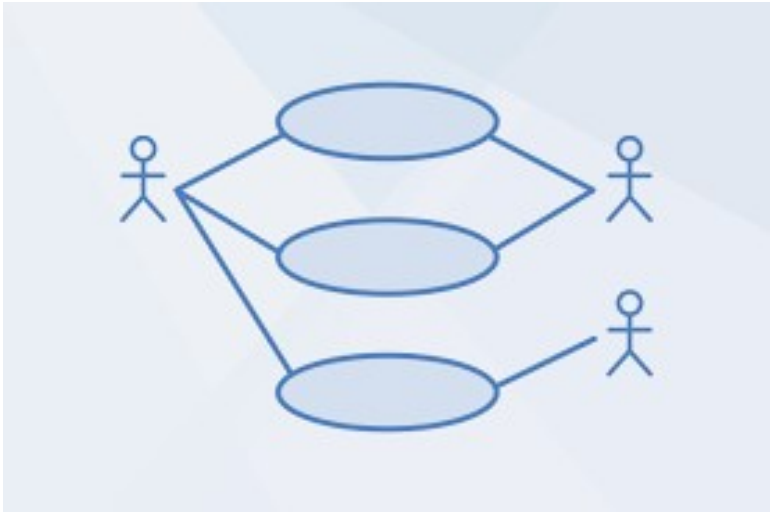
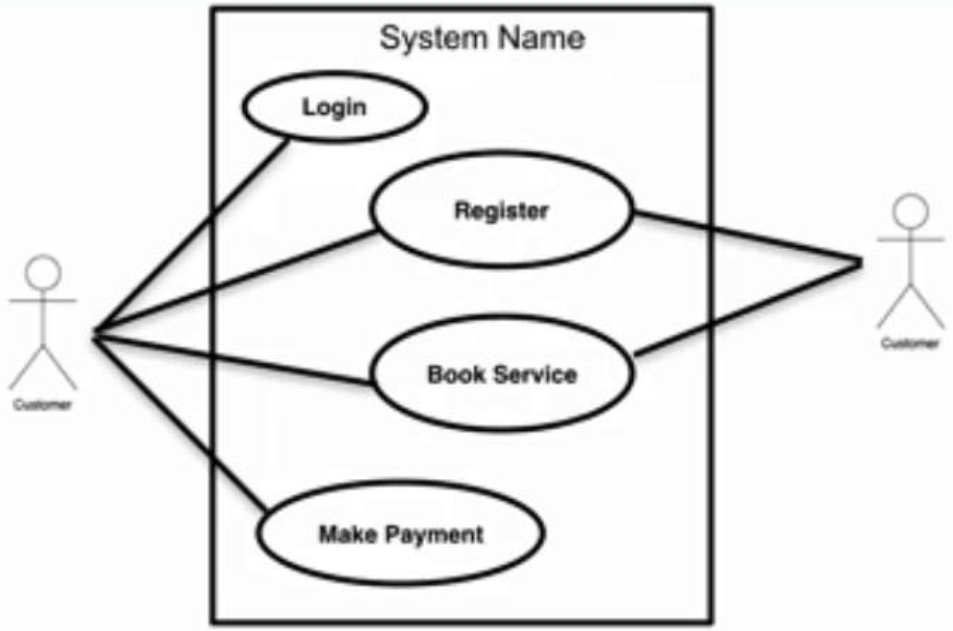


USECASE DIAGRAM

What is the Use Case Diagram?

Use Case Diagram captures the system's functionality and requirements by using actors and use cases. Use Cases model the services, tasks, function that a system needs to perform. Use cases represent high-level functionalities and how a user will handle the system. Use-cases are the core concepts of Unified Modelling language modelling.

Simply use case diagram shows a system or application, people or organization, or others that interact with the system.



4 different elements of the USE CASE DIAGRAM

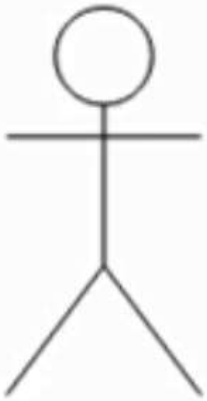
1. **System**
2. **Actors**
3. **Use Cases**
4. **Relationships**

1. System The project that you are developing is a system.

application software, a website, software component, business process, or any other projects

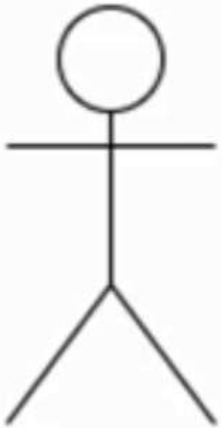
2. Actor

Anyone how use to perform a certain function in the system is actors.

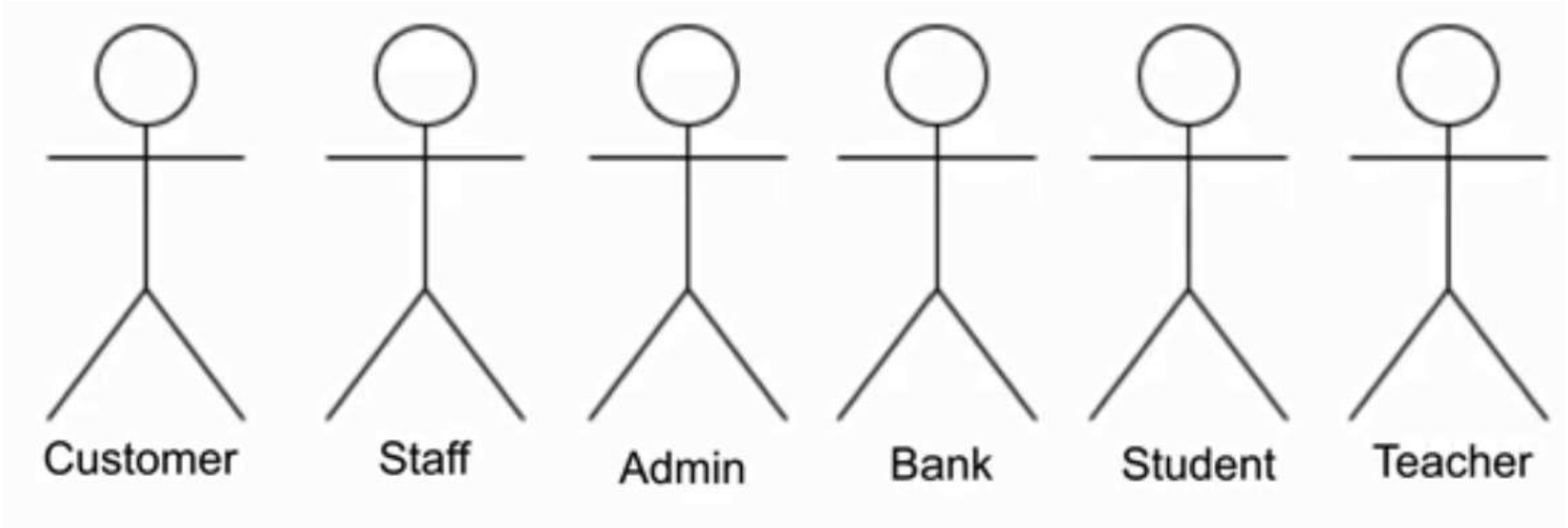


customer, staff, admin, bank, student, teacher, etc.

