Roll No: 67
NAME: HERAMB. R. PAWAR
SUB: SEPM
ASSIGNMENT:

ASSIGNMENT

# What is UML DESIGN?

UML stands for Unified Modelling Language. It is a graphical Notation used to represent and communicate Softwae design. It Consist of various diagrams that can be used to model the different aspects of Softwae System, including its Structure behaviour and interactions

UML Diagram are standardize and widely used in Softwae Industry to help developers, designers and Stakeholder understand and communicate about Softwae designs

# Importance of UML:

They are useful in Agile Development environment they help to keep development productive and Focused. UML teams Can help engineering teams:

a) Bring new team member or developers Switching team up to speed quickly

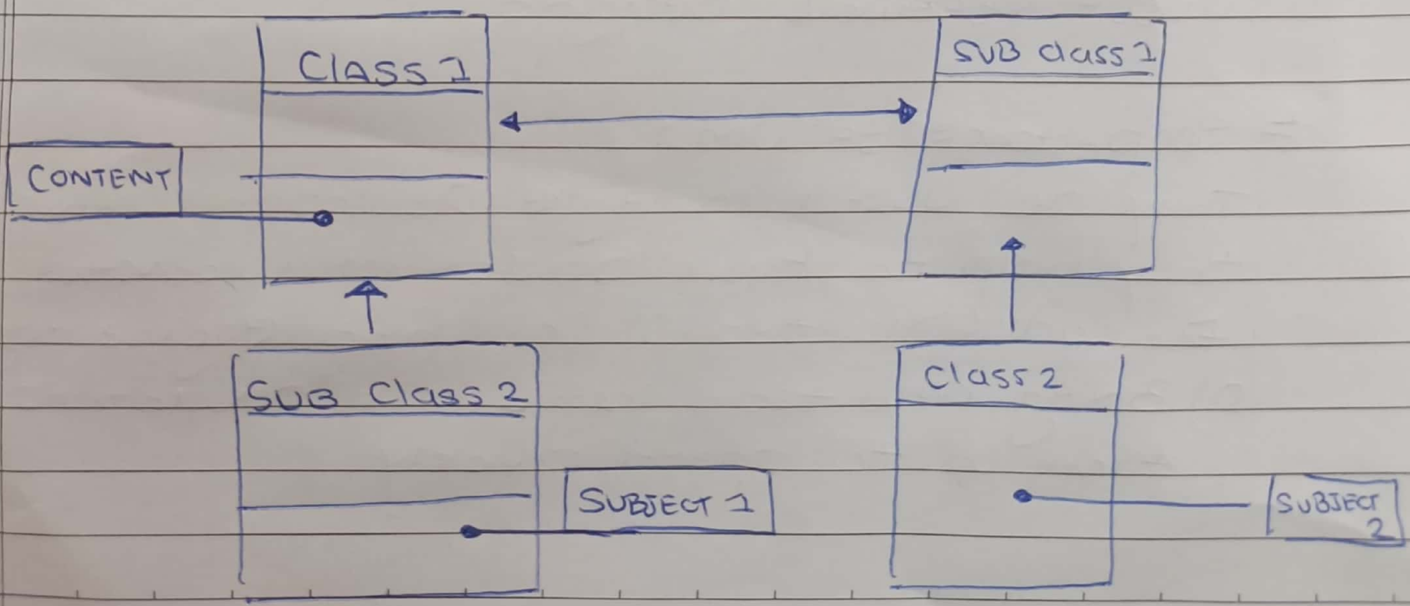
B) Navigate Source code
c) Plan out new feature before any programming takes place
D) Communicate with technical and non-technical audiences more easily.

NOTE: DIAGRAMS That Don't evolve with a project are useless So it's necessary to have Constantly evolving Diagram.

