Software Requirement

Use Case Model

- In the first lesson we discuss expressing requirement type and say we have model for this type like
- Use case , story board , user story
- We will discuss use case because it is the most popular

use case

A use case diagram is a graphic depiction of the interactions among the elements of a system.

used to analyze various systems. They enable you to visualize the different types of roles in a system and how those roles interact with the system.

Importance of Use Case Diagrams

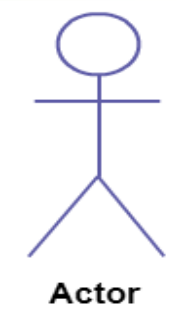
- To identify functions and how roles interact with them
- For a high level view of the system
- To identify internal and external factors

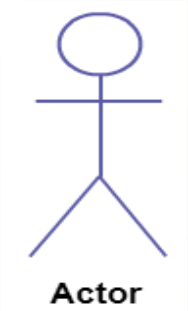
Use Case Diagram objects

- Actor
- Use case
- System
- Package

Actor

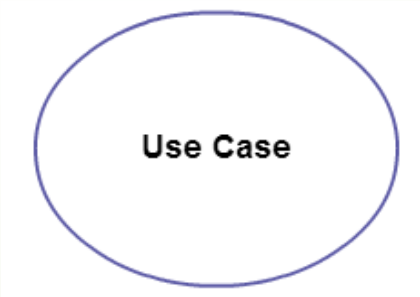
- Actor in a use case diagram is any entity that performs a role in one given system. This could be a person, organization or an external system and usually drawn like skeleton





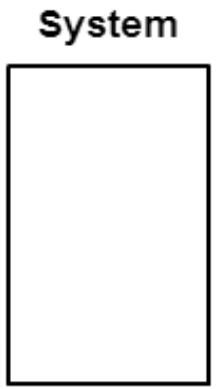
Use Case

- A use case represents a function or an action within the system. Its drawn as an oval and named with the function.



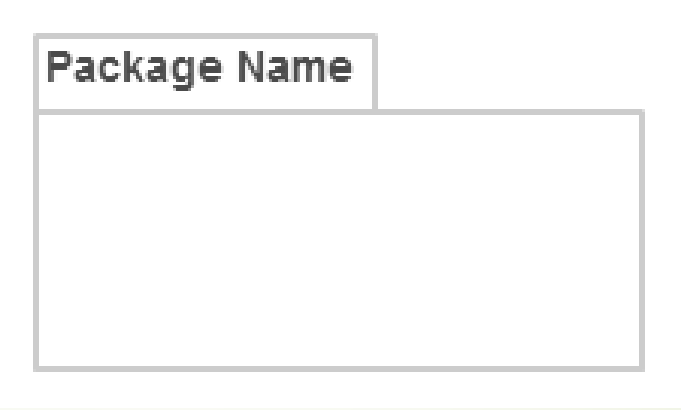
System

- System is used to define the scope of the use case and drawn as a rectangle



Package

- Package is another optional element that is extremely useful in complex diagrams



Relationships in Use Case Diagrams

- There are main type

1. Association
2. Include Relationships

Other type

1. Generalization of an actor
2. Extend relationship between two use cases

Association

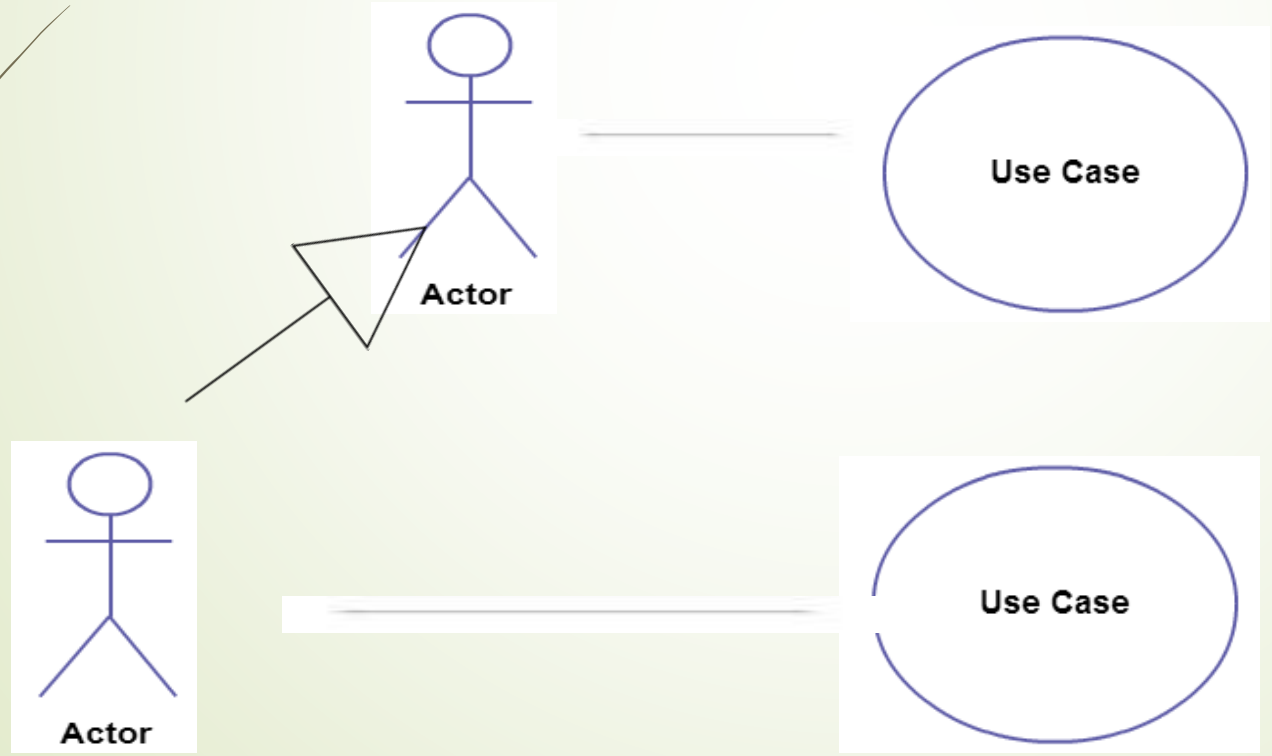
- An association is a connection between an actor and a use case. An association indicates that an actor can carry out a use case

