Name of student: Abhay OmPrakash Prajapati		
Roll no: 41		**Tutorial No: 4**
Title of LAB Assignment: UML Diagrams (Use Case Diagrams)		
DOP: 25-09-2023		**DOS:02-10-2023**
CO Mapped: CO1,CO2,CO3	**PO Mapped:** PO1 PO4 PO6	**Signature:**

Aim: UML Diagrams (Use Case Diagrams)

Description:

The user scenario provides a depiction of how the system's features are perceived by external users, known as actors. A use case represents a cohesive unit of functionality expressed as interactions between actors and the system. Its purpose is to outline the actors, use cases, and their associations.

Understanding Dynamic System Behavior

A use case diagram serves as a visual representation of a system's dynamic behavior

Incorporating System Functionality

It encapsulates the functionality of the system by integrating use cases, actors, and their relationships.

Modeling System Tasks and Services

This modeling approach defines the tasks, services, and functions required by a system, particularly within a subsystem of an application.

Illustrating High-Level System Functionality

A use case diagram provides an overview of the system's high-level functionality and demonstrates how users interact with the system.

How to Draw a Use Case Diagram?

A use case diagram is a visual representation of the interactions between a system and its users. It is a type of Unified Modeling Language (UML) diagram that is used to capture and document the functional requirements of a system.

Use Case Diagram Symbols and Notation

The following symbols and notation are used in use case diagrams:

Actors: Actors are external entities that interact with the system. They can be people, organizations, or other systems. Actors are represented by stick figures.

Use cases: Use cases are units of functionality that the system provides to its users. They are represented by ovals.

Relationships: Relationships show the interactions between actors and use cases. There are three types of relationships:

Association: An association shows that an actor is involved in a use case. It is represented by a solid line.

Include: An include relationship shows that one use case includes the functionality of another use case. It is represented by a dashed line with an arrow pointing to the included use case.

Extend: An extended relationship shows that one use case extends the functionality of another use case. It is represented by a dashed line with an arrow pointing to the extended use case.

Example Use Case Diagram

The following image shows a simple use case diagram for an online banking system:

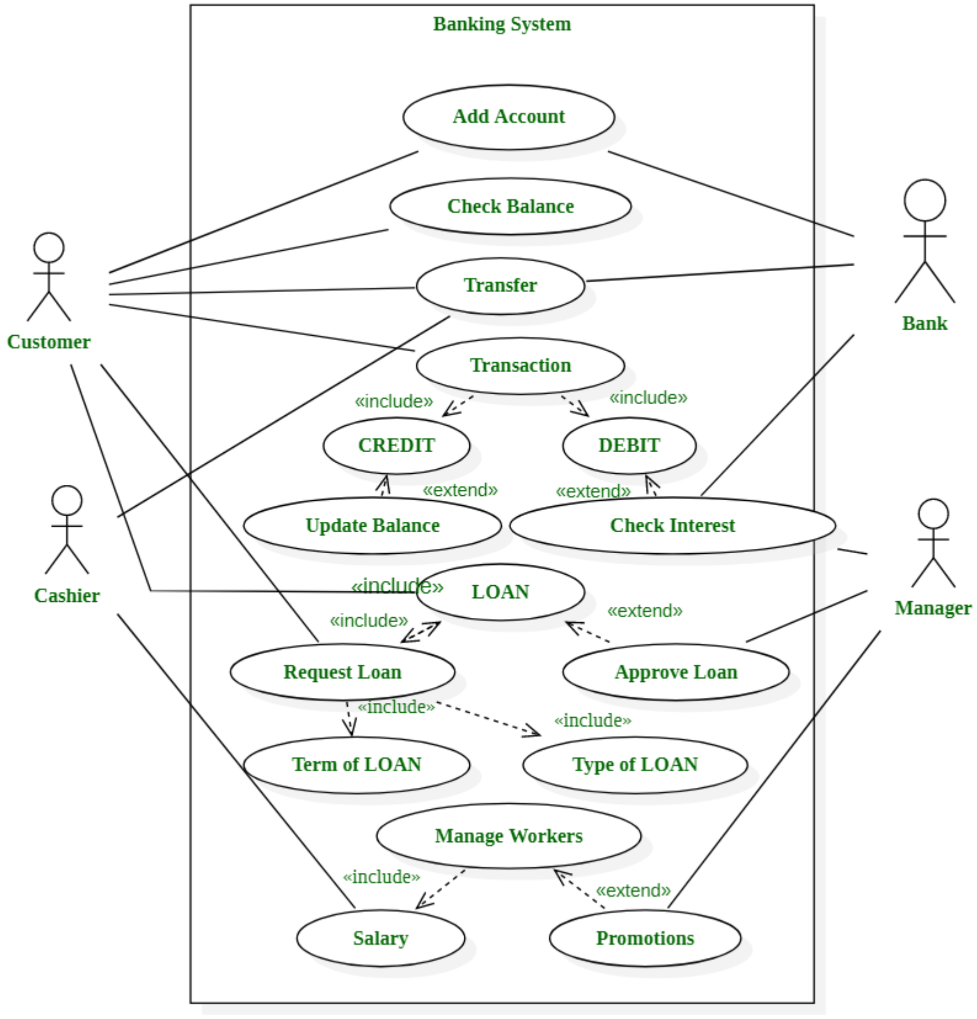
The actors in this diagram are customers and bank employees. The use cases are:

- **Customer:**
  - WithDrawCash
  - Deposit cash
  - Transfer money
  - View account balance
- Bank employee:
  - Open a new account
  - Close an account
  - Issue a credit card
  - Process a loan application

The relationships in this diagram show that:

Customers can withdraw cash, deposit cash, transfer money, and view their account balance.

Bank employees can open new accounts, close accounts, issue credit cards, and process loan applications.



Conclusion:

Use case diagrams are a valuable tool for system analysis and design. They can help you to understand the system's functionality from the user's perspective and to identify and prioritize the system's features.