USE CASE DIAGRAM

System Analysis & Design

Istanbul Arel University

Spring 2023

Use case

- A use-case is a step-by-step description of how a user will use the system to accomplish business goals.
- It is an *example behavior* of the system.
- Written from actor's point of view, not the system's.
- Use cases capture **functional requirements** of a system.

Use case

- A major process that the system will perform that benefits the actor(s).
- Use cases help us understand and clarify the users' required interactions with the system.
- Also used to discover user and functional requirements.
- Use case represents how a system interacts with its environment by illustrating the activities that are performed by the users of the system and the system's responses.
- The goal is to create a set of use cases that describe all the tasks that the users need to perform with the system.

Use case

- A use case depicts a set of activities performed to produce some output.
- Each use case describes how an external user (actor) *triggers* an *event* which the system must respond.
- Use cases are created when they are likely to help to understand the situation better. For very simple processes which are explained in requirements definition, we don't often create use cases.

Use case

- A requirements analysis concept
- A *case* of a *use* of the system/product
- Describes the system's actions from the point of view of a user.
- It **tells a story**
  - A sequence of events involving
  - Interactions of a user with the system
- There are multiple ways to show use cases.
  - Textual or tabular
  - User stories
  - Diagrams

Benefits of use cases

- Establish an understanding between the customer and the system developers of the requirements.
- Alert developers of problematic situations, error cases to test.
- Capture a level of functionality to plan around.

Use case descriptions

- We draw the overall list of system's use cases as high-level diagrams by using;
    - Actors as *stick men* with their names (noun)
    - Use cases as *ellipses* with their names (verbs)
    - We use *line associations* to connect an actor to a use case.
- Used to help in structuring systems
- Use case diagrams *should not* be too complex.

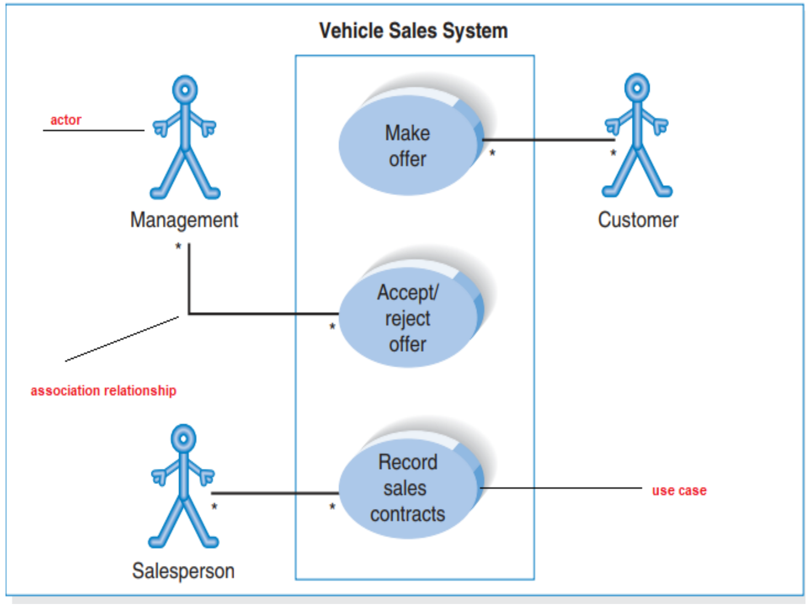
Use case diagram

- It is built in early stages of development.
- Purpose is:
  - Specify the context of the system
  - Capture the requirements of a system
  - Validate a systems architecture
  - Developed by analysts and domain experts

Use case diagram

- Present the functionality in different ways and each view has a different purpose.
- A use case may represent several *paths* that a user can take while interacting with the system.
  - Each part is called a *scenario*.
- Drawn early on SDLC, when the analyst is gathering and defining requirements for the system.
  - It provides a simple, straightforward way of communicating to the users what the system will do.

Use case diagram



- An actor is a person or a system that derives benefit from and is *external* to the system.
  - Labeled with the role.
  - Placed outside the system boundary.
- Use case represents a major piece of system functionality.
  - Can extend or use another use case.