Actor is representing by the **human stick image**



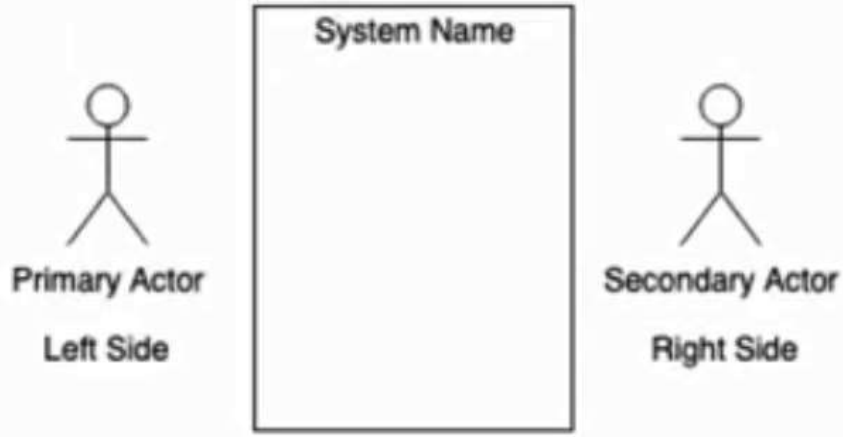
Actor

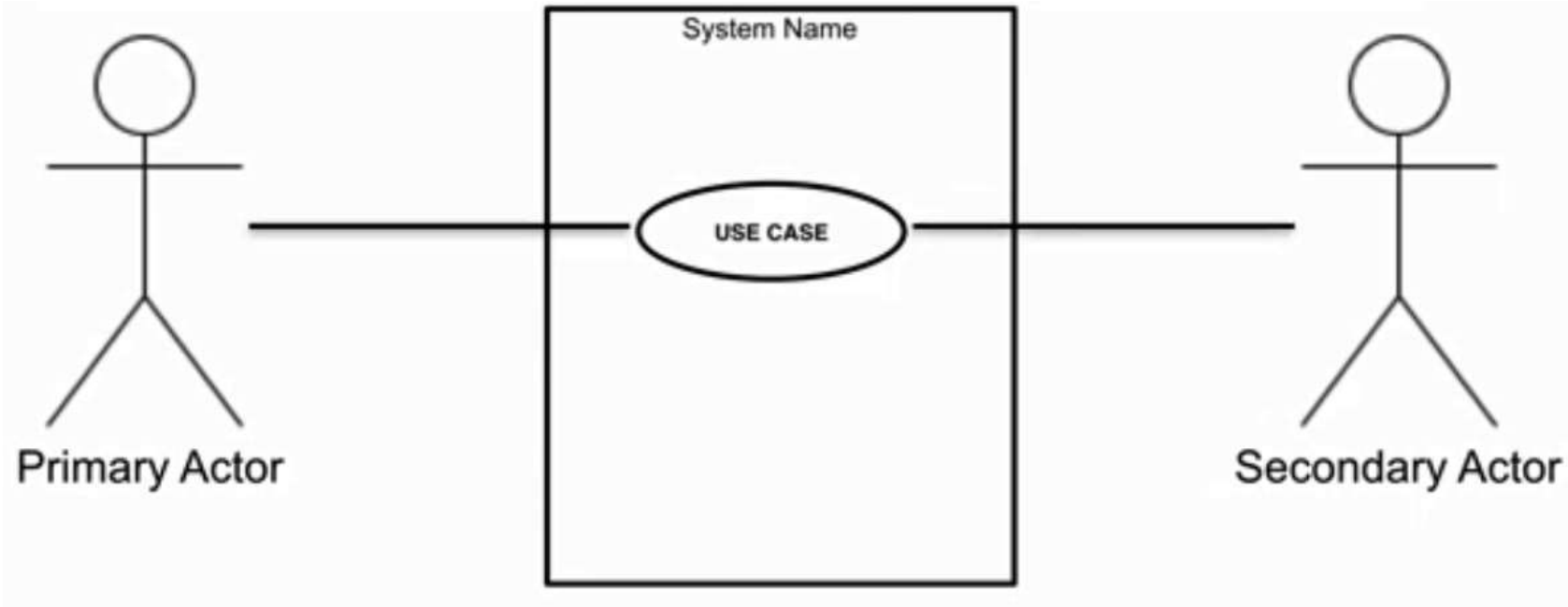

Customer
Staff
Admin
Bank
Student
Teacher

2. Actor

Type of Actors

1. Primary Actor
2. Secondary Actor




System Name
USE CASE
Primary Actor
Secondary Actor

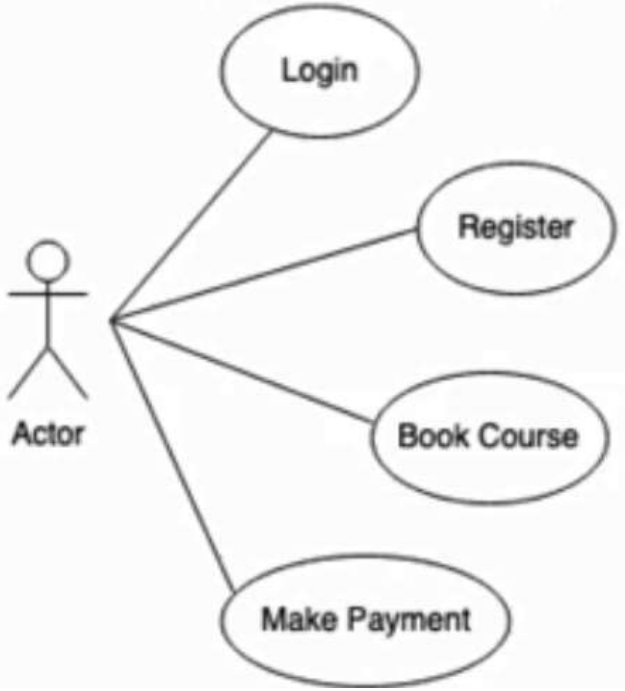
3. Use Cases

All the functionality that the system does is the USE CASE.

login, check balance, transfer fund, make a payment

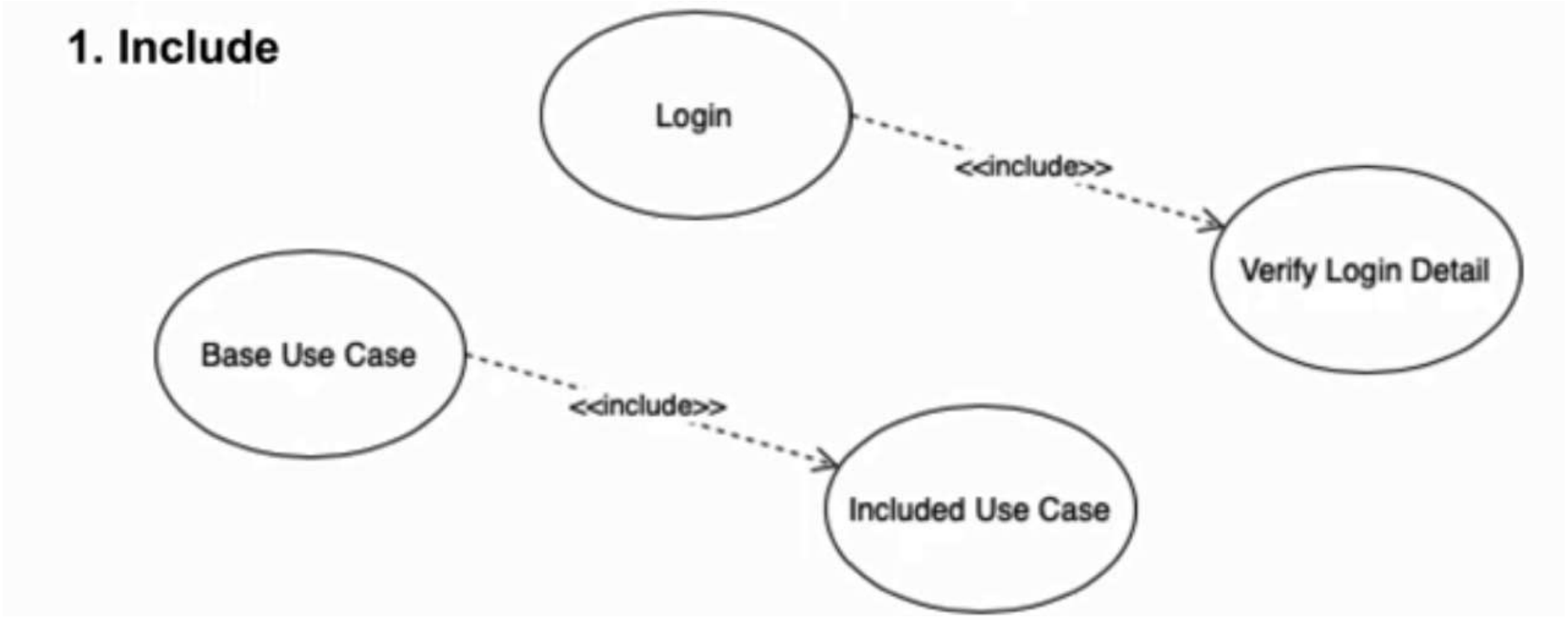
4. Relationship

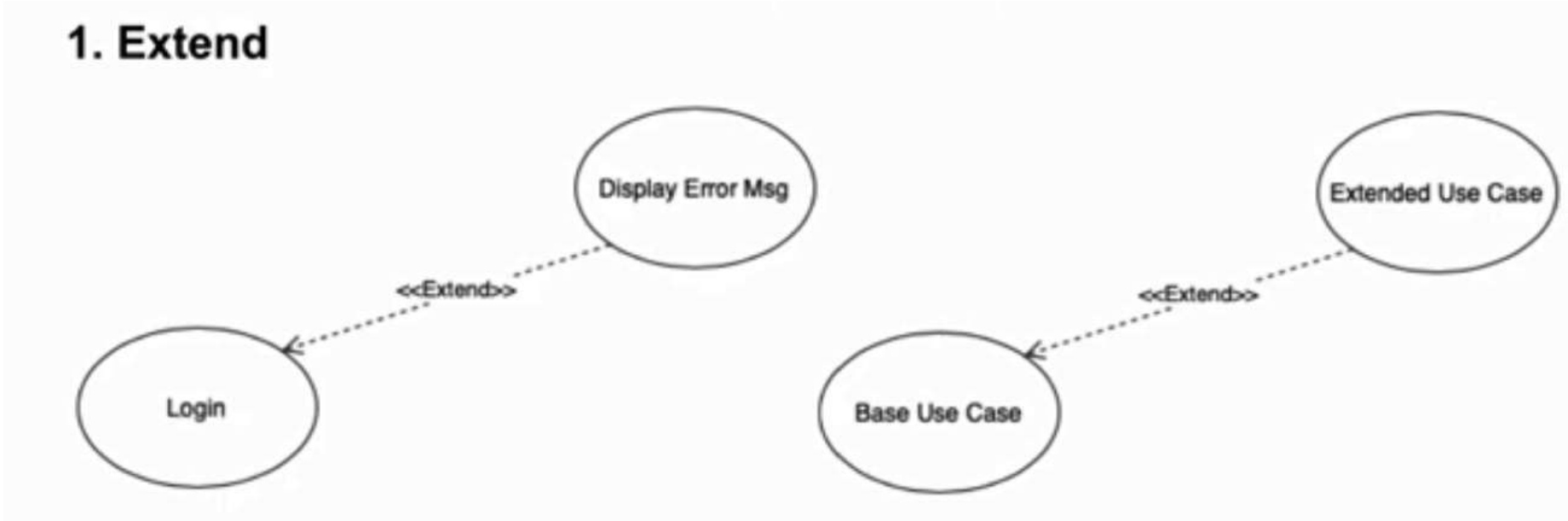
we draw the solid line between the Actor and the Use Case to show the relationship.



Type of Relationship in addition to the Association

1. **Include**
2. **Extend**
3. **Generalization**


1. Include
Login
<<include>>
Verify Login Detail
Base Use Case
<<include>>
Included Use Case


1. Extend
Display Error Msg
<<Extend>>
Login
Extended Use Case
<<Extend>>
Base Use Case

Extend Relationship between Two Use Cases

The extending use case is dependent on the extended (base) use case. In the below diagram the "Calculate Bonus" use case doesn't make much sense without the "Deposit Funds" use case.

The extending use case is usually optional and can be triggered conditionally. In the diagram, you can see that the extending use case is triggered only for deposits over 10,000 or when the age is over 55.

The extended (base) use case must be meaningful on its own. This means it should be independent and must not rely on the behaviour of the extending use case.

