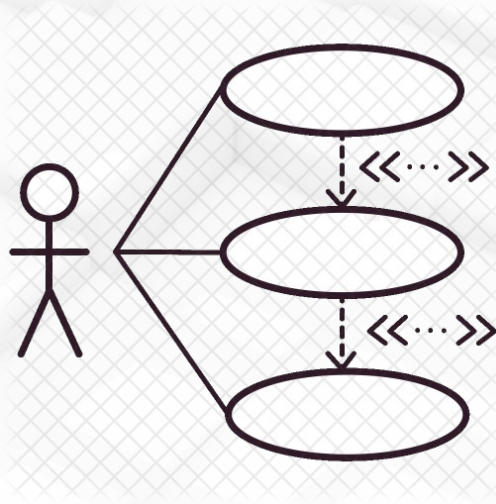
The background of the slide is a 3D grid of white cubes. One cube, located in the upper right quadrant, is red and stands slightly higher than the others, creating a focal point. A red semi-transparent rounded rectangle is positioned on the left side of the slide, containing the title and author information in white text.

Object Oriented Analysis and Design ICT2142

Chanduni Gamage
Dept. of ICT
Faculty of Technology
University of Ruhuna

LECTURE 06:

Use Case Diagrams



A decorative graphic on the left side of the slide. It features a 3D-rendered red cube with a glossy finish, positioned on a grid of light gray squares. The grid extends from the bottom left towards the center of the slide, creating a sense of depth and perspective.

Objectives

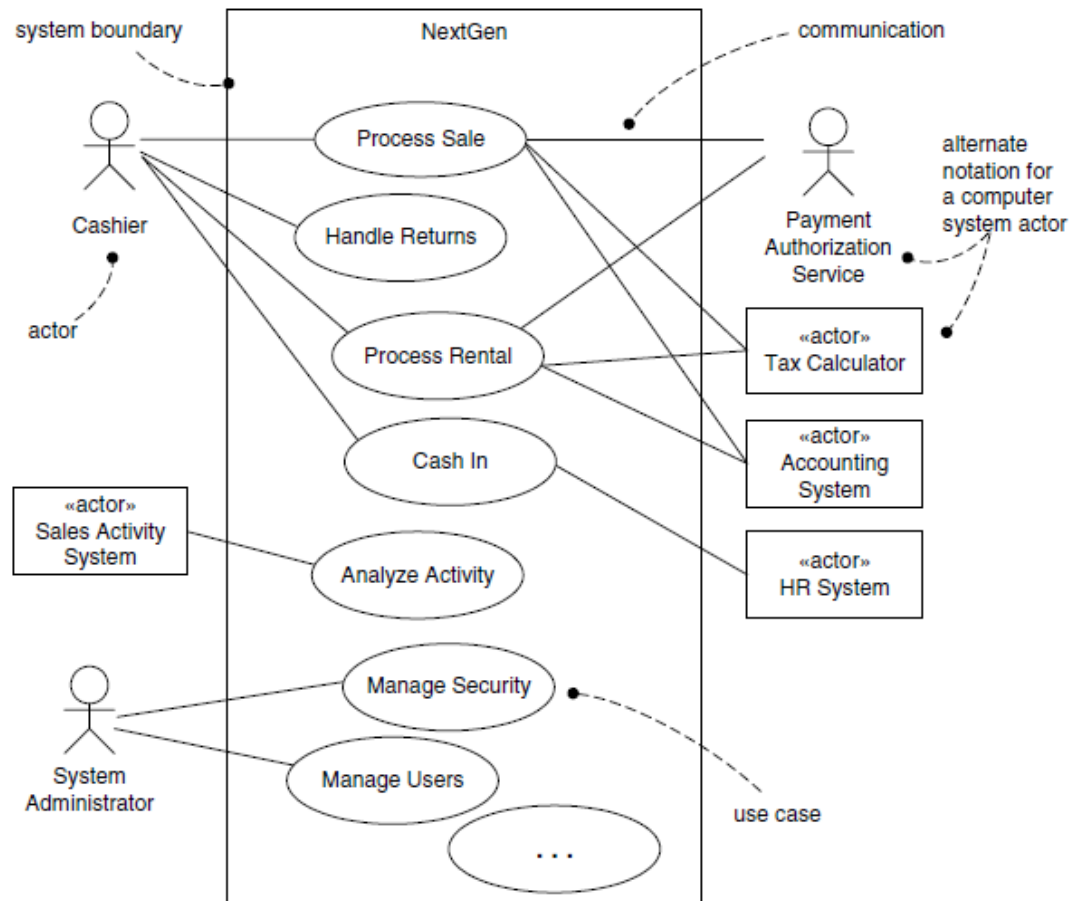
After successful completion of this lecture, students should be able to:

- Draw use case diagrams for given scenarios

Introduction to Use Cases

- The UML provides a notation called the use case diagram to illustrate use cases, actors and the relationships between them.
- Describe **a set of use cases** that some system or systems should or can perform in collaboration with one or more external users of the system.





Components of a Use Case Diagram

Actors

- Roles played by people or things that use the system.

Use cases

- Things that the actors can do with the system.

Relationships

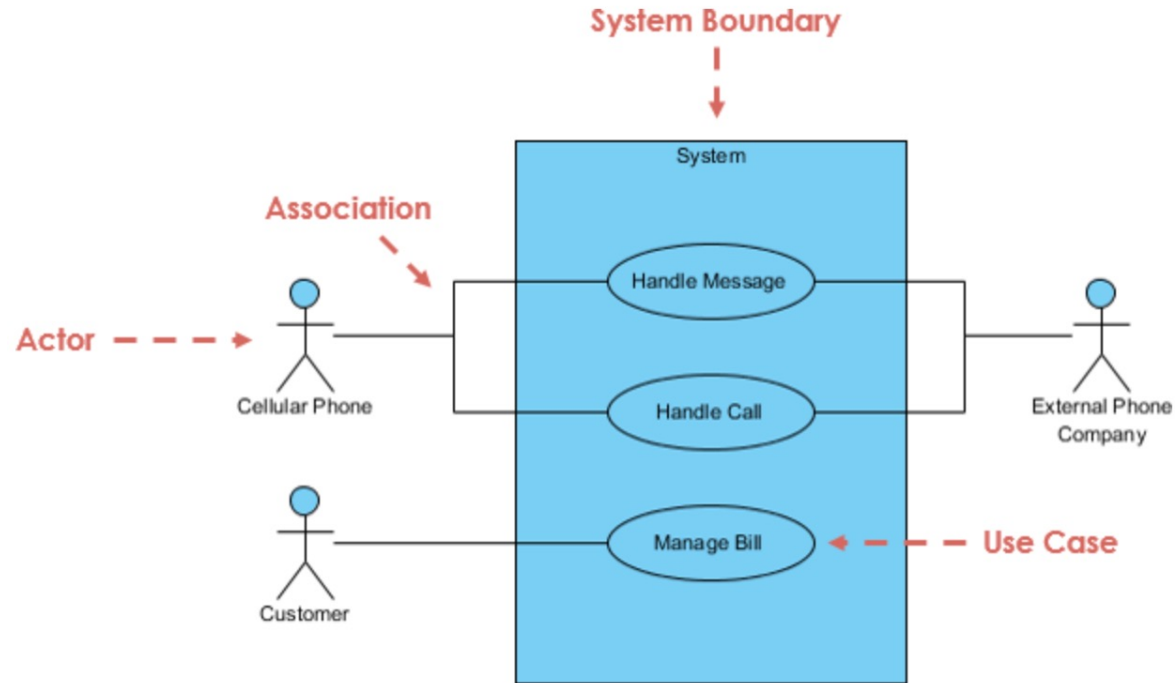
- Meaningful relationships between actors and use cases.

System boundary

- A box drawn around the use cases to denote the edge or boundary of the system being modeled.



