SOFTWARE ENGINEERING PRACTICAL

PRACTICAL NO: 2

AIM: Study and implementation of Use case diagrams.

SOLUTION

A use case diagram is used to represent the dynamic behaviour of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

In software and systems engineering, a use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modelling Language as an actor) and a system, to achieve a goal. The actor can be a human, an external system, or time.

Another way to look at it is a use case describes a way in which a real-world actor interacts with the system. In a system use case, you include high-level implementation decisions. System use cases can be written in both an informal manner and a formal manner.

Following are the purposes of a use case diagram given below:

It gathers the system's needs.

It depicts the external view of the system.

It recognizes the internal as well as external factors that influence the system.

It represents the interaction between the actors.

Following example will illustrate on how to plan use cases:

Use Case: What is the main objective of this use case. For e.g., Adding a software component, adding certain functionality etc.

Primary Actor: Who will have the access to this use case. In the above examples, administrators will have the access.

Scope: Scope of the use case

Level: At what level the implementation of the use case be.

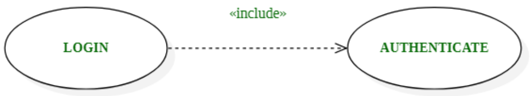
Flow: What will be the flow of the functionality that needs to be there. More precisely, the work flow of the use case.

Difference between <<include>> and <<extend>> in Use Case Diagram:

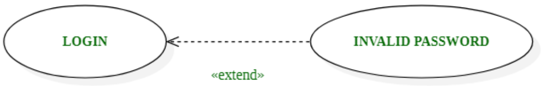
TRCA

<<include>> extends Base Use Case and it specifies that an Included Use Case must run successfully to complete Base Use Case. The Base Use Case is incomplete in the absence of an Included Use Case. The Included Use Case can be Base Use Case itself or it might be shared by a number of distinct Base Use Cases.

<<extend>> on the other end, is used to add an Extended Use Case which extends the Base Use Case. Base Use Case can run successfully even without invoking/calling extended use case called Optional Use Case. The Base Use Case is complete in itself but under certain conditions it would require to refer to extension condition.



LOGIN is the BASE USE CASE and AUTHENTICATE is the INCLUDED USE CASE.



ILOGIN is the BASE USE CASE and INVALID PASSWORD is the EXTENDED USE CASE.