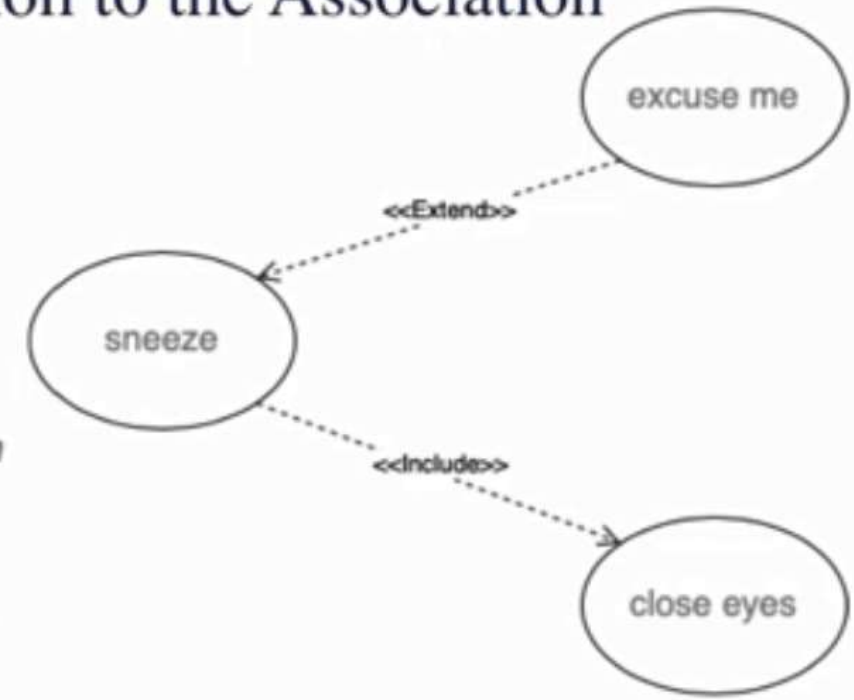
Include Relationship between Two Use Cases

Include relationship show that the behaviour of the included use case is part of the including (base) use case. The main reason for this is to reuse common actions across multiple use cases. In some situations, this is done to simplify complex behaviours. Few things to consider when using the <<include>> relationship.

- **The base use case is incomplete without the included use case.**
- **The included use case is mandatory and not optional.**

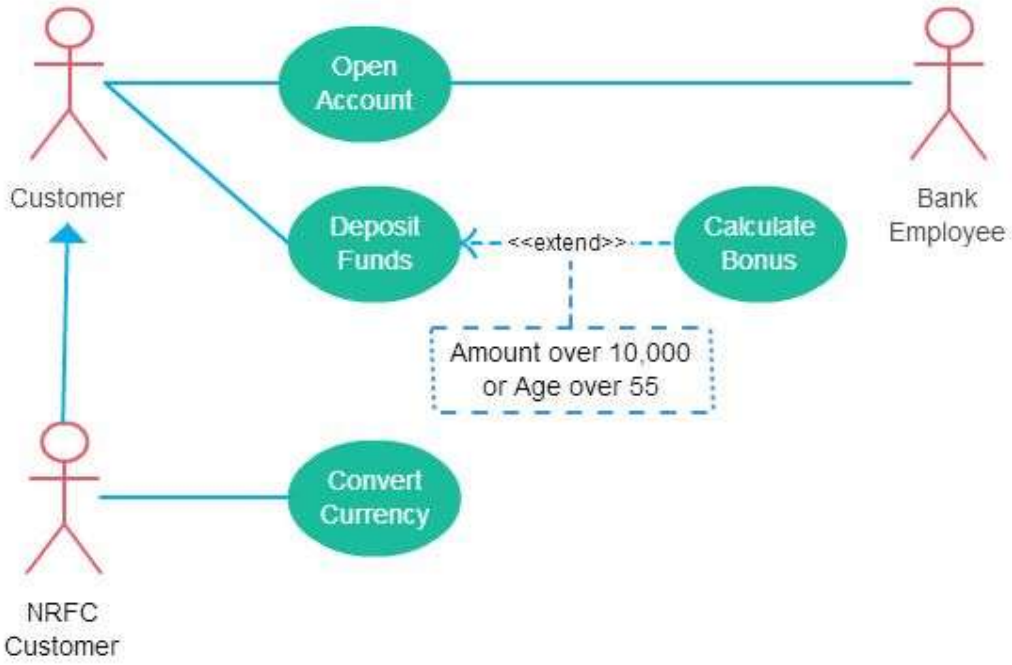
Type of Relationship in addition to the Association

Include & Extend Example

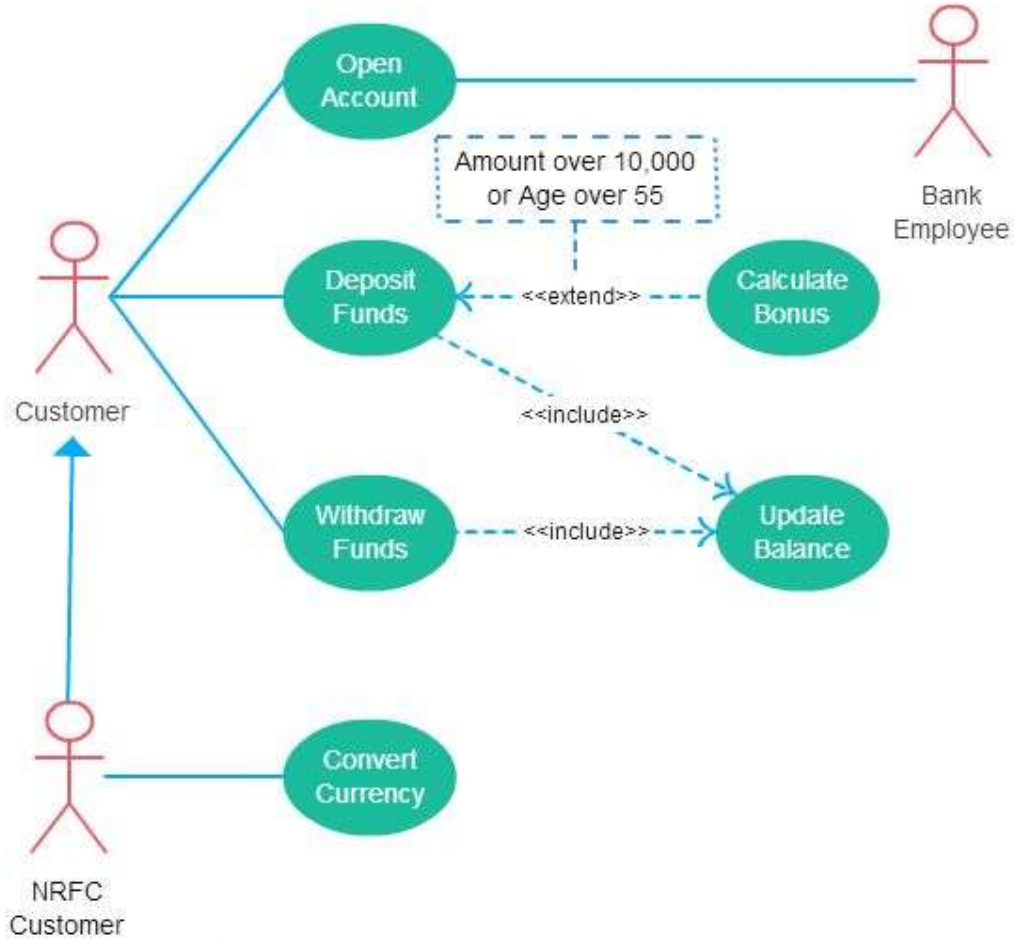


The Simpler your diagram, the Better. When producing a Use Case Model

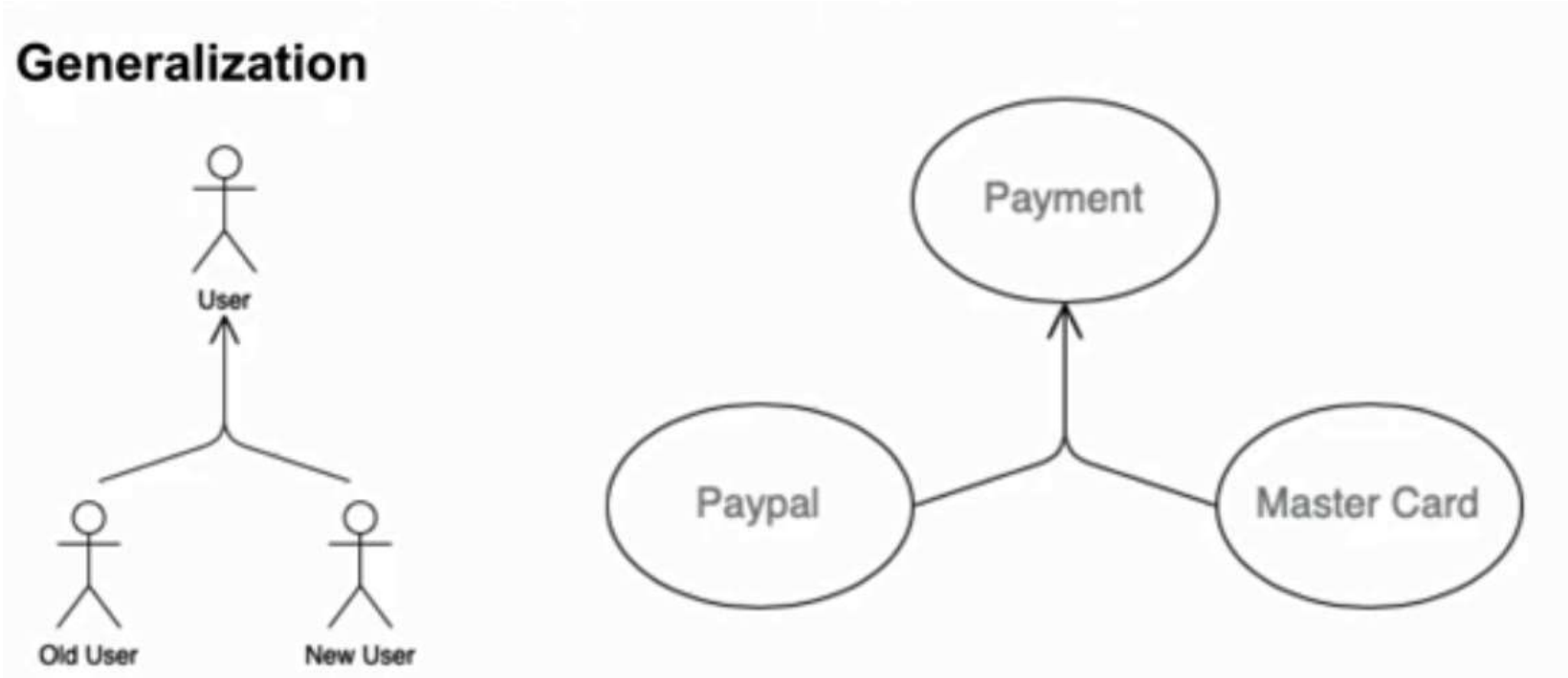
always keep in mind what you are trying to portray and for what purpose.