Actor

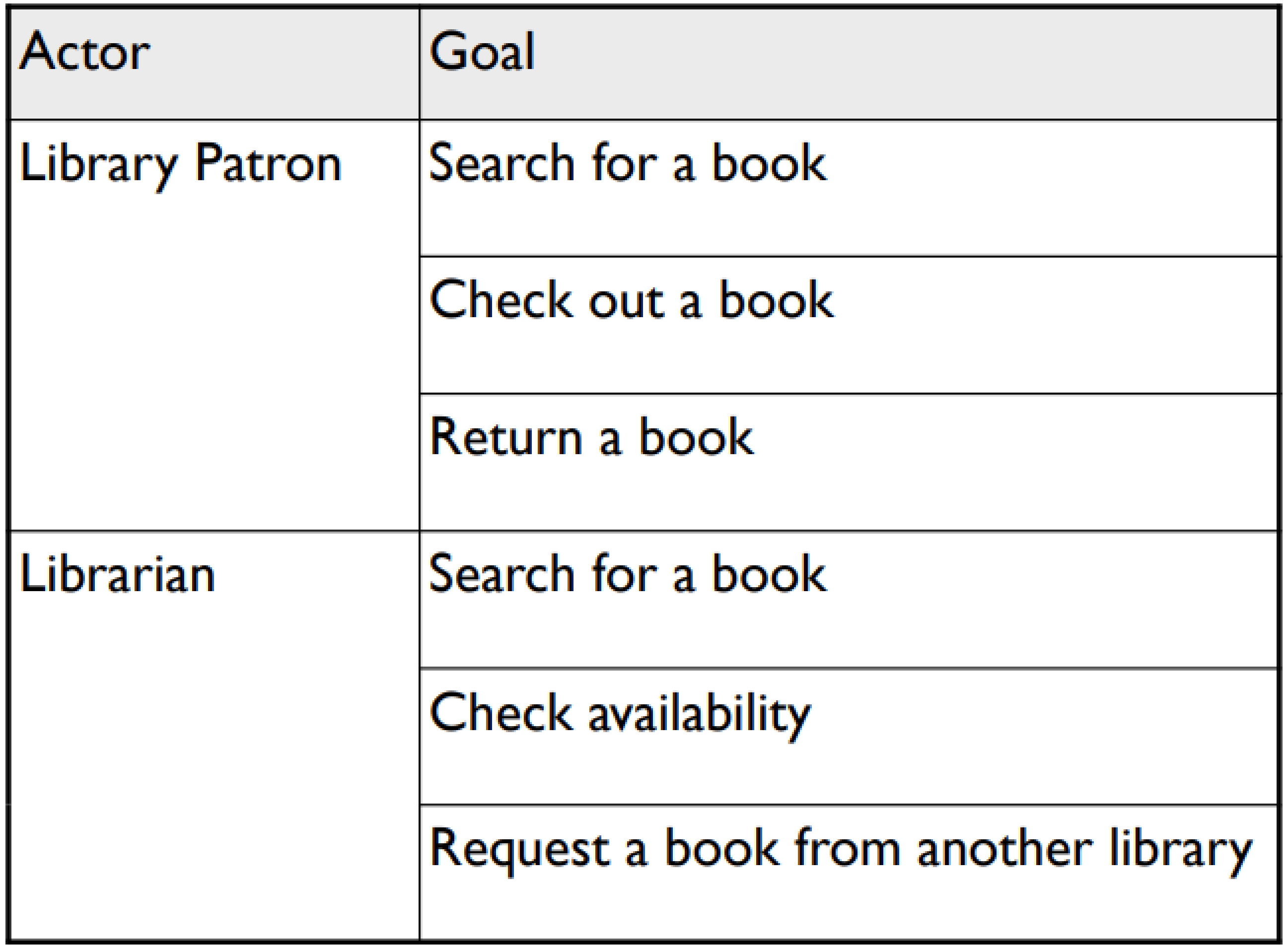
- Stick figures are called actors.
- It is an external entity. Person or another system which derives value from the system.
  - It is not a specific user but a **role** that a user can play while interacting with the system.
  - Actors are **external** to the system and initiate a use case.
- Let's say we are modeling a doctor's office appoinment system. Here, data-entry clerk would not be an actor because that is actually **in** the scope of the system (not external). But a *customer* will be an actor.
- Actors **triggers** use cases.
- They have *responsibility* toward the system (inputs) and *expectations* from the system (output).