Components of a Use Case Diagram



Use Case Notations



Actor

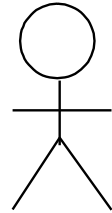
- Actors are the entities that interact with a system.
- Although in most cases, actors are used to represent the users of system, actors can actually be anything that needs to exchange information with the system.
- So, an actor may be people, computer hardware, other systems, etc.
- Note that actor represents a role that a user can play but not a specific user.
- So, in a hospital information system, you may have doctor and patient as actors but not Dr. John, Mrs. Brown as actors.



Actor

Actors can be represented using the following notations.

external computer system actors





Cashier

or

«actor»
Inventory
Control
System

Use Cases

- A use case is something an actor wants the system to do.
- Use cases are *always* started by an actor.
- Use cases are *always* written from the point of view of an actor.
- The best way of identifying use cases is to start with the list of actors, and then consider how each actor is going to use the system.
- Using this strategy you can obtain a list of candidate use cases. Each use case must be given a short, descriptive name.



PlaceOrder

GetStatus
OnOrder

Use Cases

For example, the following would be considered use cases for a university information system:

Enroll students in courses

Output seminar enrolment lists

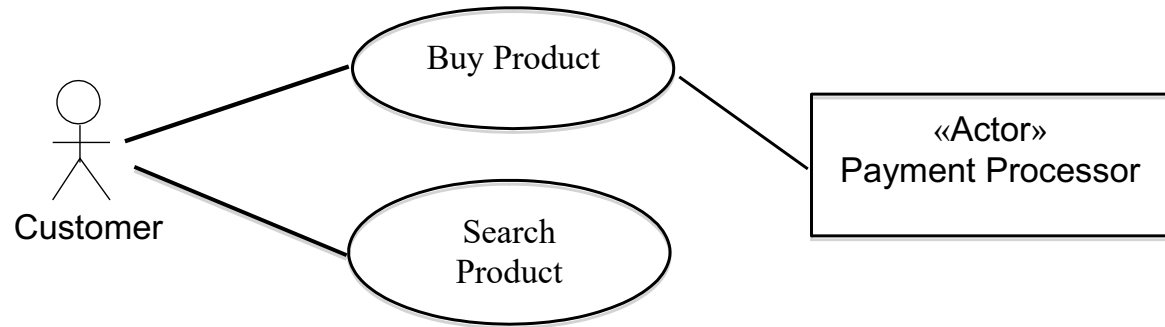
Remove students from courses

Produce student transcripts



Association

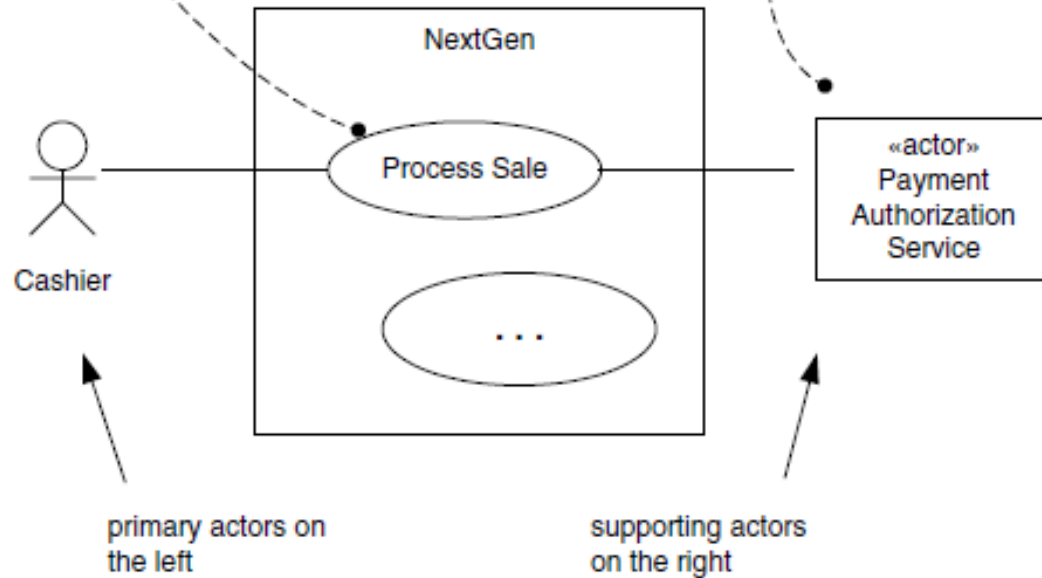
Actors are associated with use cases by solid lines