Include

- It indicates that the use case to which the arrow points is included in the use case on the other side of the arrow.
- The base use case is incomplete without the included use case.
- The included use case is mandatory and not optional.

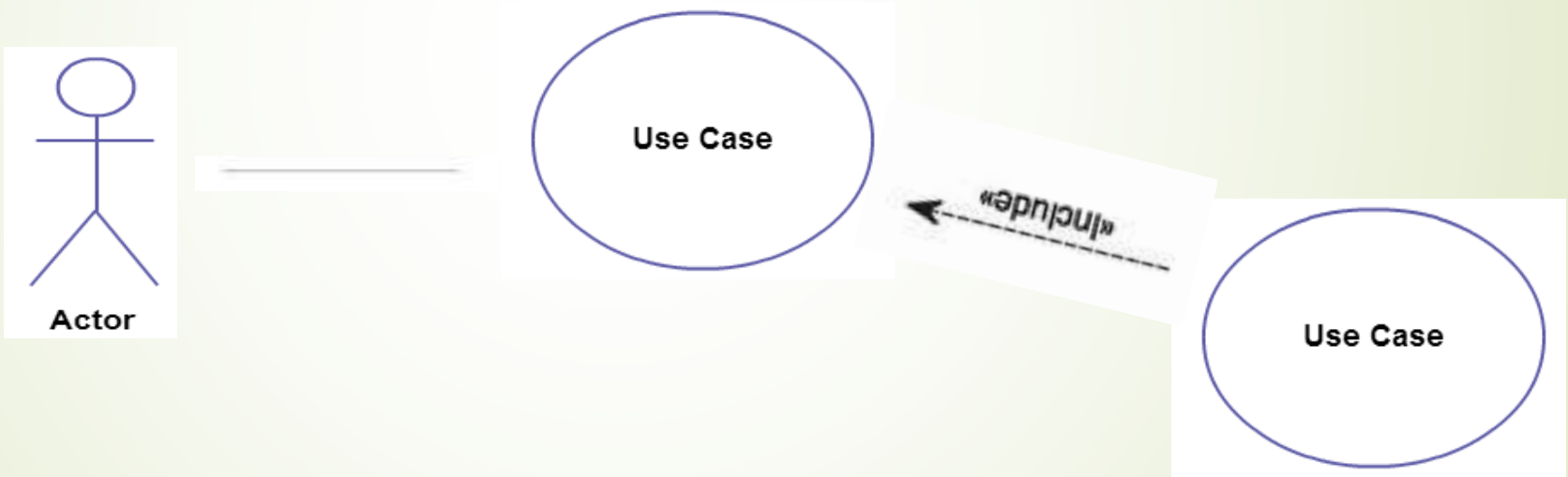
1. Generalization of an Actor:

- Generalization of an actor means that one actor can inherit the role of an other actor. The descendant inherits all the use cases of the ancestor.