Use case descriptions > Actor

- There are three types of actors:
  - **primary**: a user swhose goals are fulfilled by the system. (define user goals)
  - **supporting**: provides a service to the system. (clarify external interfaces and protocols)
  - **offstage**: has an interest in the behavior but not primary or supporting (e.g. government) (ensure all interests are identified and satisfied)

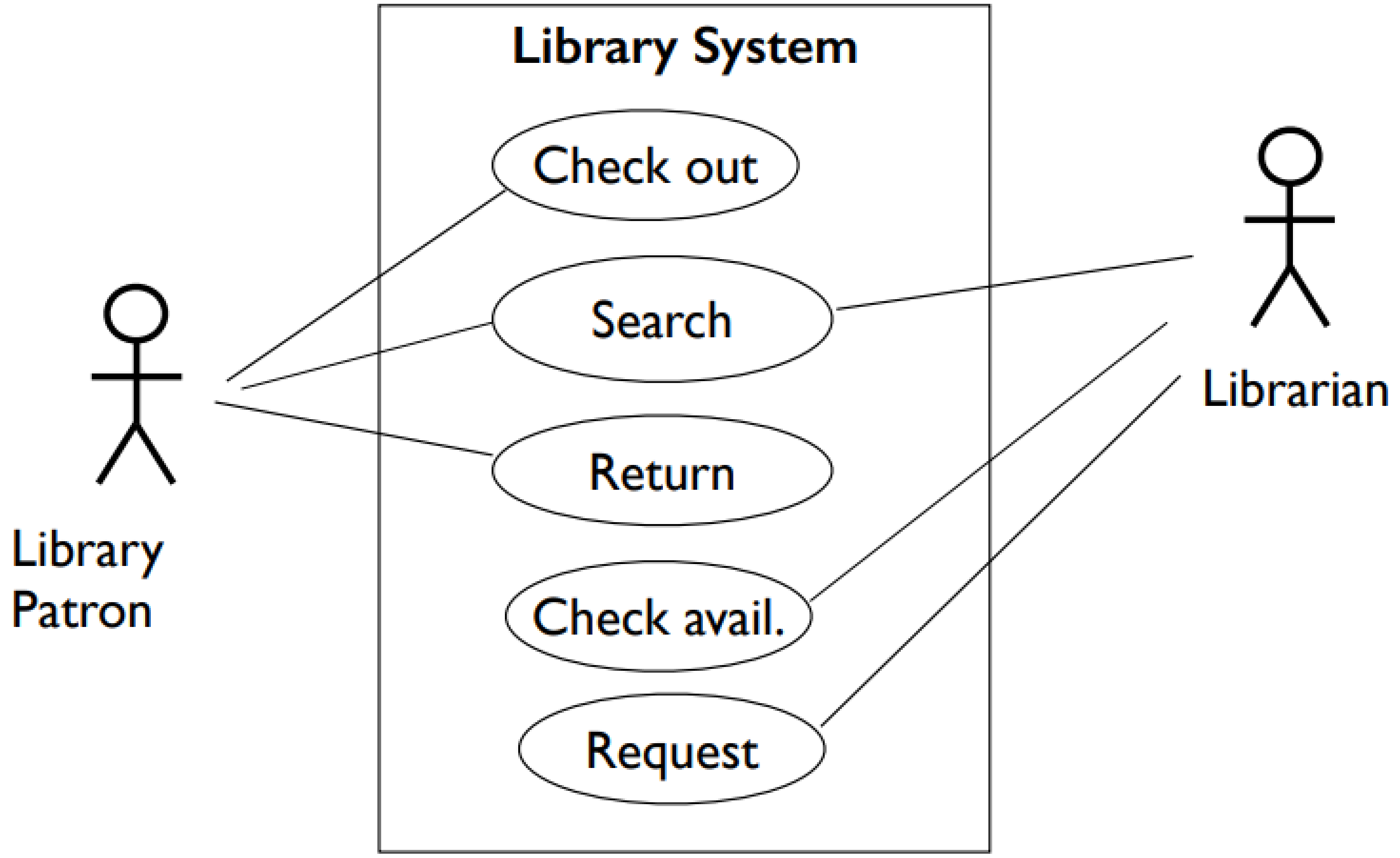
Actor-goal lists

- It can be useful to create these tables before we create the use-case diagram.



Actor	Goal
Library Patron	Search for a book
	Check out a book
	Return a book
Librarian	Search for a book
	Check availability
	Request a book from another library

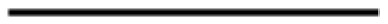
Actor-goal lists



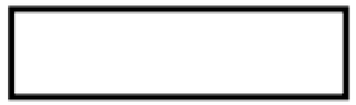
Use case diagram

- Text is still essential to the system.
  - Use case diagrams are not always enough to understand.
  - That is why there are tables which capture the full information, but diagrams give an idea. So, if there is a problem we refer to those tables.