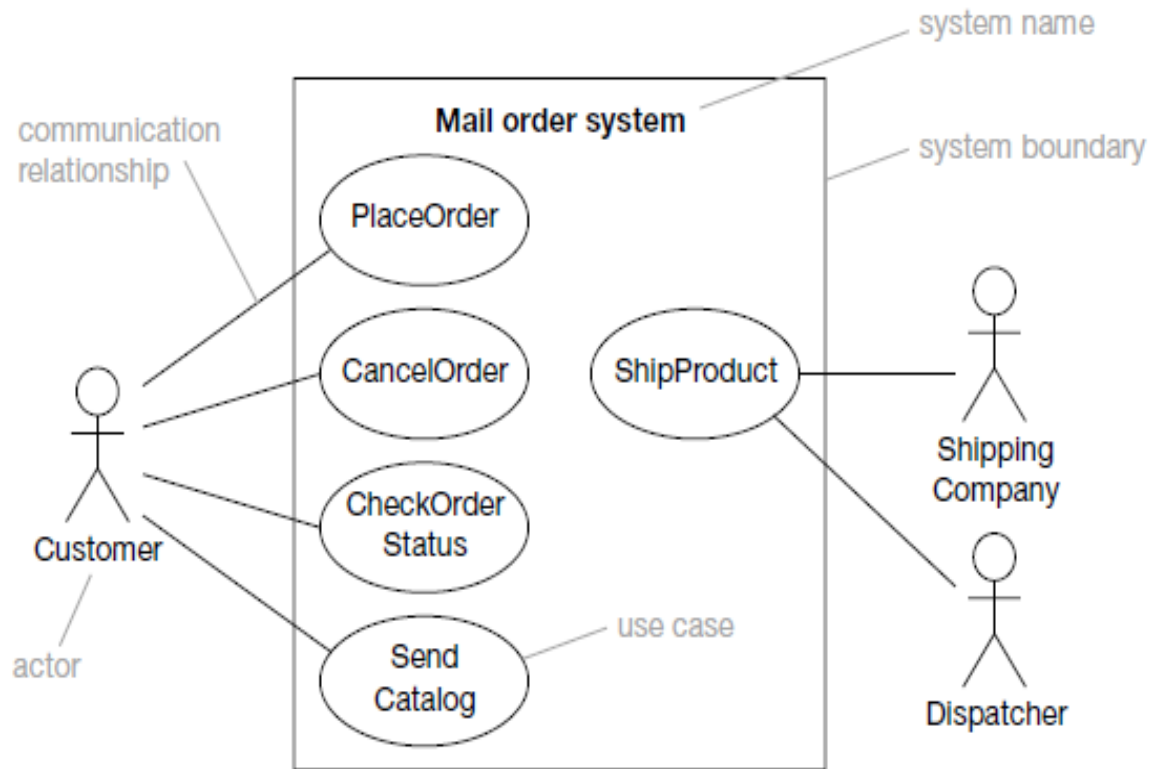
Summary

For a use case context diagram, limit the use cases to user-goal level use cases.

Show computer system actors with an alternate notation to human actors.



