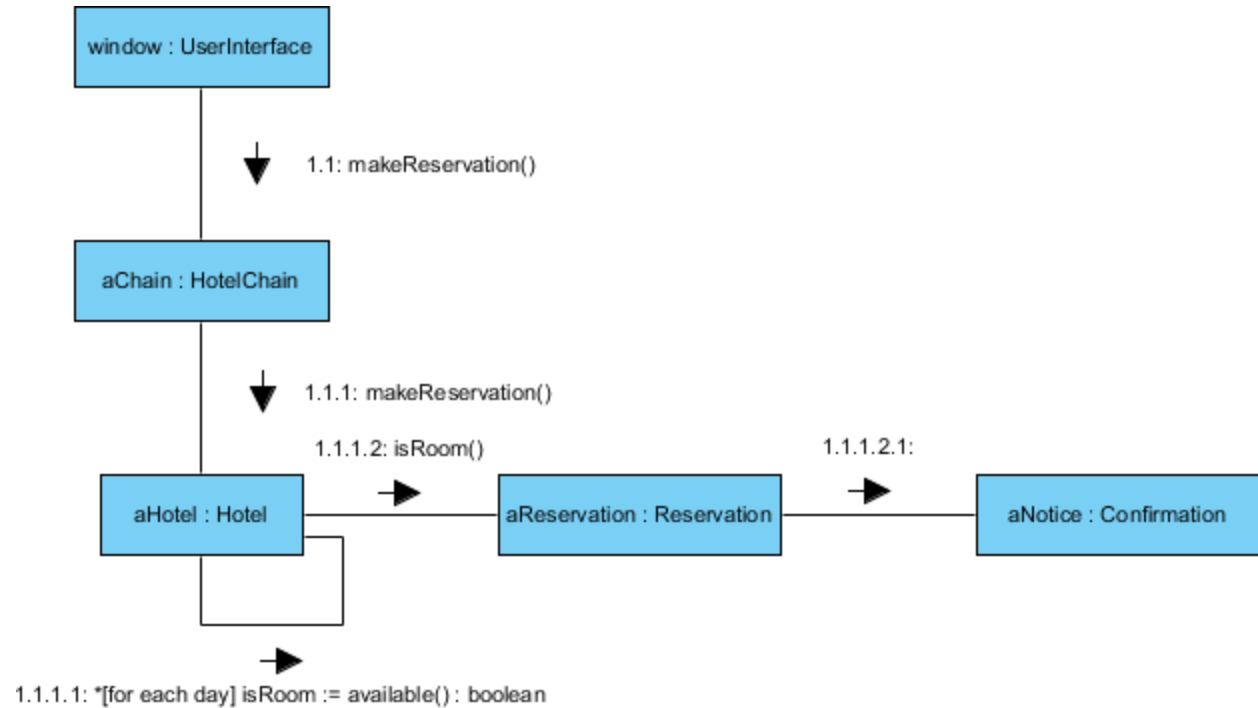
USE CASE DIAGRAM

Use case diagram is a graphical representation that shows how different users will interact with the system to perform their task and which actor has which rights.