Extend relationship in use case diagrams

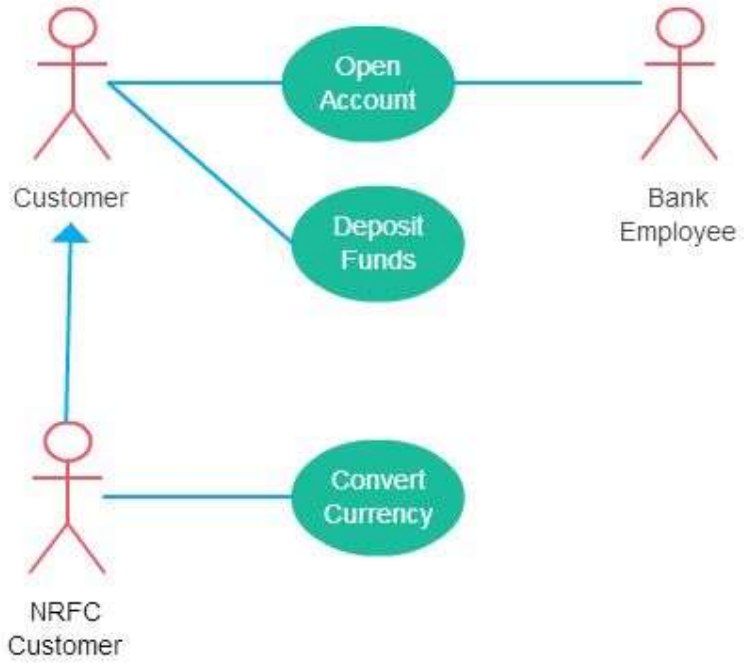


Includes is usually used to model common behavior


Generalization
User
Old User
New User
Payment
Paypal
Master Card

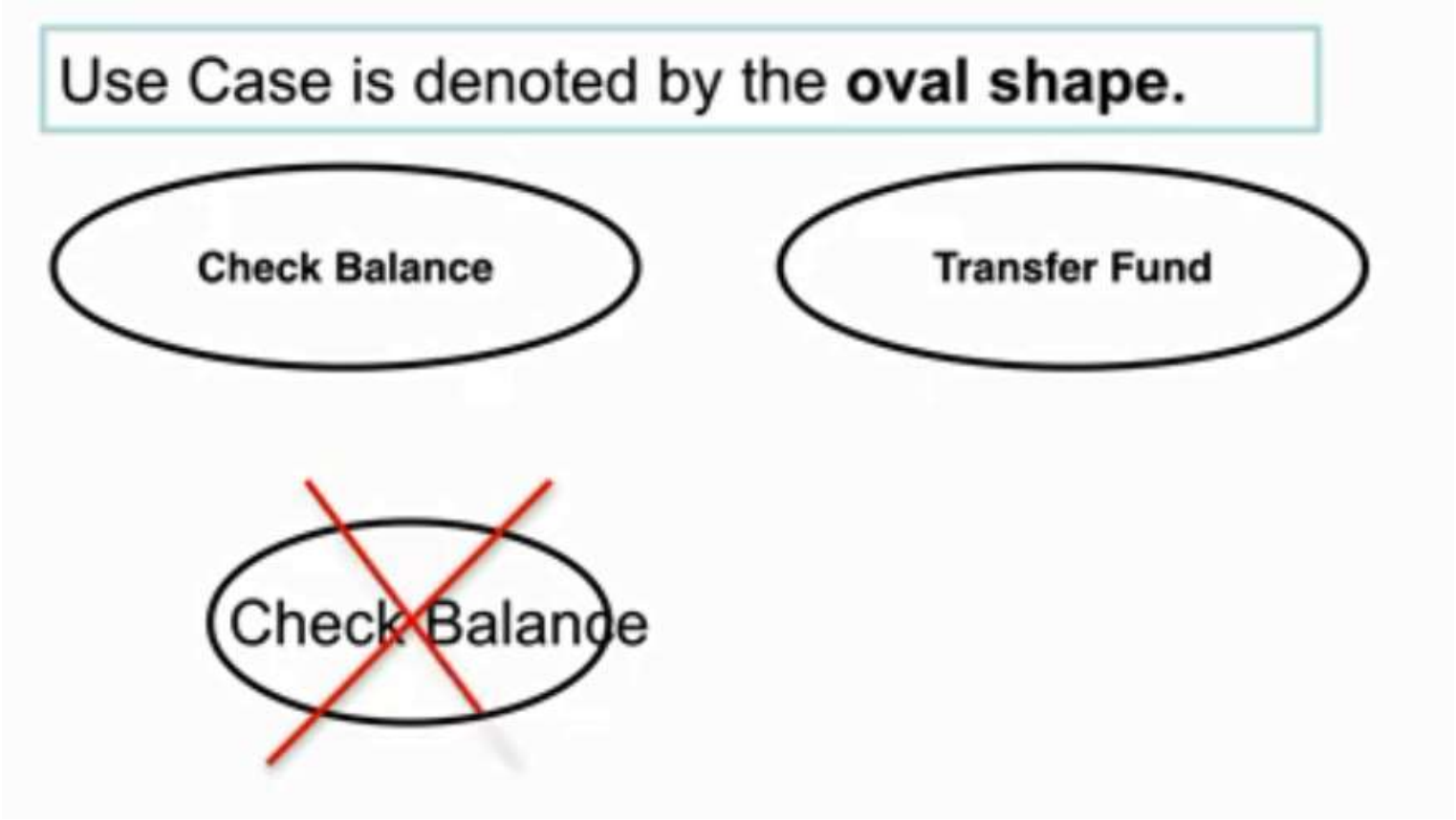
Generalization of an Actor

Generalization of an actor means that one actor can inherit the role of the other actor. The descendant inherits all the use cases of the ancestor. The descendant has one or more use cases that are specific to that role. Let's expand the previous use case diagram to show the generalization of an actor.



A generalized actor in an use case diagram

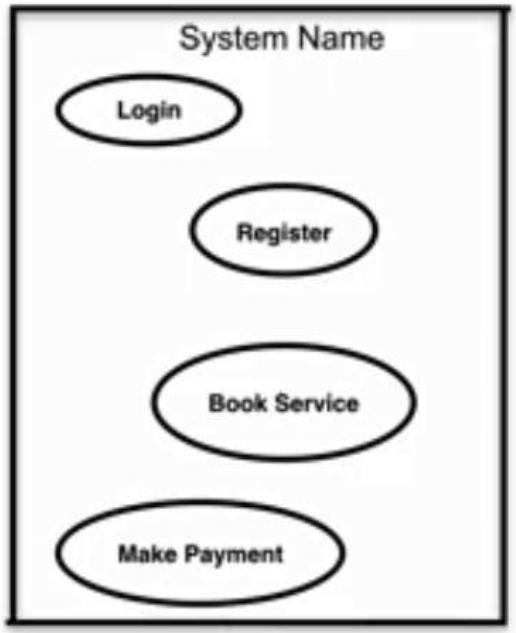
Mention only the Functional Requirements of the system.


Use Case is denoted by the **oval shape.**
Check Balance
Transfer Fund
Check Balance

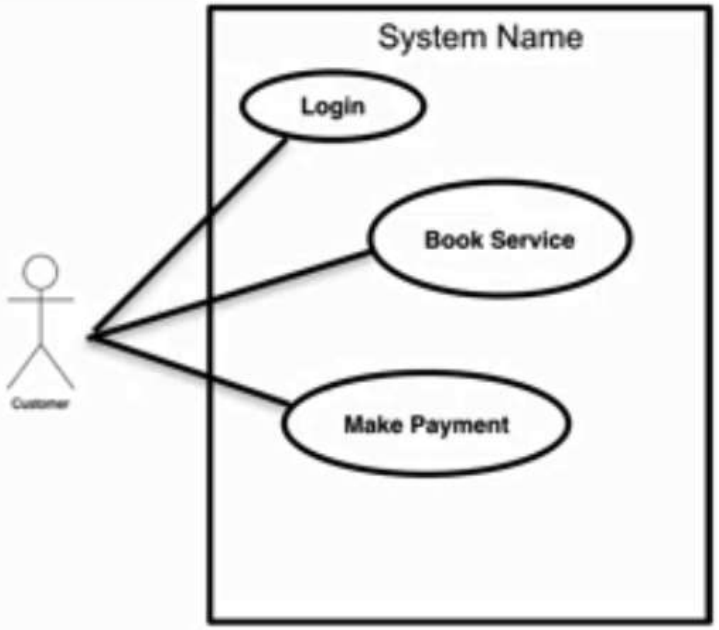
Use Case	Actor
Check Balance	User, Staff
Transfer Fund	User
Make Payment	User
Display Error Msg	
Verify Log In Details	