Use Case Diagrams



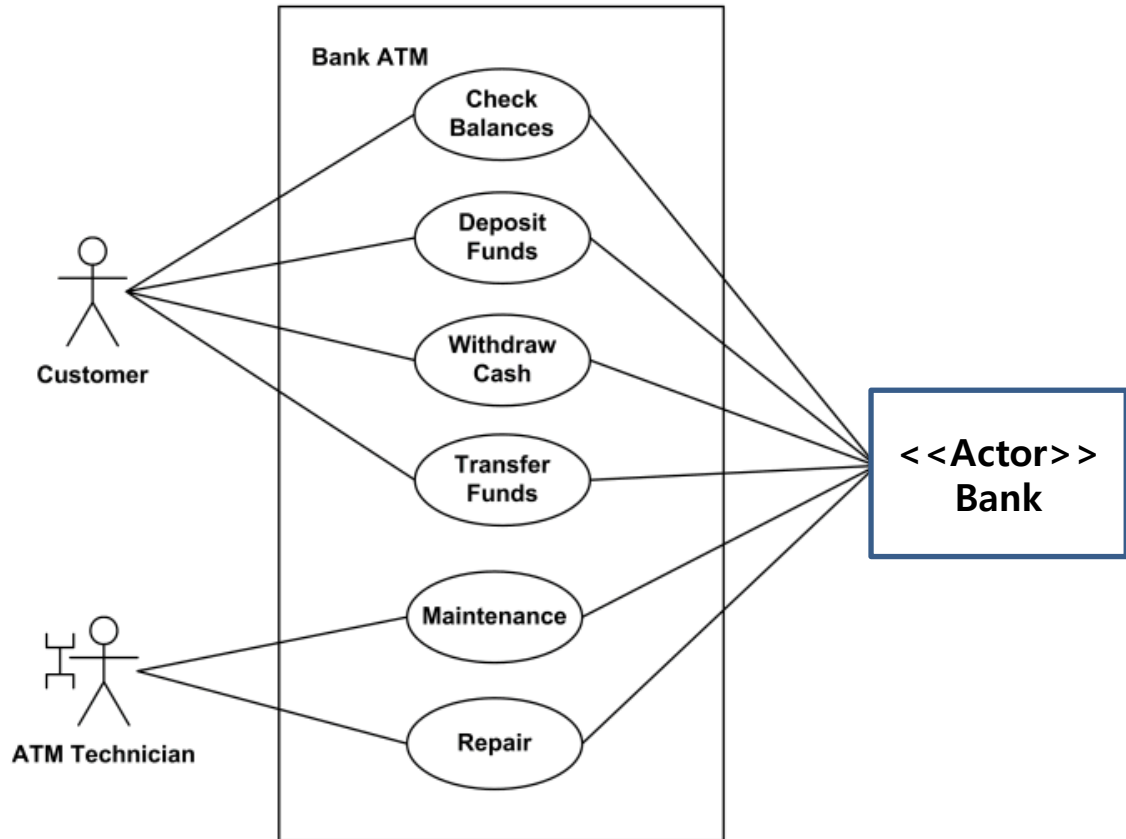
Exercise 1

Describe use cases that an automated teller machine (ATM) or the automatic banking machine (ABM) provides to the bank customers using a use case diagram.

Customer uses a bank ATM to check balances of his/her bank accounts, deposit funds, withdraw cash and/or transfer funds (use cases). ATM Technician maintains and repairs the ATM