BANKING MANAGEMENT SYSTEM USE CASE DIAGRAM

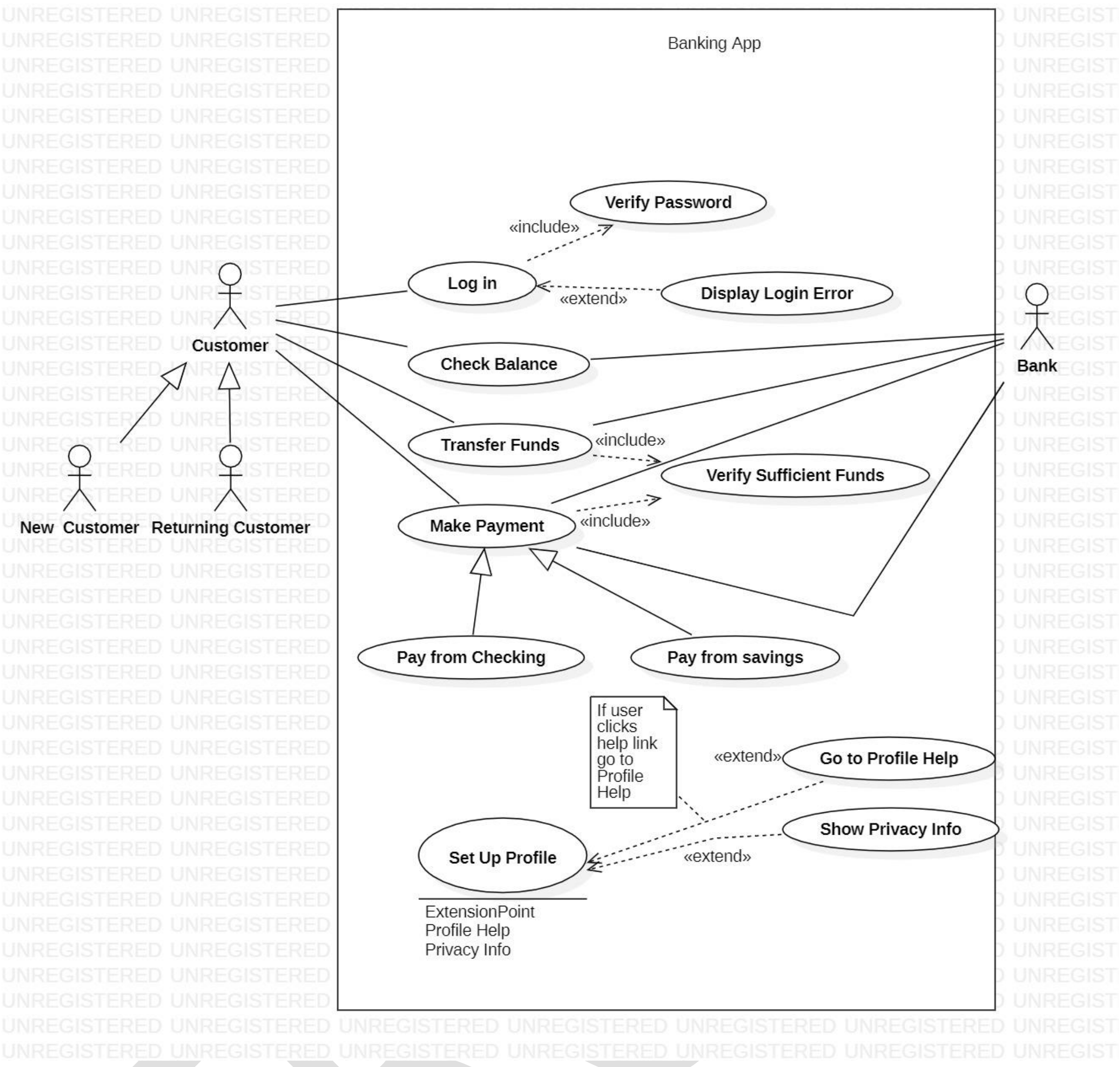
This Use case diagram is a graphic depiction of the interactions among the elements of Banking Management System. It represents the methodology used in system analysis to identify, clarify and organize system requirements of Banking Management System.