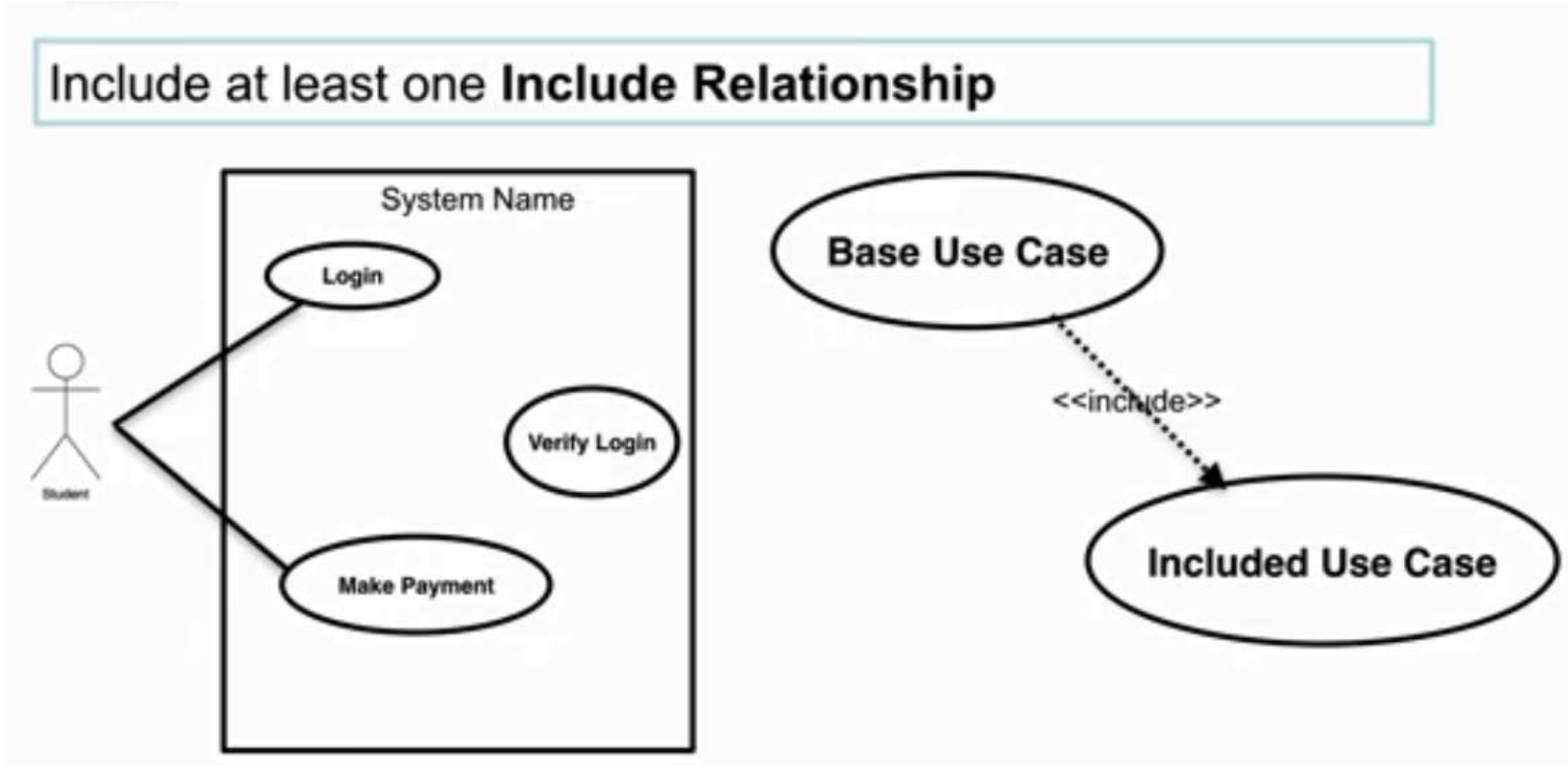
An active verb
&
A noun phrase

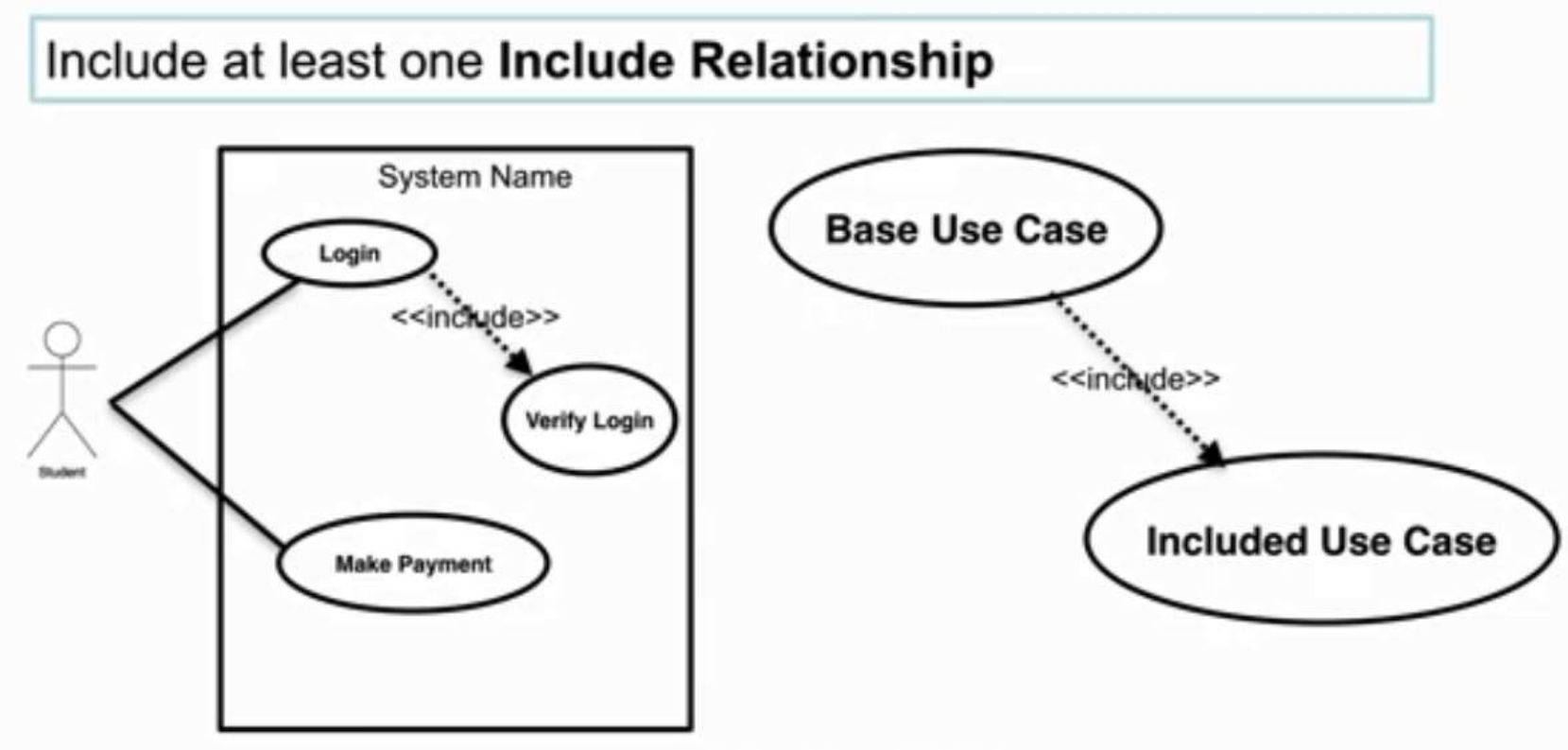
All Use Cases should be **inside the system**.