# Types OF UML DAIGRAM:

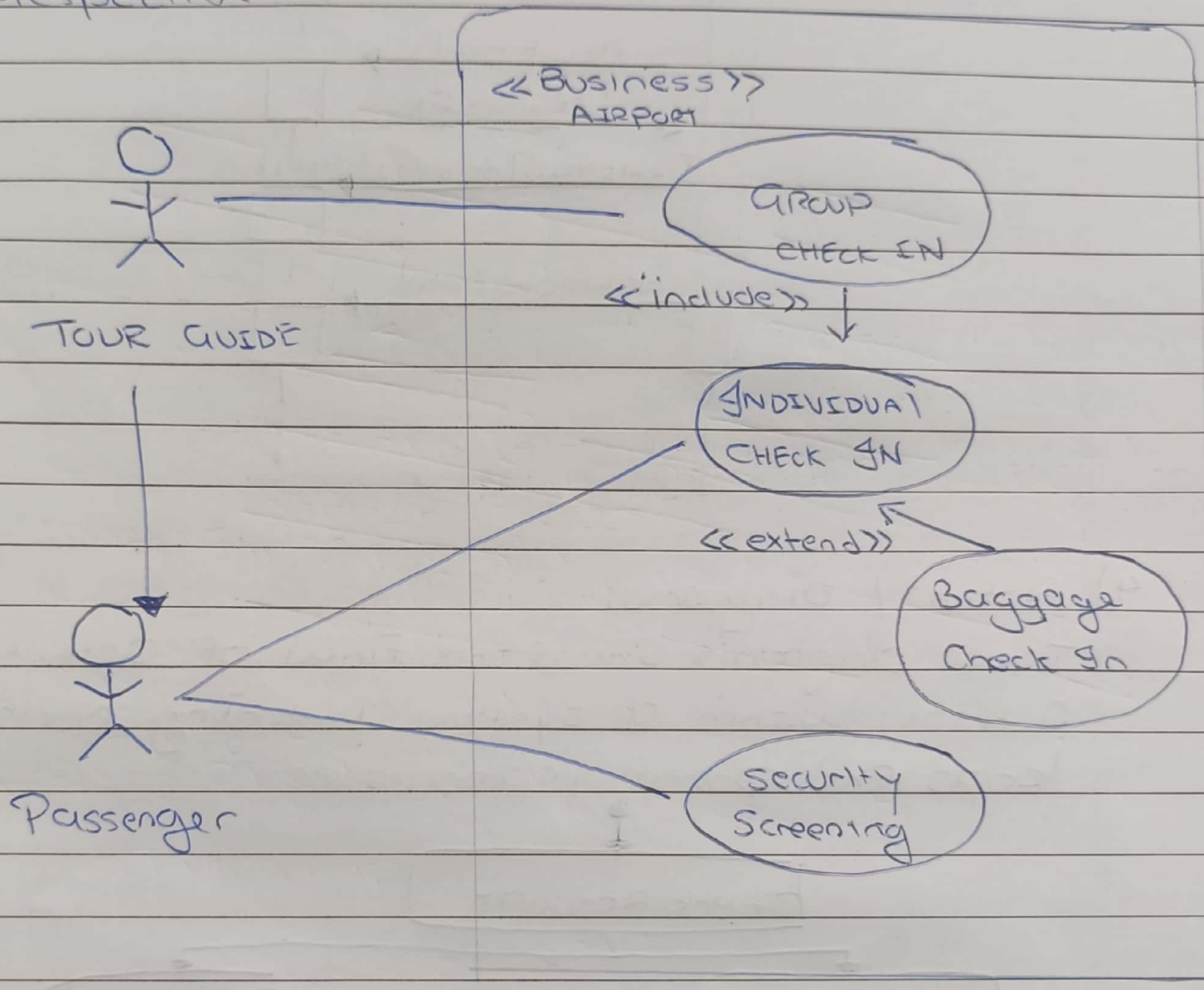
There are Several type OF UML DIAGRAM. EACH WITH A SPECIFIC Purpose and Focus:

1) CLASS DIAGRAMS:
THESE DIAGRAMS Represent the Static Structure OF a System, Including Classes, Interface and their relationship