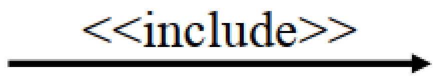
Use case diagrams > Other details



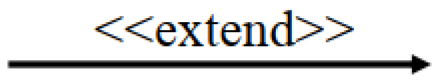
Connection between Actor and Use Case



Boundary of system



Include relationship between Use Cases (one UC must call another; e.g., Login UC includes User Authentication UC)



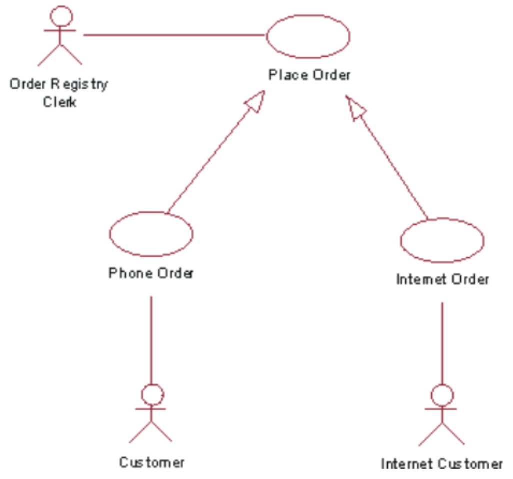
Extend relationship between Use Cases (one UC calls Another under certain condition; think of if-then decision points)

Linking use cases

- Association
- Generalization
  - An element (child) *is based on* another element (parent).
- Include
  - One use case includes the *functionality* of another.
- Extend
  - One use case extends the *behavior* of another.

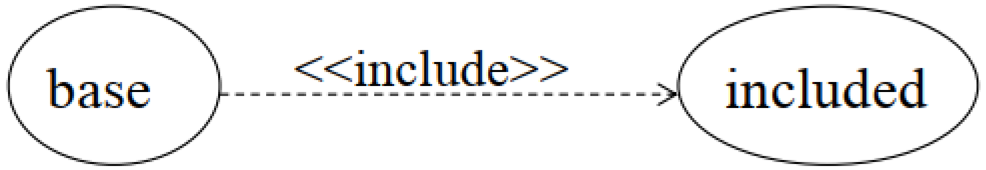
Generalization

- Child use case inherits the behavior and meaning of the parent use case.
- Child may *add* or *override* the behavior of its parent.



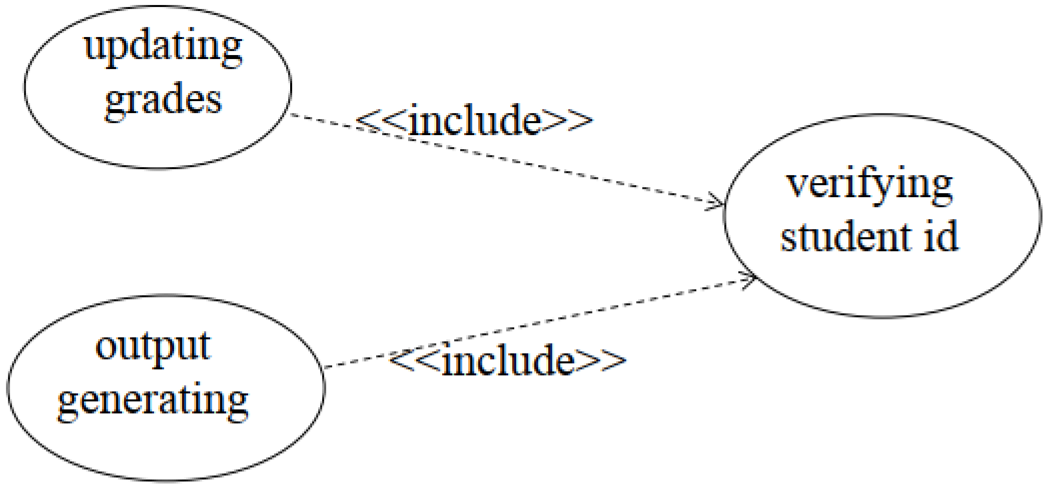
- Order registry clerk can instantiate *Place Order*. Place order can also be specialized by the *use cases* Phone order and Internet order.