The main actors of Banking Management System are: **Customer and Bank**

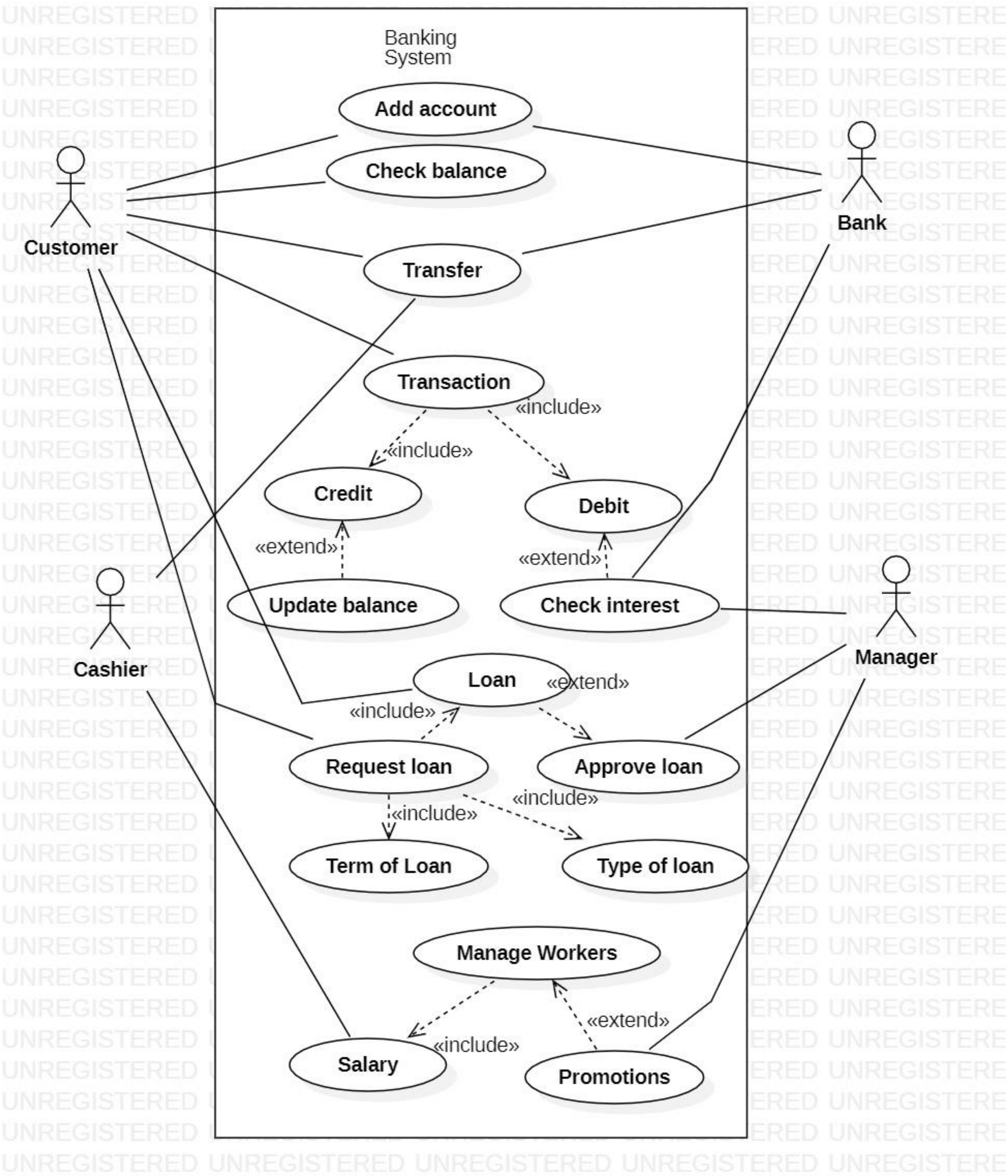
The relationships between and among the actors and the use cases of Banking Management System are as follows:

Customer: Use Cases of Customer are **Log in, Check Balance, Transfer Funds and Make Payment**

Bank: Use Cases of Customer are **Check Balance, Transfer Funds, Make Payment**



UNREGISTERED



Here, we will try to understand the design of a use case diagram for the Online Banking System. Some possible scenarios of the system are explained as follows:

1. A Customer is required to create an account to avail services offered by Bank. Bank verifies detail and creates new account for each new customer. Each customer is an

actor for the Use-Case Diagram and the functionality offered by Online Banking System to Add Account is Use-Case.

2. Each customer can check the balance in bank account and initiate request to transfer an account across distinct branches of Bank. Cashier is an employee at bank who supports service to the customer.
3. A customer can execute cash transactions where the customer must either add cash value to bank account or withdraw cash from account. Either of two or both that is credit as well as debit cash, might be executed to successfully execute one or multiple transactions.
4. After each successful transaction customer might or might not want to get details for action. Manager can check interest value for each account corresponding to transaction to ensure and authenticate details.
5. A customer can also request loan from bank where customer must add request for loan with the appropriate details.
6. The type of loan in accordance with purpose or the need for loan and term or duration to pay back the loan must be provided by customer.
7. The manager of each branch of bank has choice to either accept or approve loan to initiate process further or just reject request for loan based on terms and conditions.
8. The record for each employee of bank is maintained by bank and bank manages all employees of each branch of bank. The manager of each branch has choice to offer bonus to employees. Note here that each employee is paid as part of management of staff but promotion or bonus might or might not be offered certainly to each employee.

This is the complete design and description for Use-Case of an Online Banking System specifying the use of <<include>> and <<extend>> for certain specific Use-Cases.

TRCAC