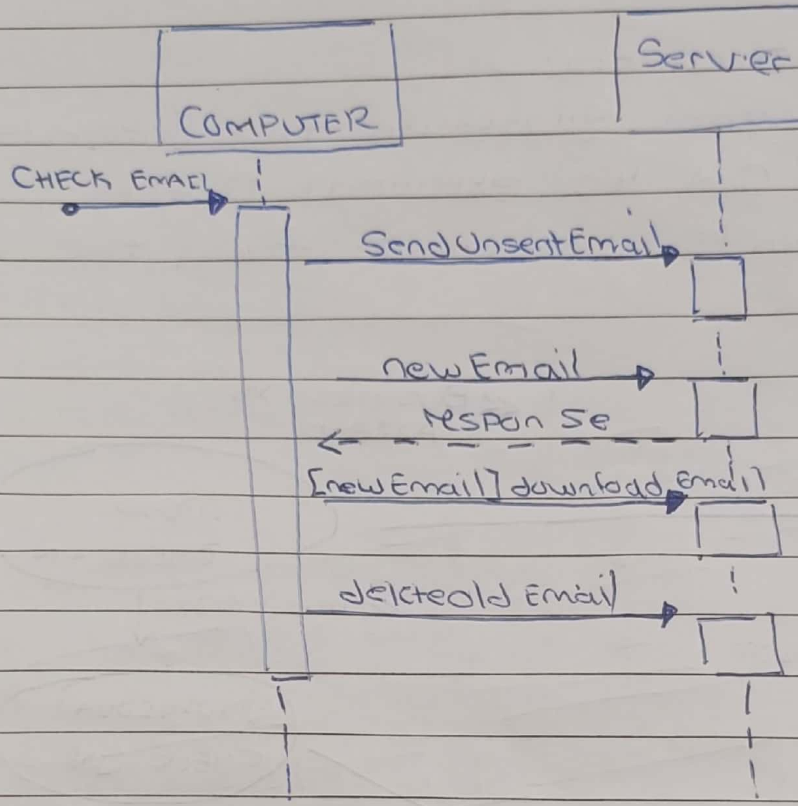
## 2) USE CASE DIAGRAM:

These DIAGRAM Illustrate the Interaction between a System and its external Actors, Showing the System's behaviour From the User's prespective



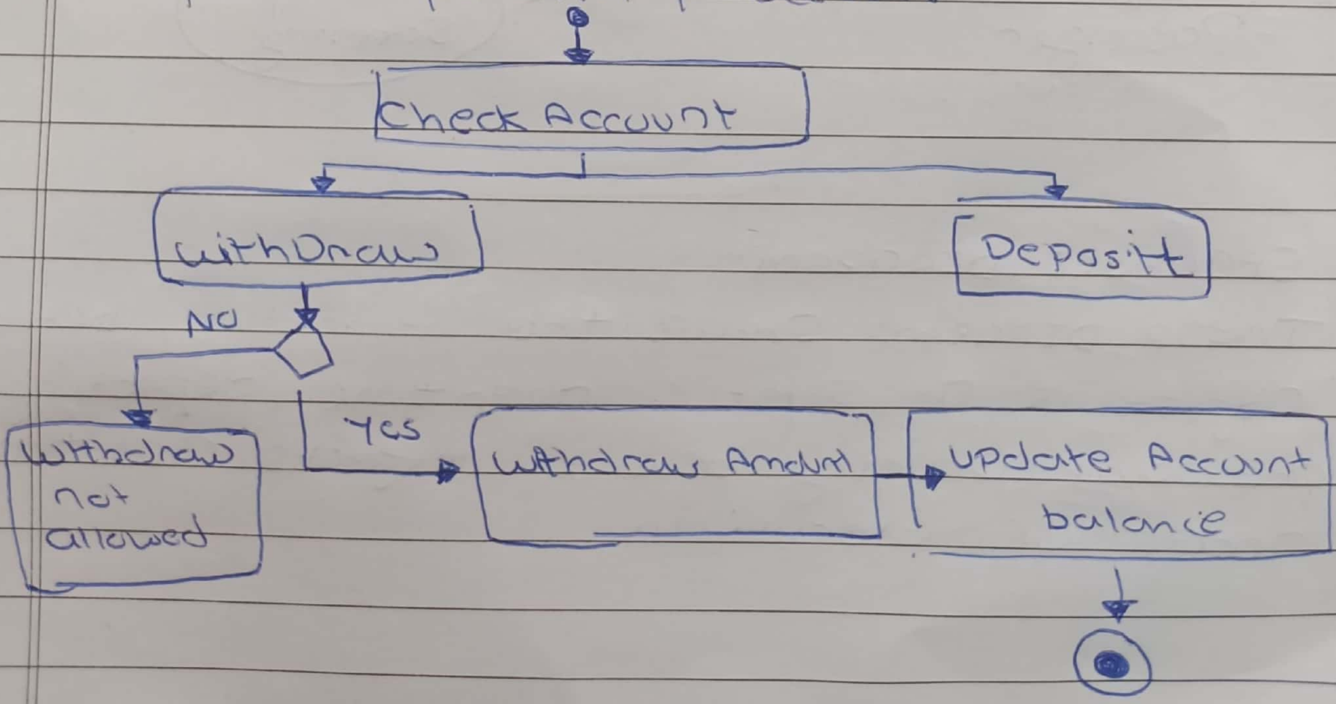
## 3) SEQUENCE DIAGRAM:

These DIAGRAM Shows the interaction between Object in a System over time, depicting the message Passing between objects and their order of execution



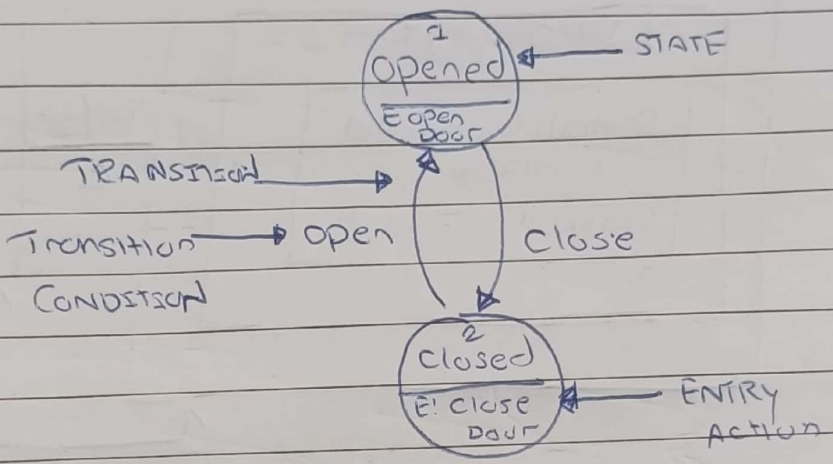
4) Activity Diagrams:

These Diagrams Show the Flow of activities and actions Within a system including decisions, loops and parallel processes.



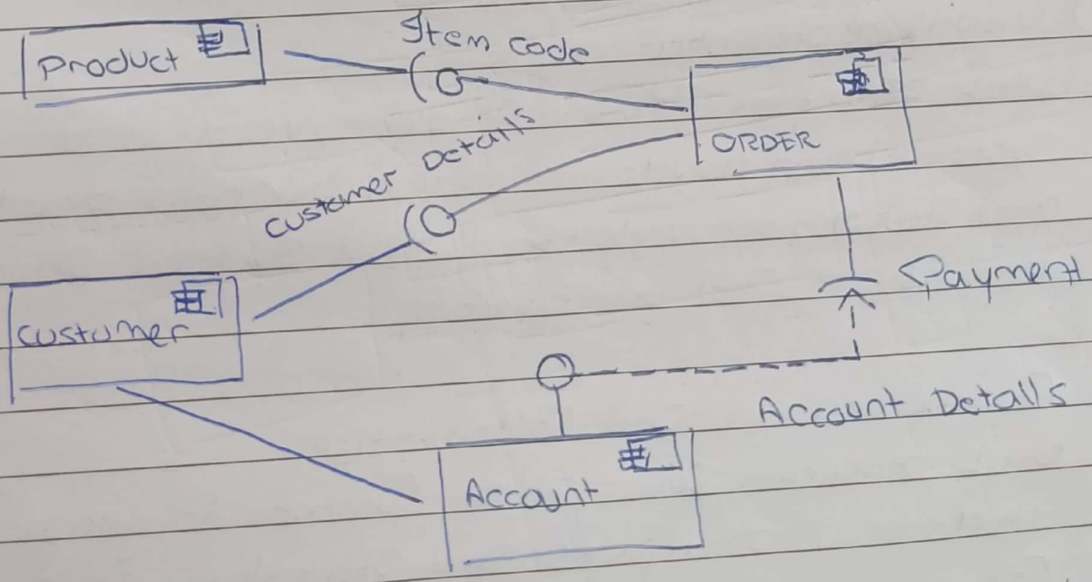
STATE DIAGRAMS:

These DIAGRAM Represents the States and transition of a object in a system, Showing how object respond to event and change over time.