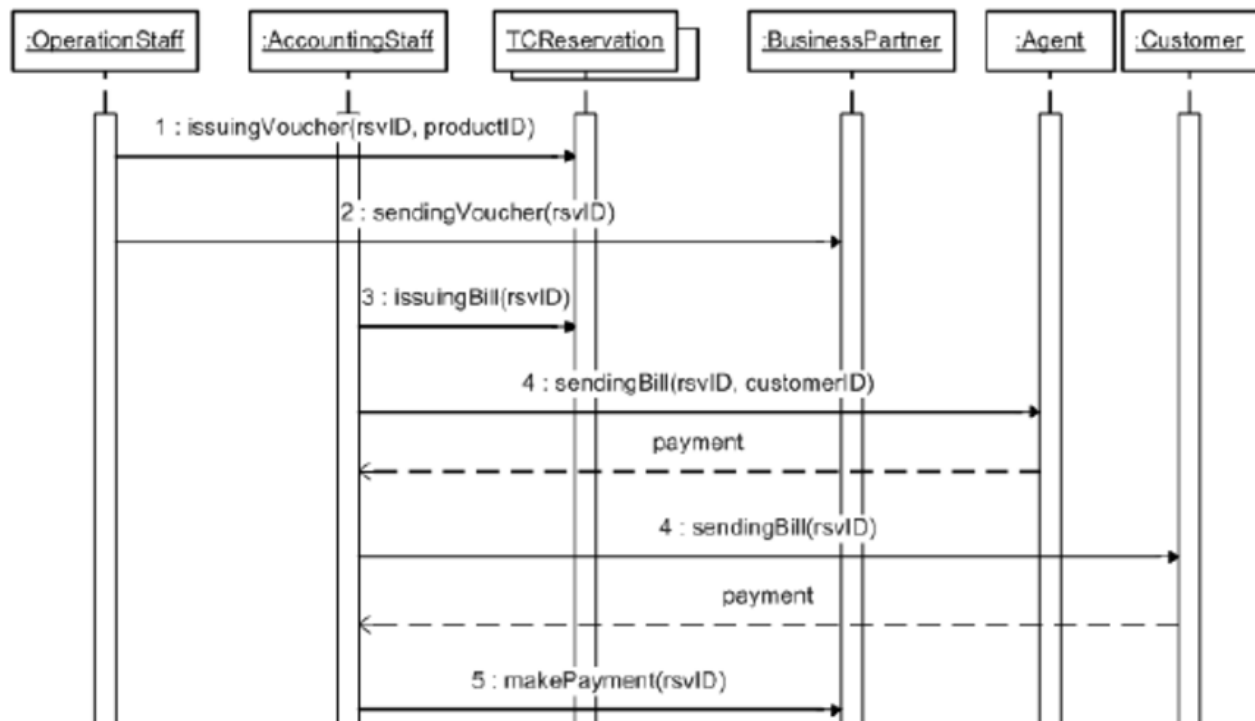
COMMUNICATION DIAGRAM

In the communication diagram, Each participant is represented by a lifeline in the interaction frame. Each action that occurs between the lifelines is represented by a message and the messages are in a specific, numbered sequence.



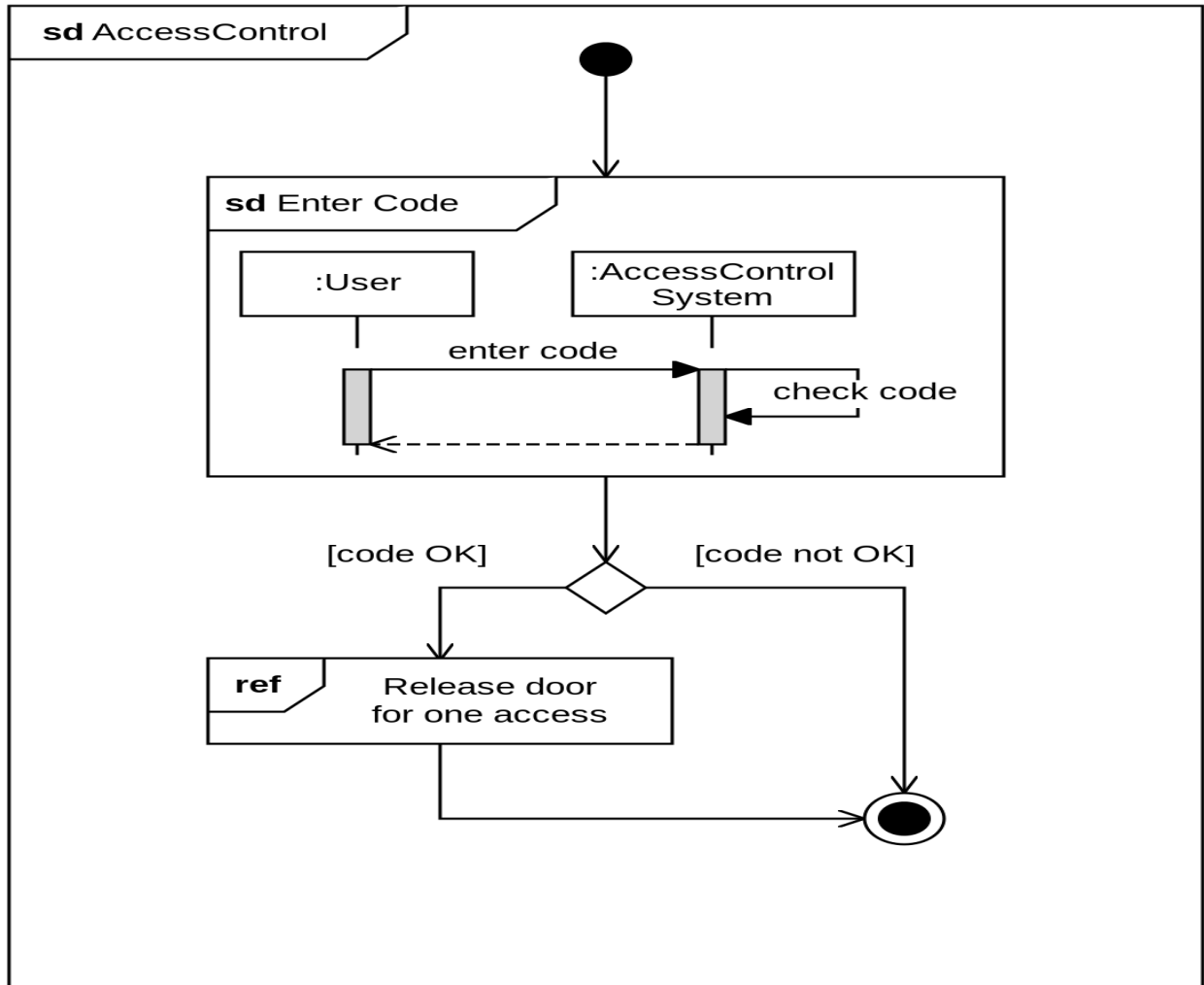
SEQUENCE DIAGRAM

A sequence diagram shows how detailed operations are carried out.



INTERACTION OVERVIEW DIAGRAM

The interaction overview diagram shows the overview of the flow of controls of interactions.



TIMING DIAGRAM

A timing diagram in the Unified Modeling Language 2.0 is a specific type of interaction diagram, where the focus is on timing constraints. Timing diagrams are used to explore the behaviors of objects throughout a given period of time.