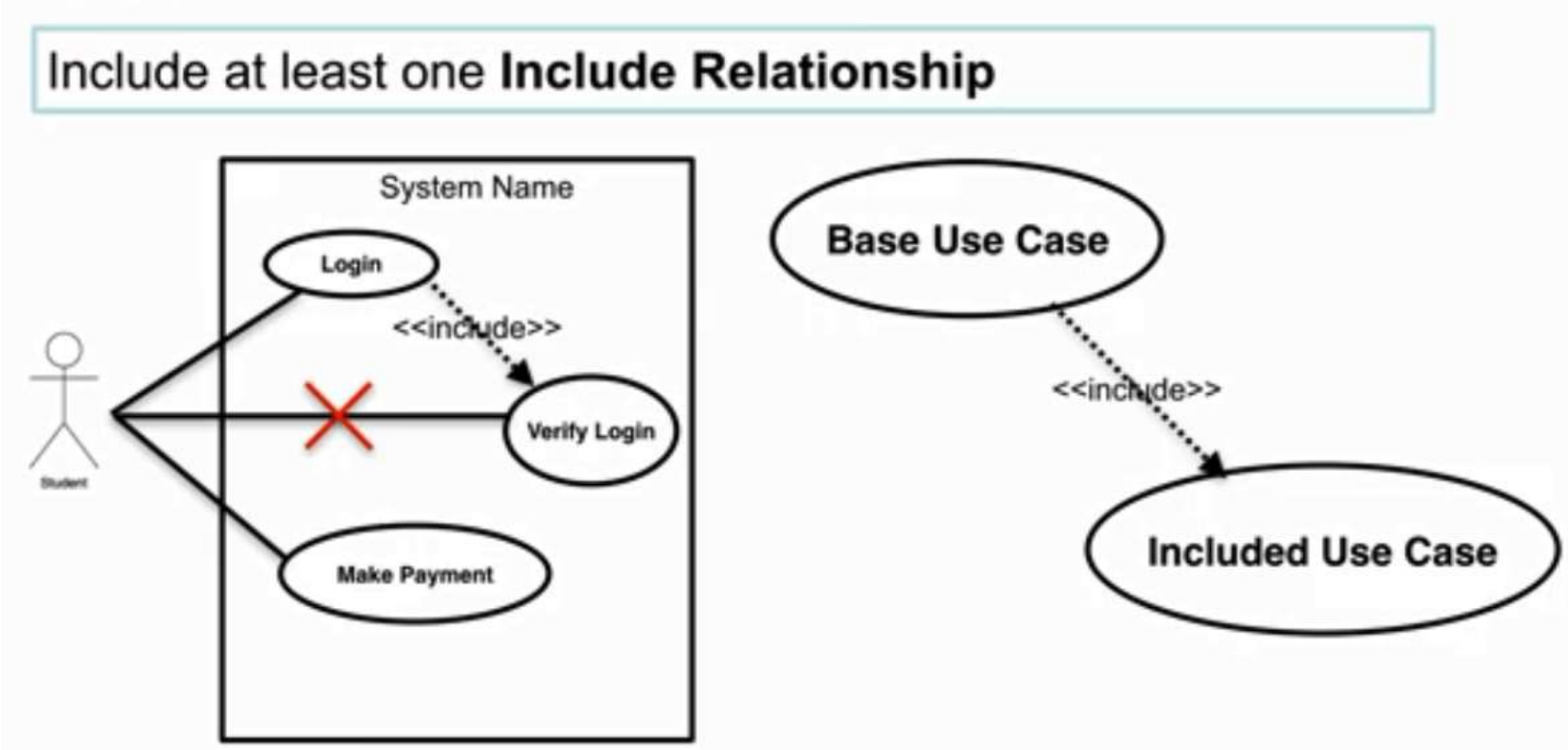


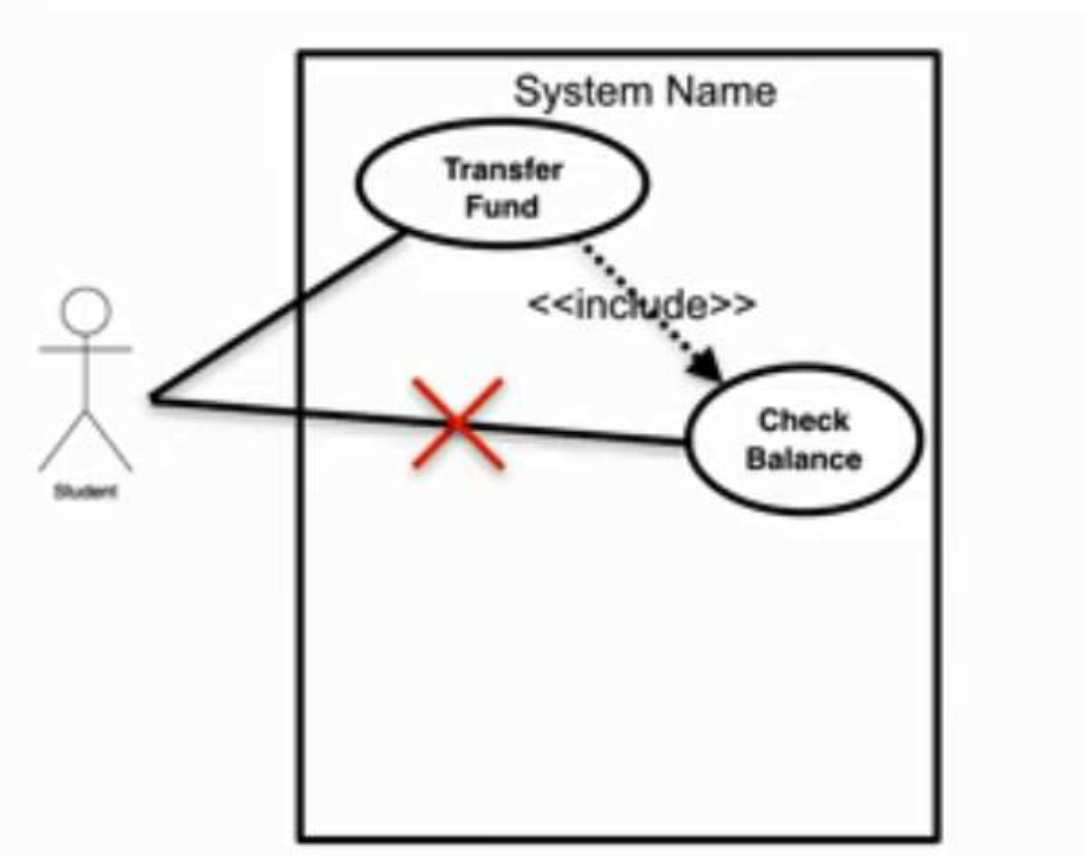
Put the Use Cases in a logical order

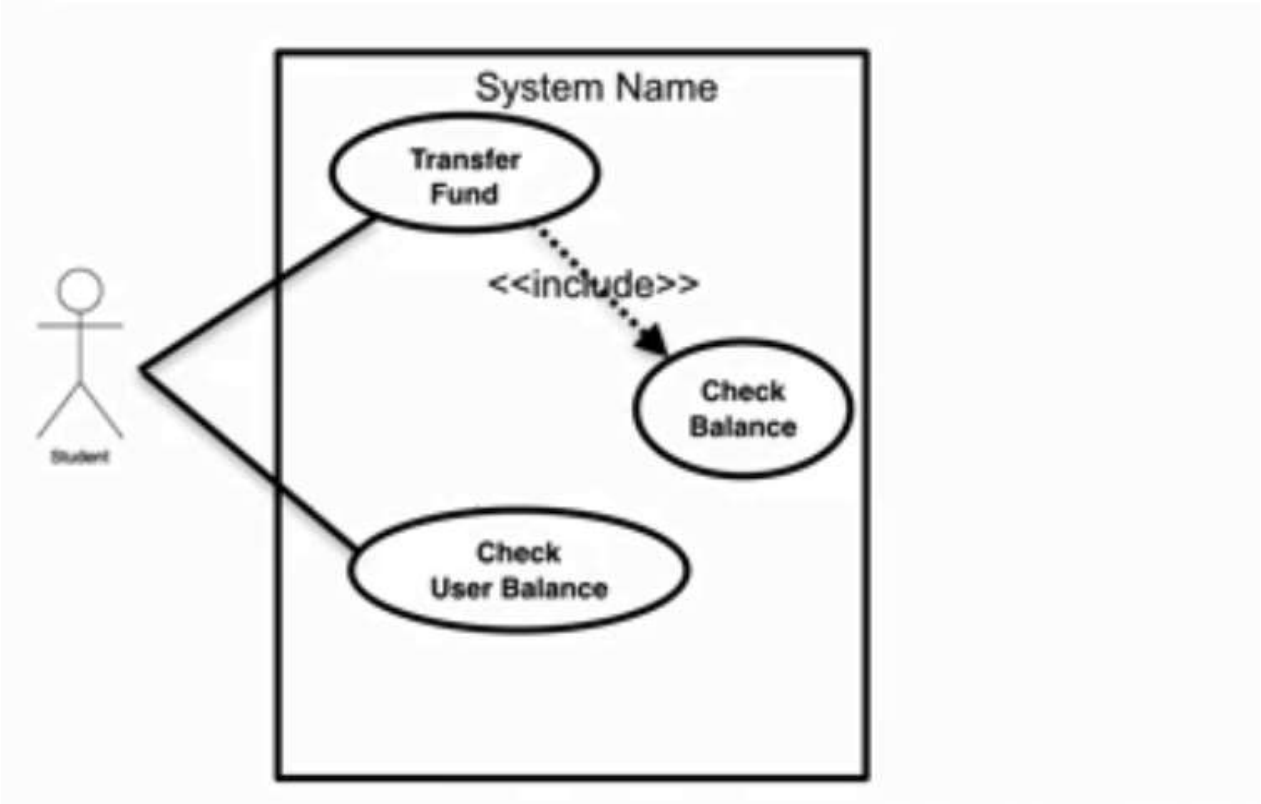


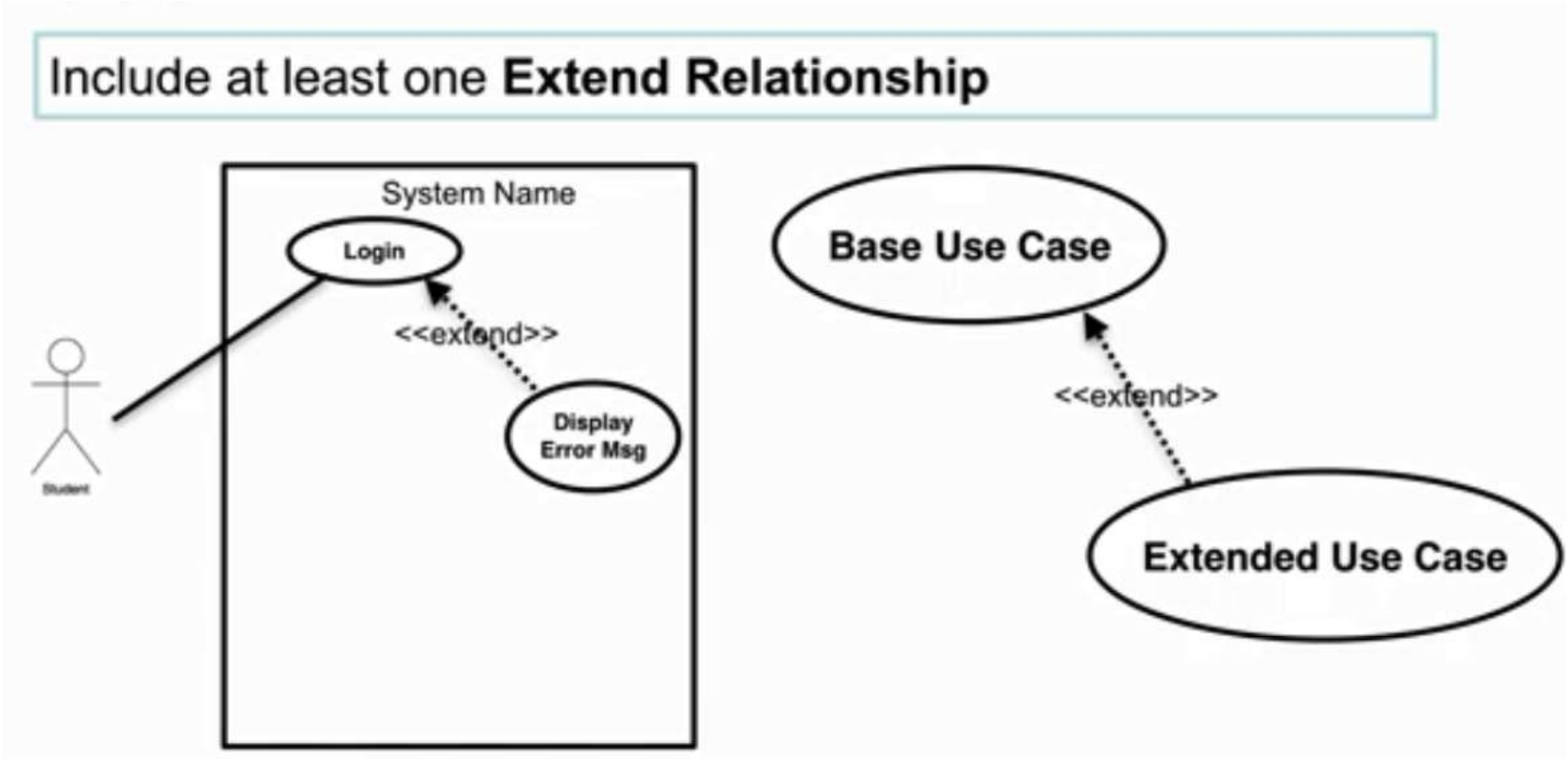

Include at least one **Include Relationship**
System Name
Login
Verify Login
Make Payment
Student
Base Use Case
<<include>>
Included Use Case