Generalization > Include



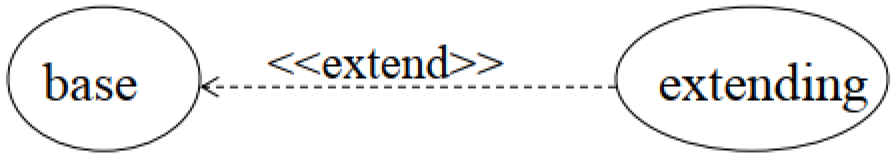
- Here, the base use case explicitly *incorporates* the behavior of another use case.
- The *included* use case never stands alone. It only occurs as a part of some larger base that includes it.

Generalization > Include



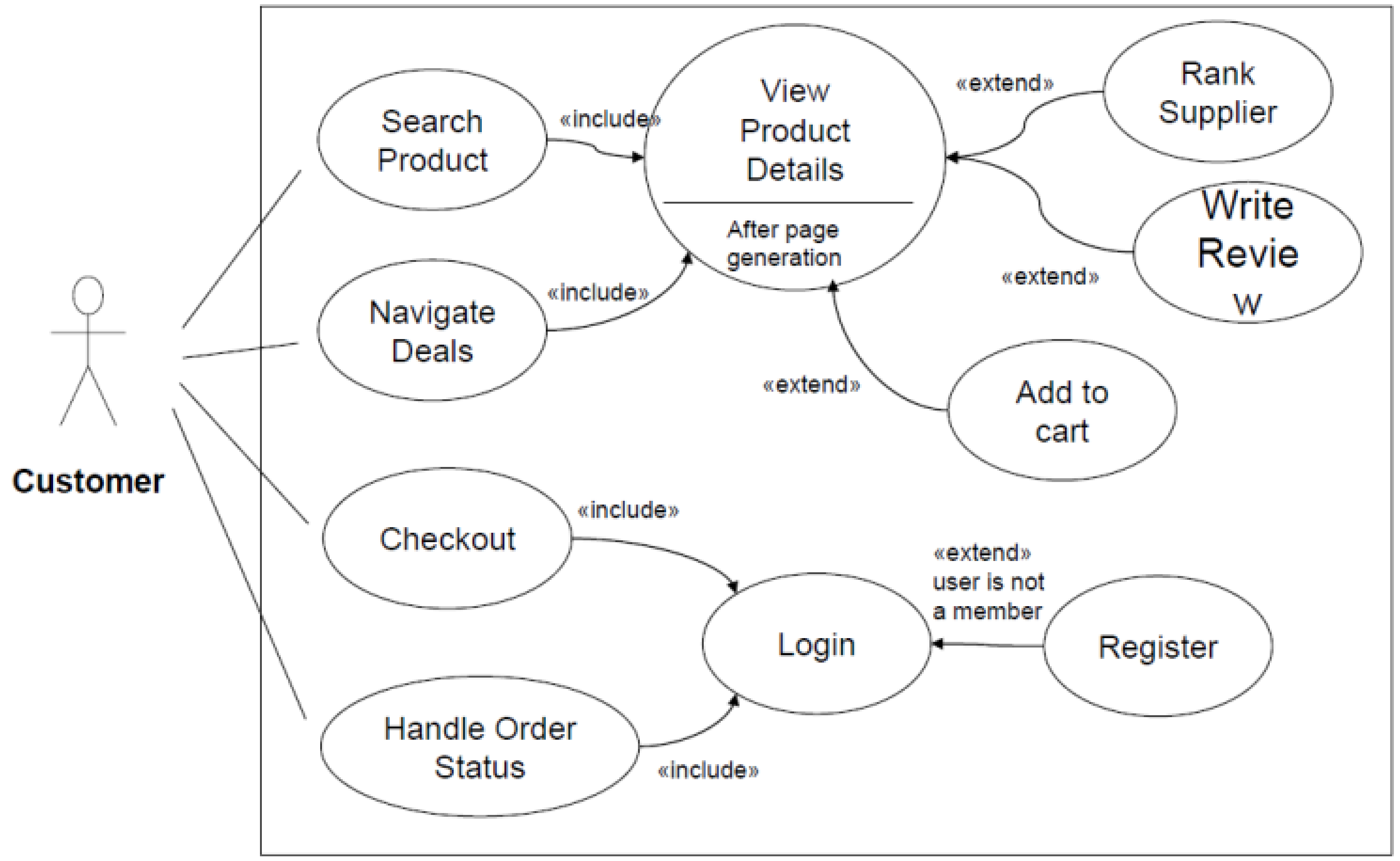
- Also enables us to avoid describing the same flow of events several times by putting the common behavior in a use case of its own.
- If there is an *include relationship*, it means that a standard case is linked to a **mandatory** use case.
- For example, to *Authorize Car Loan* (std), a clerk must run *Check Client's Credit History* (include).