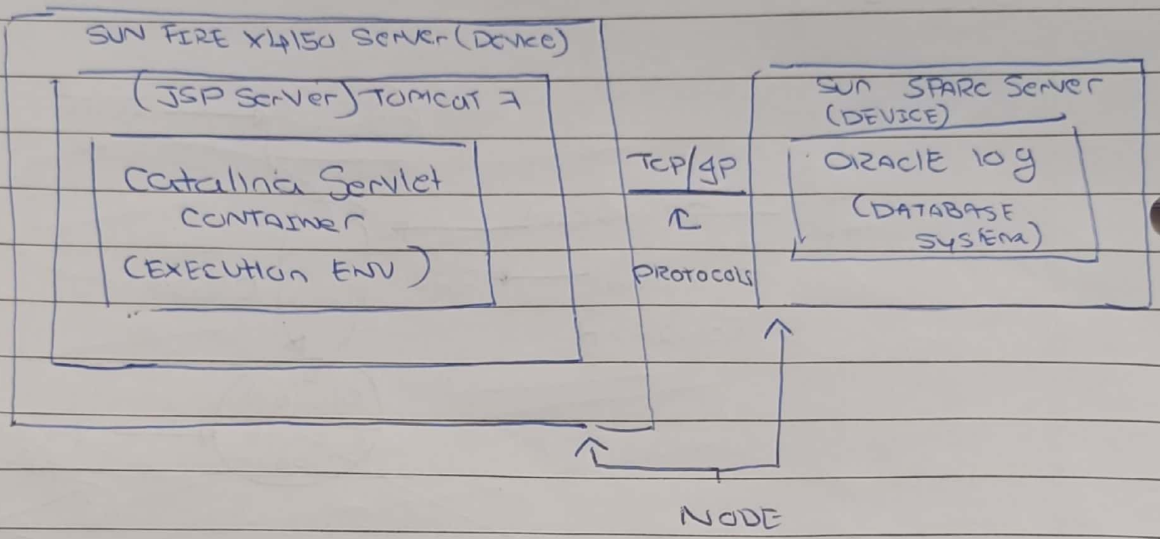
Component Diagrams:

These Diagrams Show the component of ~~the~~ a System and ther relationship including the dependencies between different Components



7) Deployment Diagram:

These Daigram Shows the Physical Deployment of Software Components or hardware devices, Including Servers, routers and other Infrastructer elements


SUN FIRE X4150 Server (Device)
(JSP Server) TOMCAT 7
Catalina Servlet CONTAINER (EXECUtion ENV)
TCP/IP
PROTOCOLS
SUN SPARC Server (DEVICE)
ORACLE 10g (DATABASE SYSTEM)
NODE

# USES OF UML DIAGRAM: (ADVANTAGES)

The Use of UML DIAGRAM in Software Development offers Several benefit's including:

a) Visual representation:

UML DIAGRAM Provide a clear, visual representation of System being developed which can help developers and Stakeholder understand the System Structure, behaviour and interaction

B) Communication and Collabration:

UML DIAGRAM Serve as a Common language that developers, designers and Stakeholders can use to Communicate and Collabrate throughout Software.

development process This help to ensure that everyone is on the same page and working toward same goal.

3) Analysis and design:
UML Diagrams Can be used to analyze and design Software System before implementation, helping to Identify potential problems or design flaws early in development process. This can save time and money by preventing issues from arising later on.

4) Documentation:
UML Diagrams Can be used to document the design of Software System, providing a clear and comprehensive reference for Developers, designers and Stakeholders

5) Code Generation:
Some UML tools can automatically generate code for UML Diagrams which Can Save time and reduce errors in Coding process

# DISADVANTAGES:
1) Complexity:
UML Diagram can be complex and Challenging to understand, especially for non-technical stakeholder who may not have a background in Software development

2) Time Consuming:
Creating and maintaining UML Diagrams Can be time consuming especially for large and Complex Software Systems

3) Learning Curve:
learning how to Create and interpret UML Diagram Can require significant Investment of time and effort

4) Lack of Standardization:
There is no Single Standard for UML DIAGRAM which can lead to inconsistencies and confusion if team members have different interpretation for same Diagram.

5) Too much Detail:
It is possible to create UML Diagram that are overly complex and detailed making them difficult to understand and use effectively.

# Tools Used for Making UML DIAGrams:

a) Lucidchart
b) Visual Paradigm
c) Microsoft Visio
D) Gliffy

e) Astah
f) Enterprise Architect
g) Planet UML