TASK-4

Identify Use Cases and develop the Use Case model for the given scenario. Draw the relevant Entity Relationship Diagram and Class Diagram.

Aim:

The aim is to identify the Use Cases and develop the Use Case Diagram, Entity Relationship Diagram, and Class Diagram for the given Online Banking System scenario.

Use Case Diagram

A Use Case Diagram represents how users interact with the Online Banking System and illustrates the relationships between users (actors) and various system functionalities.

Use Case

A use case describes a sequence of actions performed by the system to achieve a specific result visible to the user, such as performing a funds transfer or checking an account balance.

Actor

An actor represents a role that interacts with the system. It could be a person (e.g., customer, admin) or an external service (e.g., payment gateway).

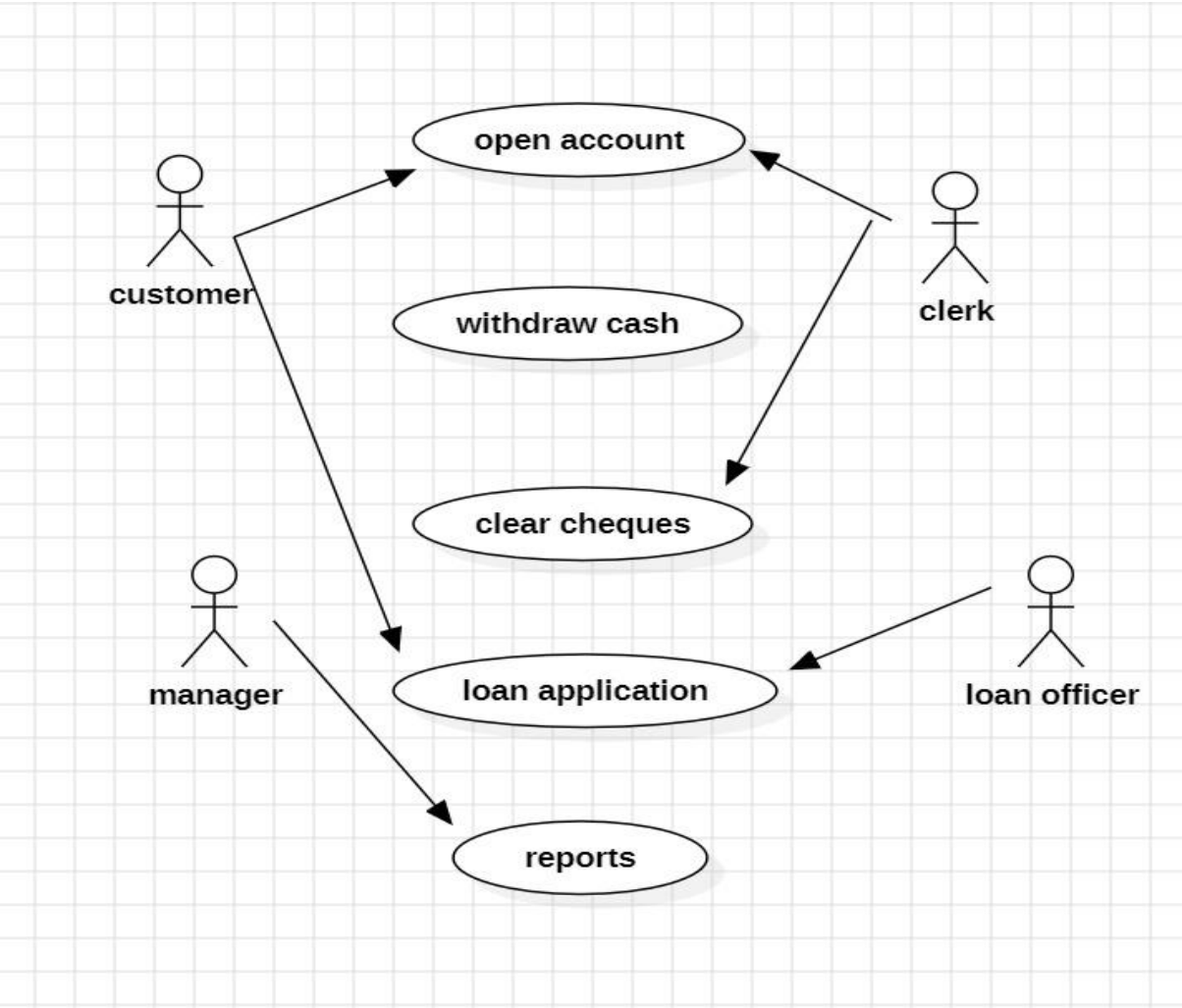
Actors Identified

- Customer
- Administrator
- System

Description:

The Online Banking System use case diagram depicts various actors and their interactions with the system.
The Customer can perform operations such as registering, logging in, checking balance, transferring funds, paying bills, managing beneficiaries, and viewing transaction history.
The Administrator can monitor user accounts, approve transactions, generate reports, and manage system configurations.
The System automatically handles tasks like OTP generation, balance updates, transaction logging, and notification delivery.

