NAME: LALITHA MADHURI PUTCHALA

MY CENTRAL ID: 700657631

EMAIL: [LXP76310@UCMO.EDU](mailto:LXP76310@UCMO.EDU)

CS 4910 Software Engineering Fall’16

CRN 13944 – Section 8:00 AM to 10:30 AM class

Assignment on UML

Q1.

**UML Diagram:**

UML is a standard language for visualizing, specifying, constructing, and documenting the software systems’ artifacts.

**Goal of UML:**

* To define simple and user-friendly general purpose modeling language where all modelers can use.
* These diagrams are not specific for developers but also for common people, business people and anybody who is interested in understanding of the system. Therefore, we can say that UML is not a development method but a process to visualize the model of the system which is either software or non-software.
* My conclusion about the goal of UML can be framed as a simple modeling mechanism to model all tentative practical systems in present multifaceted environs.

The below three elements hold key in mastering the conceptual model of UML:

* Rules to connect the building blocks
* UML building blocks
* Common mechanisms of UML

The below are UML’s building blocks:

* Relationships
* Things
* Diagrams

The different perspectives of the system where UML plays an important role is:

* System Design
* System Implementation
* System Process
* System Deployment

The Use Case view is located at the center connecting these four perspectives. The functionality of the system is represented by a Use case diagram and the use case diagram connects other perspectives.

* The Design of a given system consists of components like interfaces, classes and collaboration. To support this, we can use object diagram and class diagram of the UML.
* Implementation defines the components collected together to make the whole system. UML component diagram is used to support implementation perspective.
* The flow of the system is controlled by the process. Hence, the same elements that are design phase are also used to support this perspective.
* The deployment in UML represents the physical nodes of the system that forms components of the hardware. This is supported by UML diagram.

Distinguishing of the UML models play a major role in selection criteria. The diagrams vary with respect to UML modeling. The main types of UML modelling are below:

**Structural modeling:**

This mainly captures the static features of a given system. It consists of below diagrams:

* Objects diagrams
* Classes diagrams
* Deployment diagrams
* Composite structure diagram
* Component diagram
* Package diagrams

Structural model characterizes the agenda for the system and this agenda is the place where all other components exist. Therefore, the class diagram, deployment diagram and component diagrams are the part of structural modeling. They all represent the mechanism and the elements to accumulate them.

The disadvantage of the structural model is that it never defines the dynamic behavior of the system. Mostly used structural diagram is class diagram.

**Behavioral Modeling:**

Behavioral model represents the structural diagram’s interactions. It describes the interaction happening in the system and its dynamic nature. It usually consists of:

* Interaction diagrams
* Activity diagrams
* Use case diagrams

These diagrams show dynamic sequence flow happening in the system.

**Architectural Modeling:**

Architectural model mainly gives an idea on the complete top-view framework of the system. It is a combination of both behavioral and structural elements of the system. In simple words, this model can be said as the blue print of the whole system. Example of this type of modelling is Package diagram.

**Different Types of UML Diagrams:**

Initially categorization of the system is into two broad categories and later sub-divided into more categories:

* Behavioral Diagrams
* Structural Diagrams

**Structural Diagrams:**

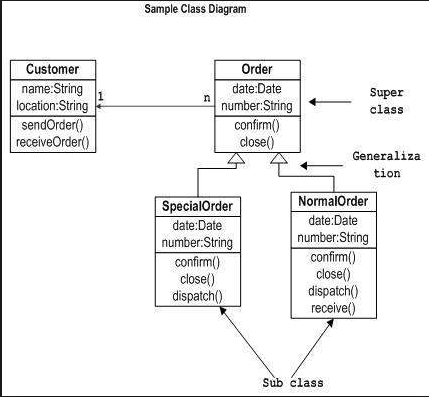
The structural diagrams mainly provides an idea on the static feature of the system. The system is stable because the static characteristics represent those parts of a diagram that forms the main structure.

Classes, objects, interfaces, nodes and components represent these static parts. The four structural diagrams are:

* Object diagram
* Class diagram
* Deployment diagram
* Component diagram

**1)Class Diagram:**

The most common diagrams used are class diagrams in UML. Class diagram consists of classes, associations, interfaces and collaboration. Class diagrams are static in nature and generally represent the object oriented view of a system. To represent the concurrency of the system in a UML, an active class is used for this purpose. Class diagram is normally used for development purpose as it represents the object orientation of a system. Also, at the time of system construction, this comes in to pretty handy.



**2)Object Diagram:**

These have more resemblance to real-life scenarios during system implementation. They can be described as an class diagram’s instance. Represent static view of the system. Like class diagrams, these diagrams are set of objects and their relationships. These diagrams usage is pretty similar to object diagrams whereas these diagrams in particular helps in building the prototype of the system.



**3)Component Diagram:**

These diagrams contains interfaces, classes or collaborations that represent a set of components and their mutual relationships. Hence, they give an idea on system implementation. The system’s software artifacts are arranged in different groups depending upon their relationship which are known as components. These are helpful to visualize the implementation.