Extend Relationship

- The extending use case is dependent on the extended (base) use case.
- Although extending use case is optional most of the time it is not a must



EX:
bank accounting customer

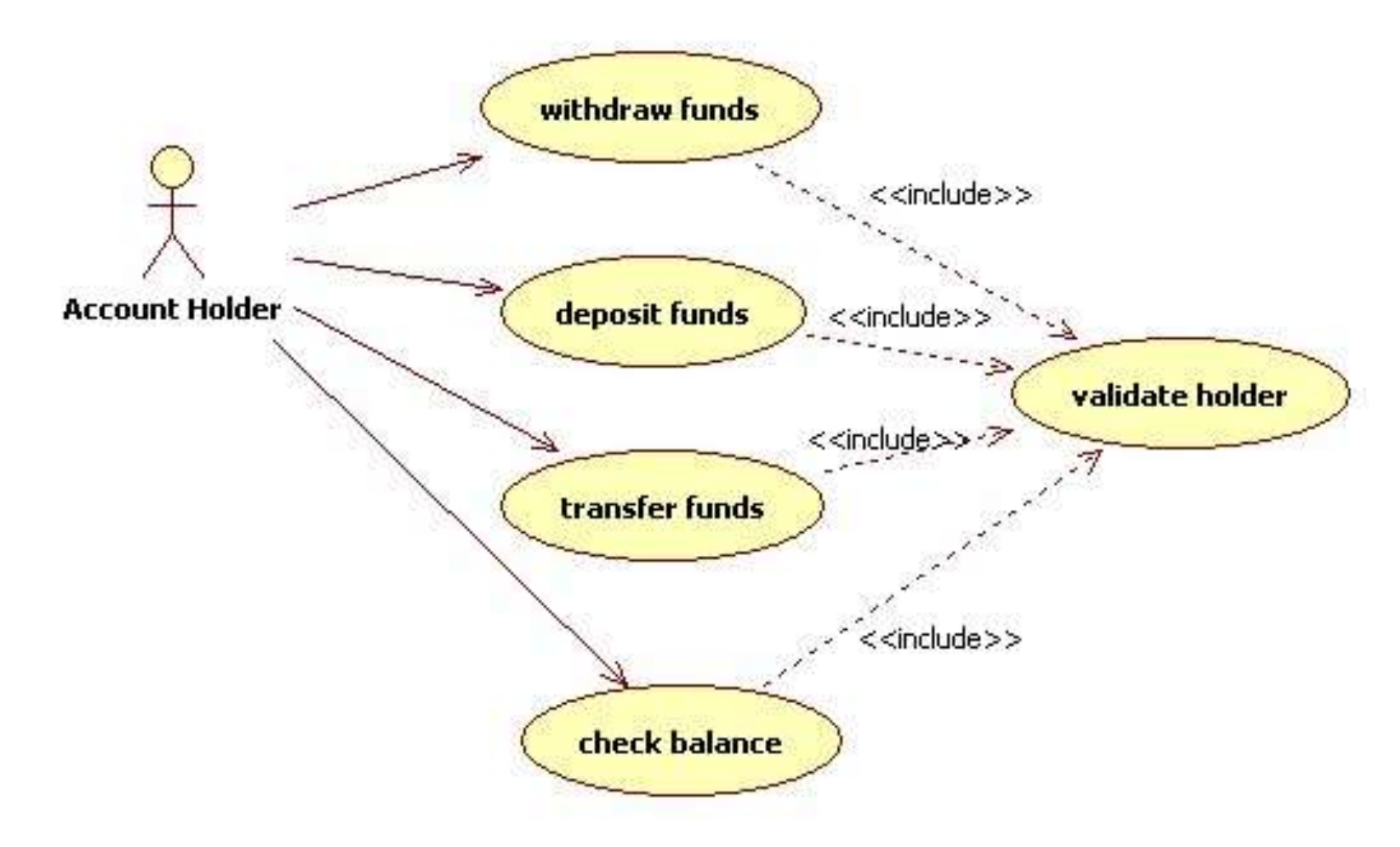
Actors

Bank , customers

Functions

Check balance , deposit ,transfer, validate balance

We will take use case of customers only


withdraw funds
Account Holder
deposit funds
transfer funds
check balance
validate holder
<<include>>
<<include>>
<<include>>
<<include>>

next step:

- You will take one use case of customers and represent it through this table
- You will write all requirement in this table about this use case and you can link it to other use case
- Through your meeting and understanding client requirement you will this data

Name	View Bill
Participating Actors	Customer
Goals	View the Bill for the Order
Triggers	Request to View Bill
Pre-Condition	Menu Items on Menu, Selecting Dish, Placing Order
Post-Condition	View Bill and Pay for Bill
Basic Flow	1) User Requests to View Bill 2) User Views Bill
Alternate Flows	User Gets Wait Staff to Print and Bring Them Bill
Exceptions	No Dishes Ordered
Qualities	• Bill Available After Order placed • Bill Takes Less then 10 Seconds to Load • All Dishes That Were Selected Appear on the Bill • Prices on Bill Match Prices on Menu