The use case diagram encapsulates the core functionalities of the Online Banking System, ensuring secure, efficient, and user-friendly banking operations through digital interfaces.



USE CASE DIAGRAM

E-R Diagram

The Entity-Relationship (ER) Diagram illustrates how entities in the Online Banking System are related. It helps design the database structure that stores information about users, accounts, and transactions.

Components of an ER Diagram

- Entities
- Attributes
- Relationships

Description:

The ER Diagram for the Online Banking System demonstrates relationships among entities such as Customer, Account, Transaction, and Administrator.

Each Customer has one or more Accounts, and each Account can have multiple Transactions. The Administrator oversees and manages these records.

Key relationships include:

- Customer–Account: One-to-many relationship (a customer can hold multiple accounts).
- Account–Transaction: One-to-many relationship (an account can have multiple transactions).
- Administrator–Transaction: One-to-many relationship (admin can monitor multiple transactions).

Each entity includes attributes like:

- Customer: CustomerID, Name, Address, Contact, Email
- Account: AccountNo, AccountType, Balance, Status
- Transaction: TransactionID, Date, Amount, Type, Status
- Administrator: AdminID, Name, Role

Class Diagram

A Class Diagram describes the structure of the Online Banking System by defining its classes, their attributes, methods, and relationships.

Purpose:

- To analyze and design the static structure of the system.
- To define class responsibilities and relationships.
- To serve as the foundation for database and component design.

Common Elements in the Class Diagram:

- Classes
- Interfaces
- Associations
- Dependencies and Generalizations

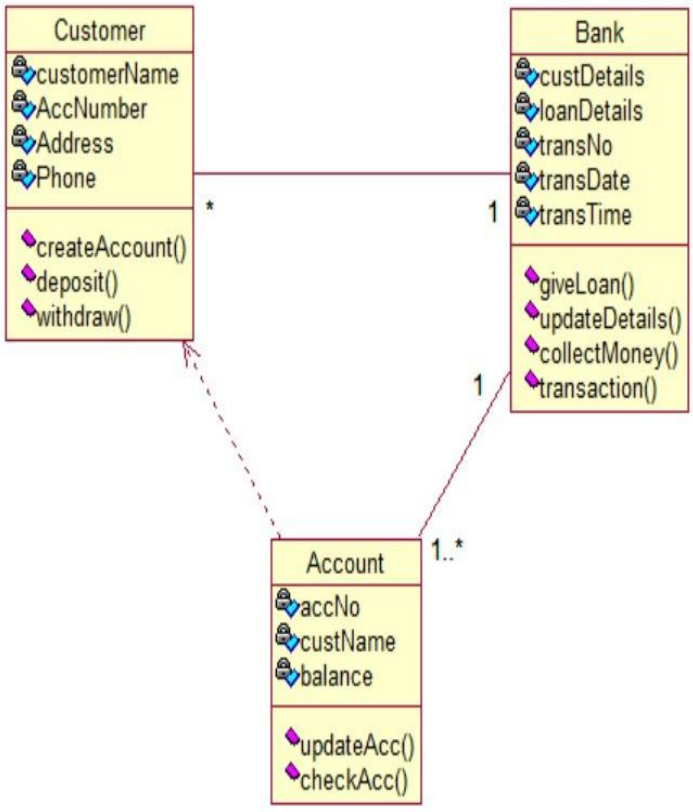
Description:

The Class Diagram for the Online Banking System consists of the following main classes:

1. Customer Class

- Attributes: customerID, name, contact, email, password
- Methods: register(), login(), viewAccount(), transferFunds(), viewTransactions(), updateProfile()