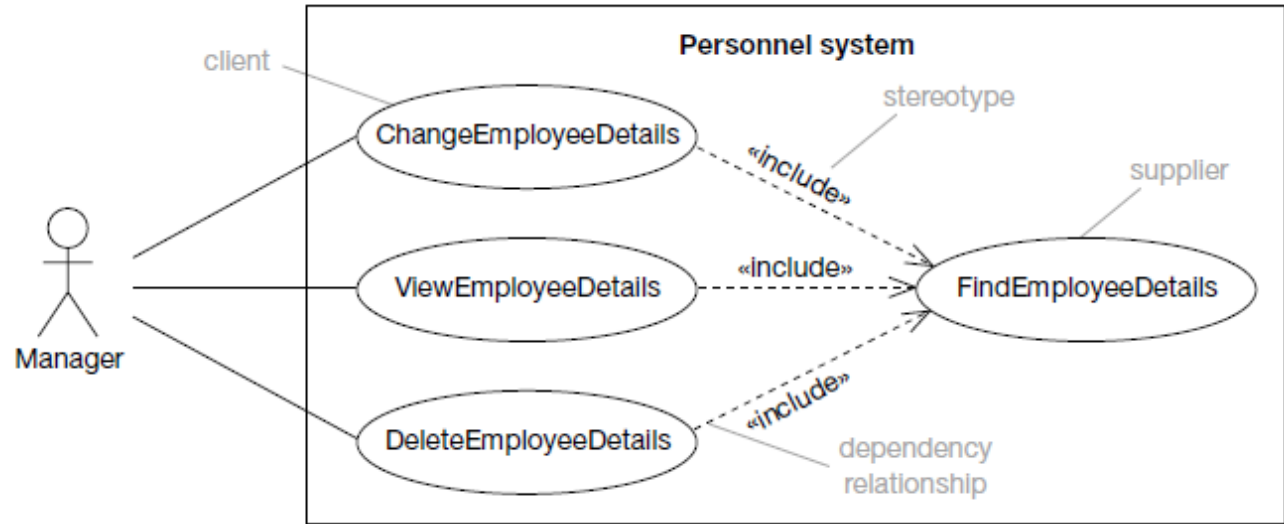


Answer



<<include>>

The «include» relationship between use cases allows you to include the behavior of a supplier use case into the flow of a client (base) use case.



A red cube is positioned on a grid of white cubes, which is part of a larger 3D structure. The red cube is the focal point, standing out from the white ones.

<<include>>

- The *including* use case is referred as the *client* use case, and the *included* use case as the *supplier* use case.
- This is because the included use case supplies behavior to its client use case.
- The base use case is incomplete without the included use case (mandatory).

Execution

- The client use case executes until the point of inclusion is reached, then execution passes over to the supplier use case.
- When the supplier finishes, control returns to the client again.