**4)Deployment Diagram:**

These are a set of nodes and their relationships. Nodes can be physical entities in the place where components deployment happens. These are mainly used to deployment visualization of the system which are normally used by deployment teams.

**Behavioral Diagrams:**

To quote a system design is complete, we have to consider both static and dynamic aspects. Till now, the above diagram considered static aspects of the system. The dynamic aspect of the system is captured by the behavioral diagrams. This aspect can be defined further as the moving or changing parts of a system.

These five types of behavioral diagrams are more specific classifications of UML:

* Sequence diagram
* Use case diagram
* State chart diagram
* Collaboration diagram
* Activity diagram

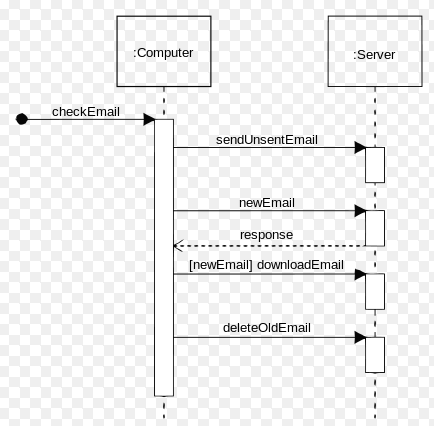
**1) Use case Diagram:**

Use case diagrams represent systems view which consist of a set of actors and their relationships. Each functionality of a system is represented by a use case. This is dependent up on controllers called actors whose internal and external relationships are described by the use case diagram.



**Sequence Diagram:**

A sequence diagram basically said to be an interaction diagram. It mainly deals with some sequences of messages passing between objects as we get from its name. Components interactions is key when seen from execution and implementation perspective. Therefore, this kind of diagrams are used for visualization of calls in system that is performing a specific functionality.

s

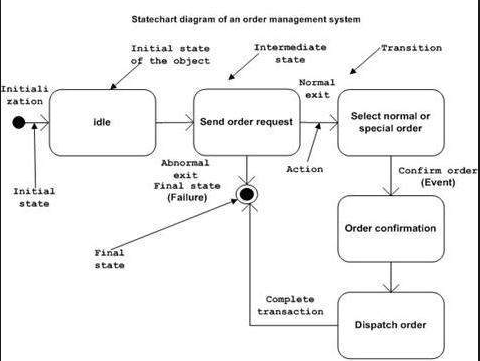
**Collaboration Diagram:**

This represents the hierarchical structure of the system or an organization along with the messages received or sent. The links and objects constitutes structural organization. It has similar purpose of sequence diagram.

In particular, the collaboration diagram is visualized to view the organization structure and their interaction.

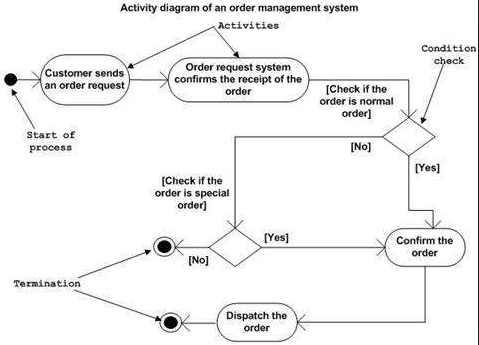
**State-chart Diagram:**

This represents the real time system interactions particularly internal or external events reactions which are responsible for changes of the state of a system. It represents the event driven changes of the state of a system. In specific, the state changes of interfaces and classes are represented. It visualizes the system reactions of system caused by external or internal factors.



**Activity Diagram:**

It gives an idea on the control flow of the system. Here the flow may be concurrent, sequential or branched which consists of links and activities. The functions are said to be activities here. These activity diagrams may be in a group are used to cover the entire system’s flow. It mainly visualizes the control flows in system. This gives an idea how system functions will work during execution.



Q2. CLASS DIAGRAM IS TAKEN FOR EASE OF MODELLING.

**AIRPORT MODELLING USING UML:**

**Boarding Employee**

**Verify List of Passengers**

**Verify Boarding Pass**

**Generate List of Passengers Not Yet in Board**

**Generate List of Passengers On Board**

**Employee:**

**Employee ID, Name and Designation**

**Customs at Destination Airport**

**Checking Customs**

**Verifying Passengers on Board**

**List of Passengers on-board**

**List of Passengers NOT YET on-board**

**Passenger List**

**Baggage Transportation:**

**Handling Agent ID**

**Verifying list of passengers**

**Lodging Luggage**

**List of Checked in Passengers**

**Luggage:**

**Identification**

**Weight**

**Id**

**Boarding Pass Details:**

**Seat No**

**Flight No**

**Gate No**

**Date**

**Depart time**

**Boarding time**

**From**

**To**

**Check in Employee:**

**Verifying Ticket**

**Checking Passenger**

**Ticket Details:**

**1) Ticket Number**

**2) Flight Number**