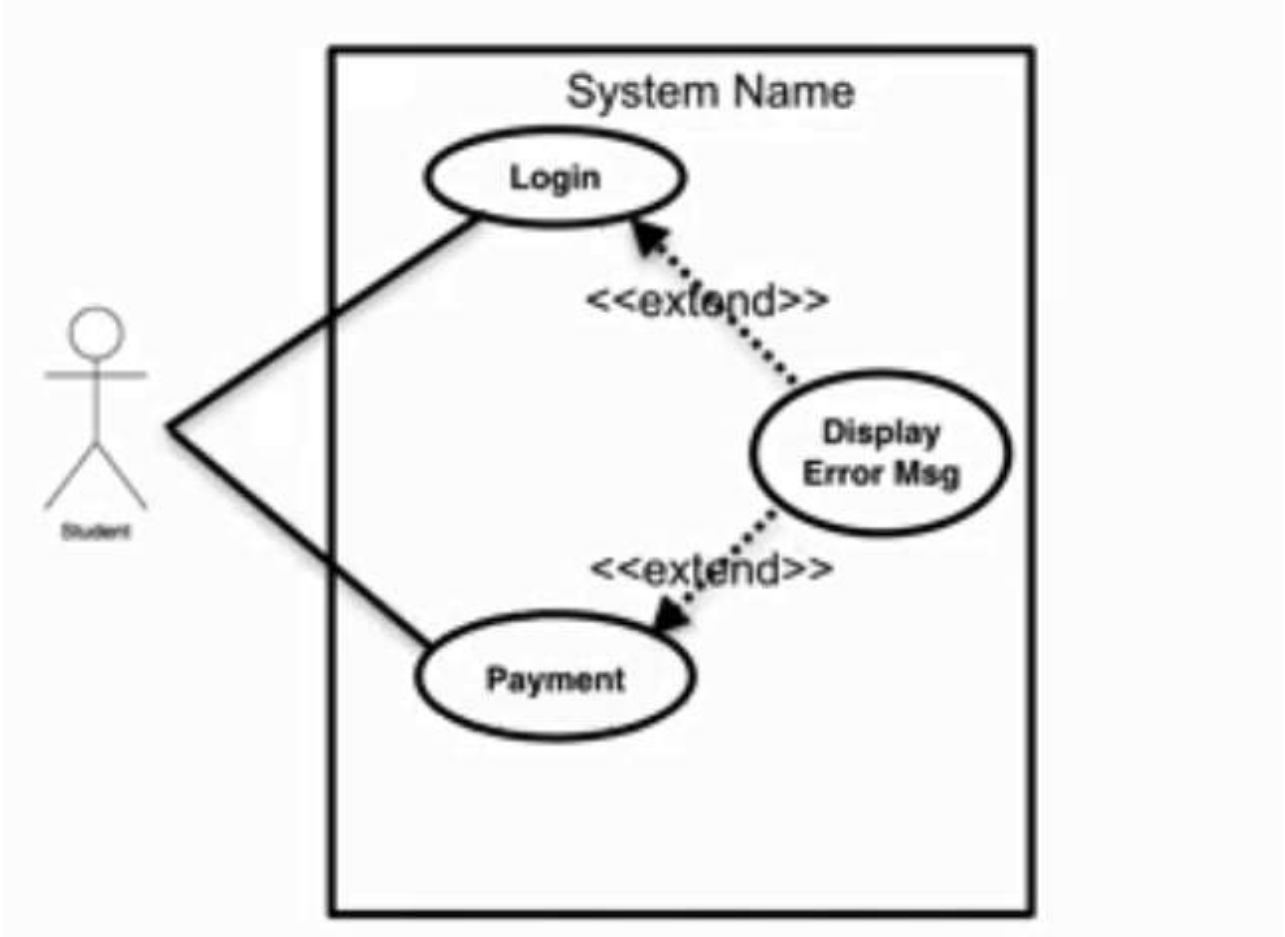

Include at least one **Include Relationship**
System Name
Login
<<include>>
Verify Login
Make Payment
Student
Base Use Case
<<include>>
Included Use Case

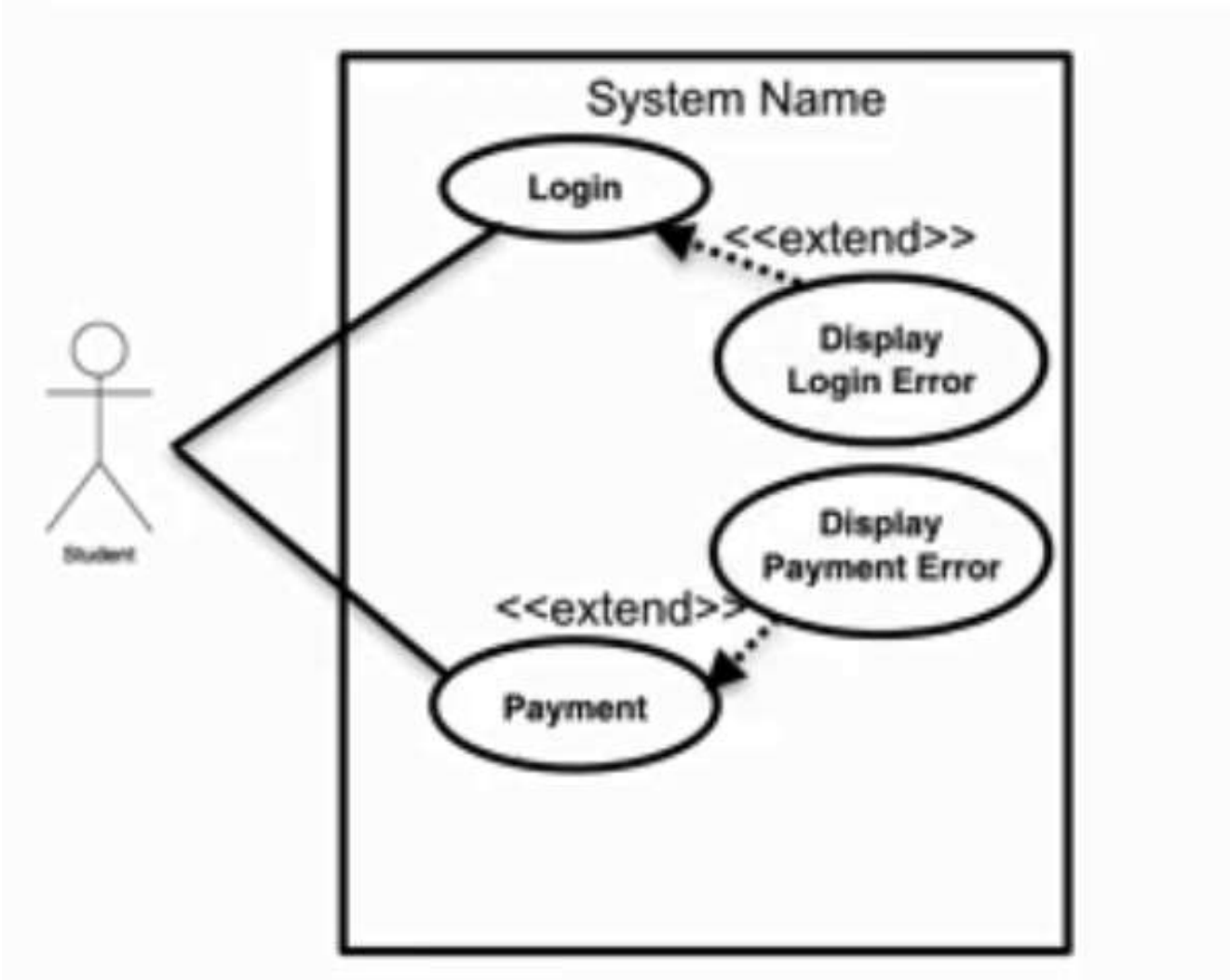

Include at least one **Include Relationship**
System Name
Login
<<include>>
Verify Login
Make Payment
Student
Base Use Case
<<include>>
Included Use Case


