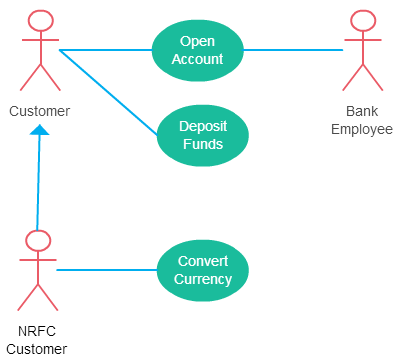
Relationships

# Relationships between actors

## Generalization relationship between actors

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.

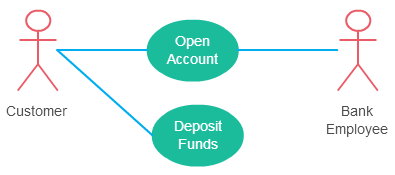


## Association relationship between actors

- 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.



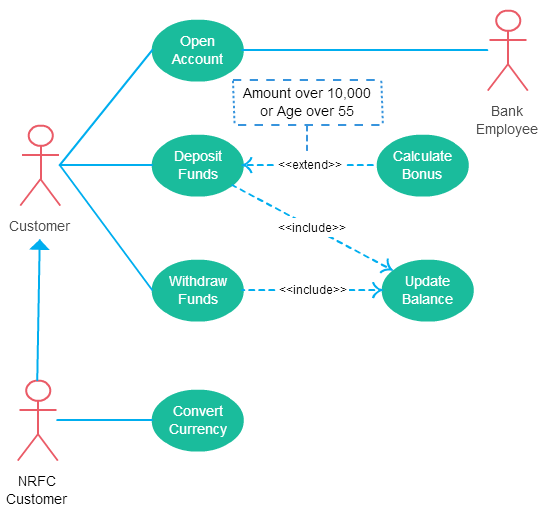
# Relationships between use cases

## Include relationship between use cases

Include relationship show that the behavior of the included use case is part of the including (base) use case. The main reason for this is to reuse the common actions across multiple use cases. In some situations, this is done to simplify complex behaviors. 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.



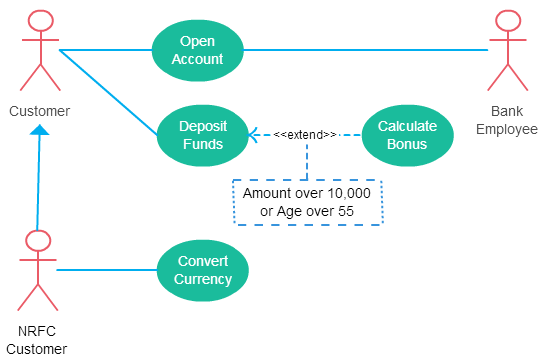
## Extend relationship between use cases

It extends the base use case and adds more functionality to the system. There are few things to consider when using the <<extend>> relationship.

- 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 behavior of the extending use case.



## Generalization relationship between use cases

This is similar to the generalization of an actor. The behavior of the ancestor is inherited by the descendant. This is used when there is common behavior between two use cases and also specialized behavior specific to each use case.

For example, in the previous banking example, there might be a use case called “Pay Bills”. This can be generalized to “Pay by Credit Card”, “Pay by Bank Balance” etc.