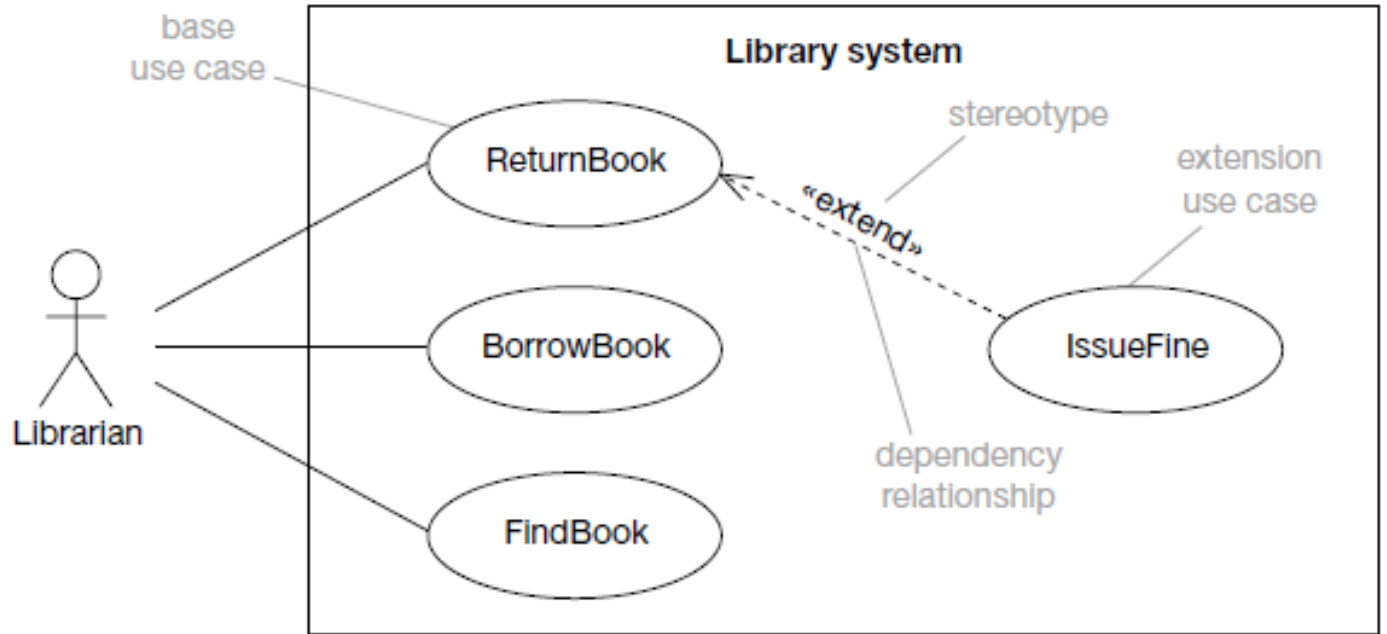
Example



- The figure illustrates an e-commerce application that provides customers with the option of checking the status of their orders.
- This behavior is modeled with a **base use case called CheckOrderStatus** that has an **inclusion use case called 'LogIn'**.
- The **LogIn** use case is a separate inclusion use case because it contains behaviors that several other use cases in the system use.
- An include relationship points from the **CheckOrderStatus** use case to the **LogIn** use case to indicate that the **CheckOrderStatus** use case *always includes the behaviors* in the **LogIn** use case.

<<extend>>

Provides a way to add new behavior to an existing use case



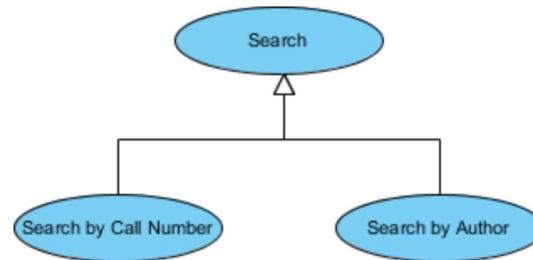
<<extend>>

- The base use case provides a set of extension points where new behavior may be added, and the extension use case provides a set of insertion segments that can be inserted into the base use case
- The direction of an <<extends>> relationship is to the extended (base) use case.
- Extend doesn't always mean it's optional. Sometimes the use case connected by extend can supplement the base use case.
- But the base use case should be able to perform a function on its own even if the extending use case is not called.



Generalization Relationship

- The Generalization Relationship establishes an "is-a" connection between two use cases.
- A generalization relationship is a parent-child relationship between use cases.
- The child use case is an enhancement of the parent use case.
- Generalization is shown as a directed arrow with a triangle arrowhead.
- The child use case is connected at the base of the arrow. The tip of the arrow is connected to the parent use case.