System Name
Transfer Fund
<<include>>
Check Balance
Student

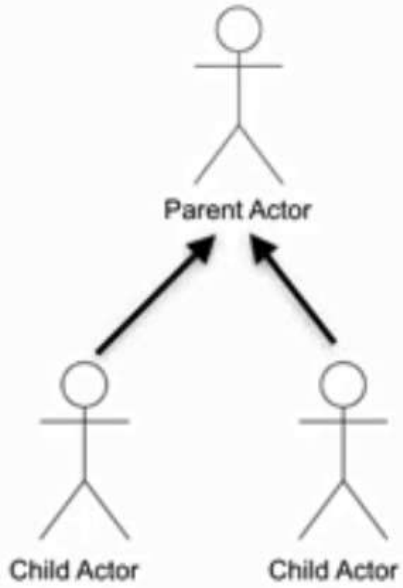

System Name
Transfer Fund
<<include>>
Check Balance
Check User Balance
Student

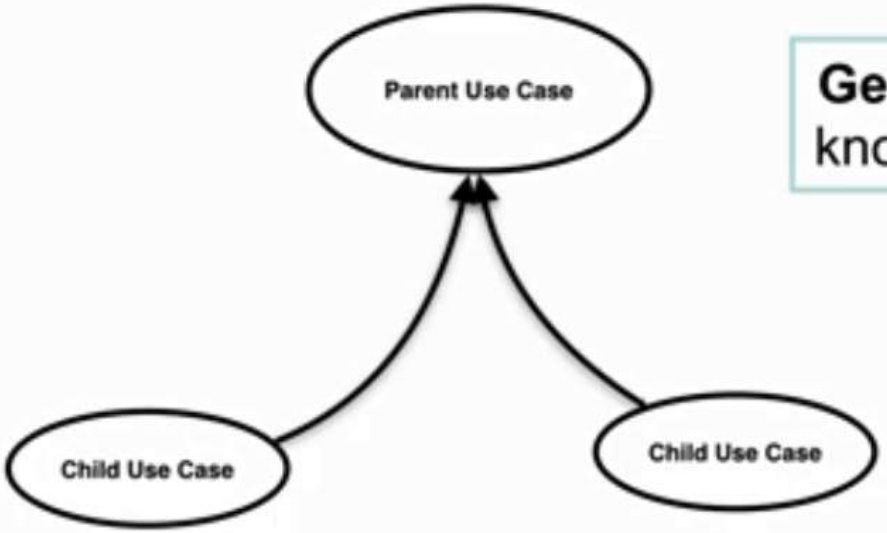

Include at least one **Extend Relationship**
System Name
Login
<<extend>>
Display
Error Msg
Student
Base Use Case
<<extend>>
Extended Use Case


System Name
Login
<<extend>>
Display
Error Msg
<<extend>>
Payment
Student

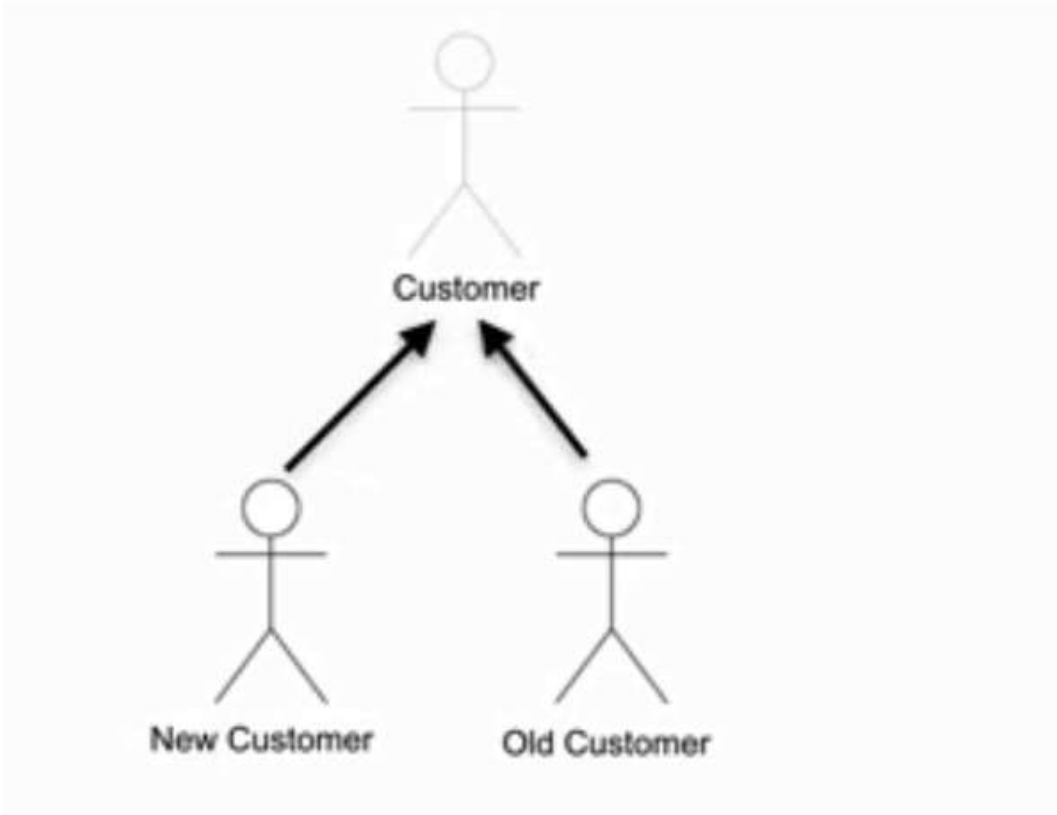

System Name
Login
<<extend>>
Display
Login Error
Display
Payment Error
<<extend>>
Payment
Student

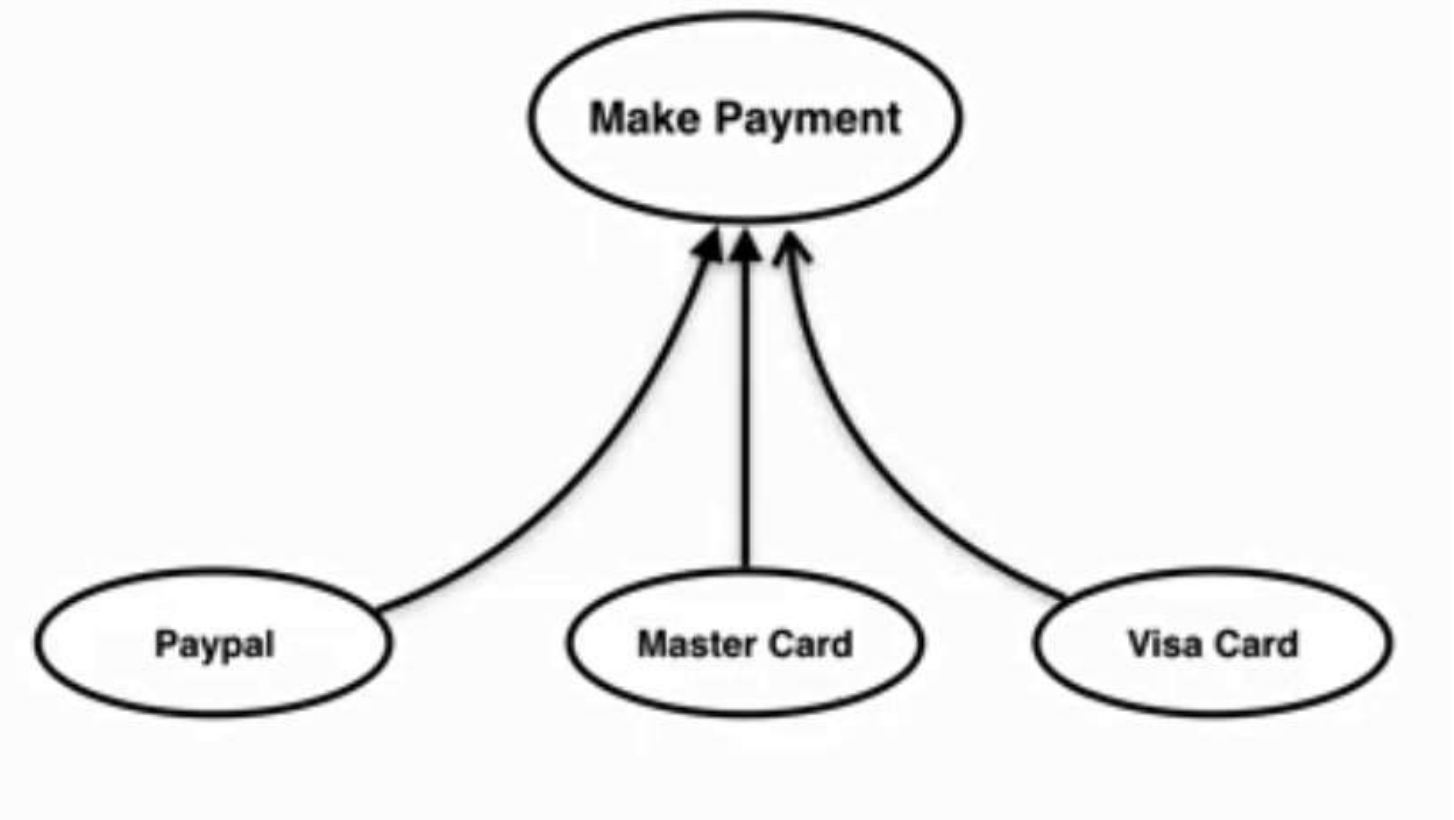
Include at least one **Generalization**





Generalization is also known as **Inheritance.**


Customer
New Customer
Old Customer


Make Payment
Paypal
Master Card
Visa Card

Types of Use Case Descriptions

1. High Level Use Case Description

A short non-detailed description of each required process

Use Case:	Enroll Student
Actor:	New Student
Description:	A new student provides personal details and his/her choice of course to the system. He / She pays fees and receives confirmation of enrollment.

Types of Use Case Descriptions

1. Expanded Use Case Description

Actor Action	System Response
1. The new Student uses the on-screen enrollment form to input personal details	2. Presents a menu of the subject areas in the college
3. Clicks on the chosen subject area	4. Presents a menu of the courses in the subject area
	6. displays the course fee
5 Clicks on the chosen course	7. requests card details
8. pays by credit/debit card	9. provides a printable course joining form including a student id number
10. prints joining form	

Association between Actor and Use Case

- An actor must be associated with at least one use case.
- An actor can be associated with multiple use cases.
- Multiple actors can be associated with a single use case.

NEXT SESSION: CASE STUDY

Credits:
master2teach.com