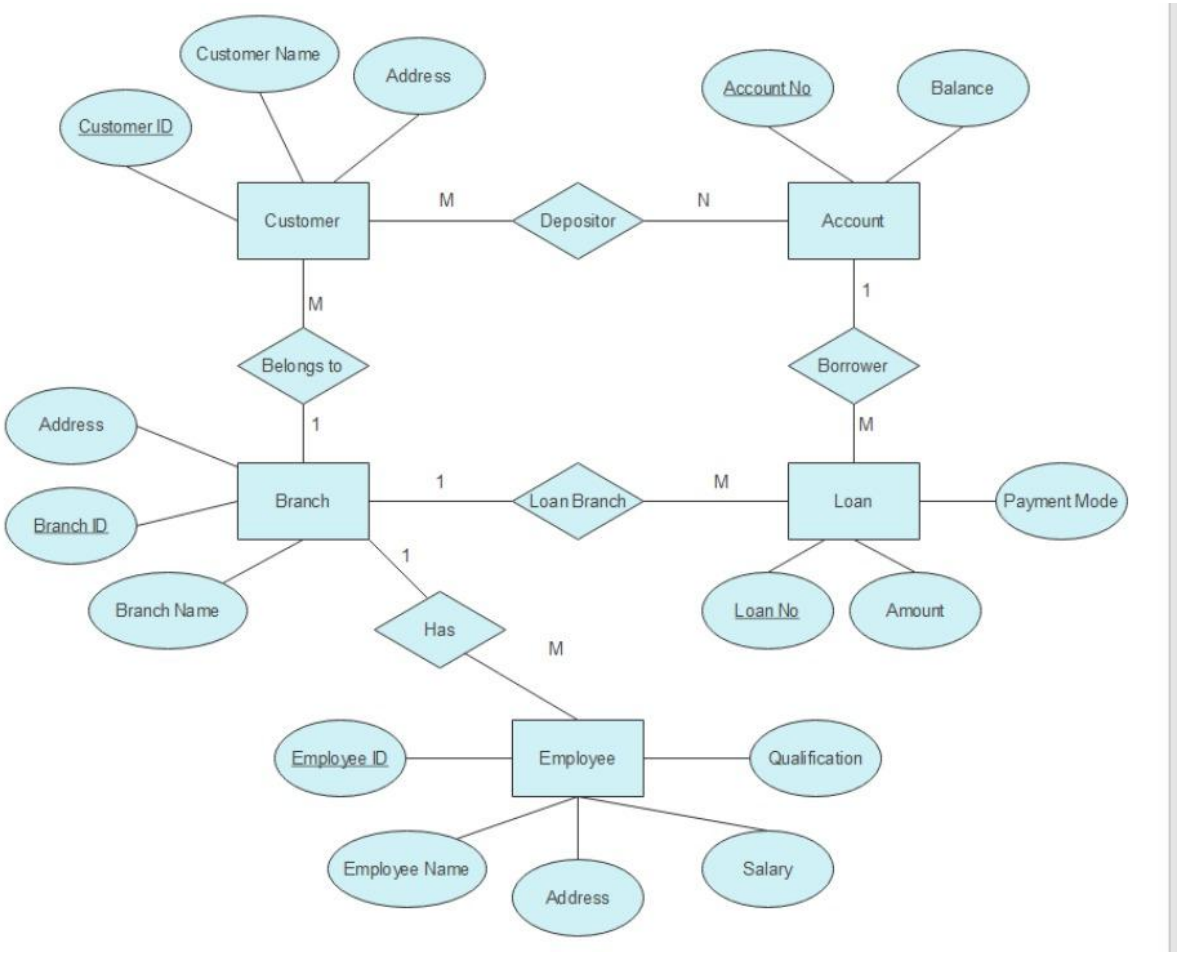
2. Account Class
   - Attributes: accountNo, accountType, balance, status
   - Methods: deposit(), withdraw(), updateBalance(), getBalance()
3. Transaction Class
   - Attributes: transactionID, date, amount, transactionType, status
   - Methods: initiateTransaction(), confirmTransaction(), generateReceipt()
4. Administrator Class
   - Attributes: adminID, name, role
   - Methods: approveTransaction(), manageUsers(), generateReport(), viewSystemLogs()



CLASS DIAGRAM

Relationships:

- The Customer class is associated with one or more Accounts.
- Each Account is linked to multiple Transactions.
- The Administrator class manages both Customer and Account data.
- Associations include one-to-many relationships between Customer–Account and Account–Transaction.



ENTITY RELATION DIAGRAM

Result:

Thus, the Use Cases have been identified, and the Use Case Diagram, Entity Relationship Diagram, and Class Diagram for the Online Banking System have been designed successfully.