Summary of Relationship in the Use Case Diagrams

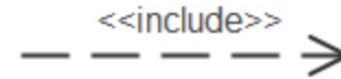
Association relationships



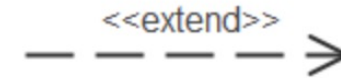
Generalization relationships



Include relationships



Extend relationships



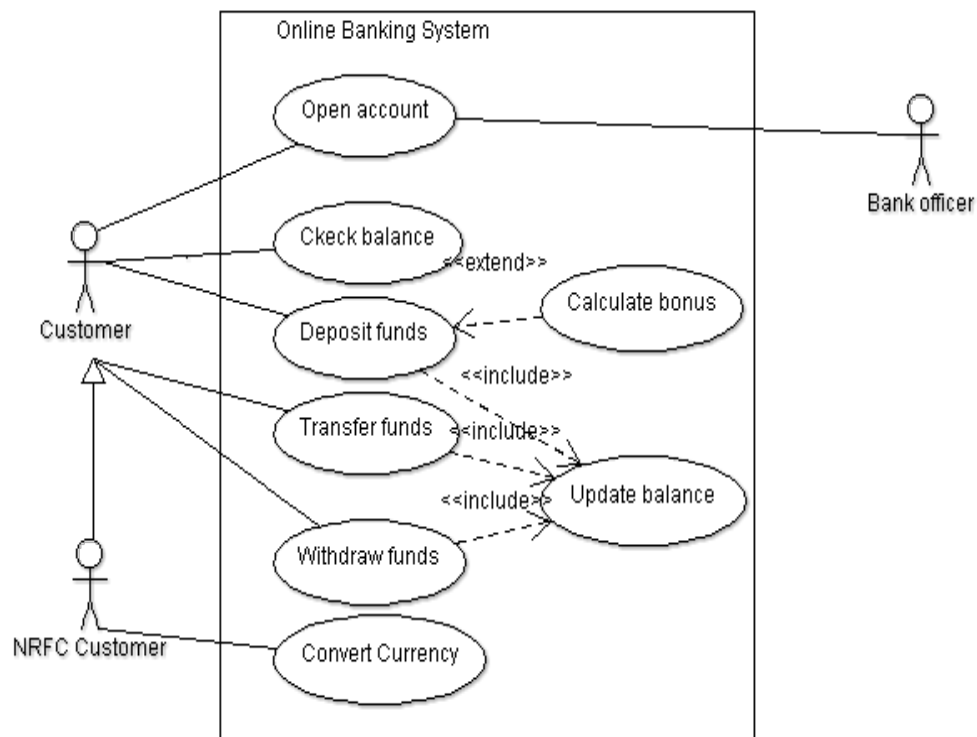
Exercise 1

In an online banking system, customers can perform various activities such as opening accounts, checking account balances, depositing funds, transfer ring funds, and withdrawing funds from their accounts. For Non-Resident Foreign Currency (NRFC) customers, the system also provides the functiona lity to convert currencies online.

When a customer opens an account online, a bank officer is involved in the process to ensure proper verification and setup. The system automatically updates the account balance whenever a customer deposits, withdraws or transfers money. Additionally, if a customer deposits an amount exceeding Rs. 10,000.00, the system calculates and applies a bonus to the deposit.



Answer





Exercise 2

A hospital reception system is designed to assist receptionists in managing patient-related tasks efficiently. The system provides various functionalities to streamline processes such as patient appointments, hospital admissions, insurance claims, and medical report filing. Receptionists play a crucial role in handling these tasks to ensure smooth hospital operations.

One of the key responsibilities of the receptionist is to schedule patient appointments. Additionally, before any hospital admission, it is mandatory to register the patient. The hospital admission process itself is divided into two categories: outpatient and inpatient admissions. Inpatient hospital admissions require an additional step of bed allotment to ensure proper patient accommodation. Scheduling a hospital admission is an extension of the patient registration process, reinforcing the importance of proper documentation before admission.

Beyond managing patient admissions, the hospital reception system also supports other essential functionalities. Receptionists can file medical reports and handle insurance forms or claims, ensuring patients have the necessary documentation for their treatments and reimbursements. Furthermore, scheduling patient appointments is an optional extension of the system, allowing patients to plan their visits in advance. By integrating these features, the hospital reception system enhances efficiency, reduces administrative burdens, and improves overall patient experience.

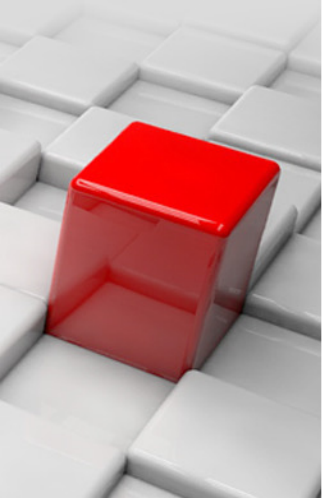
Exercise 3

A university library management system is designed to facilitate seamless interactions between users, librarians, and the system database, ensuring efficient handling of book reservations, renewals, fines, and catalog updates. The system supports two types of users: students and staff, both of whom can access various library-related services.

Before performing any actions, users must authenticate themselves within the system. If the authentication fails, the system will notify them of an invalid username or password. Once authenticated, users can reserve books, request new ones, and renew existing loans. However, any invalid renewal attempts will be flagged by the system. Users who have overdue books are required to pay fines before they can borrow additional materials. Additionally, they can provide feedback by filling out a designated feedback form. For new members, the system allows registration, which involves completing a registration form and obtaining a unique library card ID.

Librarians play a crucial role in maintaining the library's book catalog. They are responsible for updating book records, which includes adding new book items, editing existing book information, and deleting outdated or unavailable books. The system ensures that any modifications made to the catalog are accurately reflected and accessible to users.

To enhance efficiency and communication, the library management system integrates automated notifications. The system will send overdue notifications to remind users of pending returns, notify them when a reserved book becomes available, and issue alerts for reservation cancellations. By incorporating these functionalities, the university library management system streamlines library operations and improves the overall user experience.



THANK YOU!