Generalization > Extend

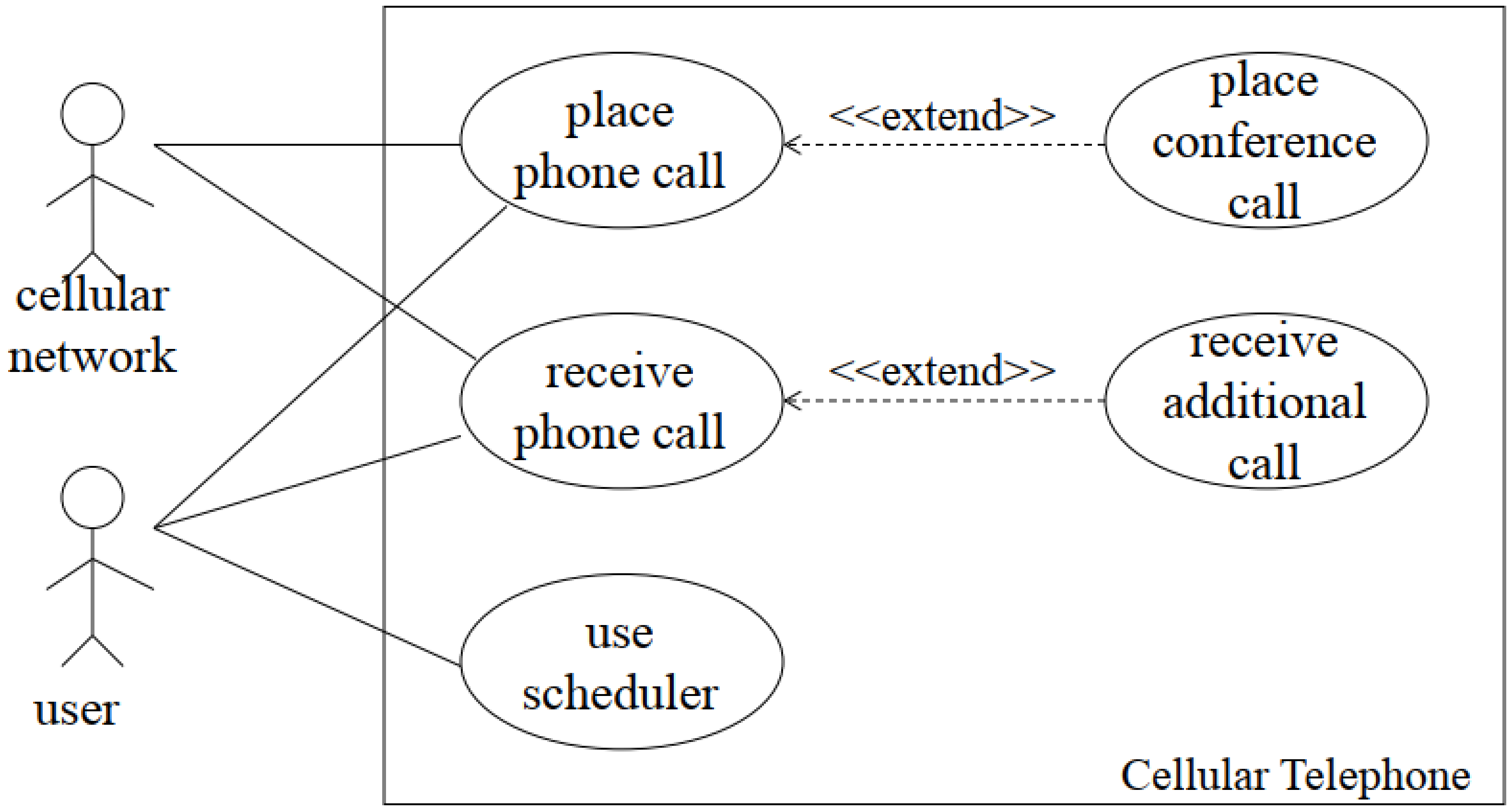


- Base use case implicitly incorporates the behavior of another use case at certain points called *extension points*.
- The base use case may stand alone, but under certain conditions its behavior may be extended by the behavior of another use case.
- In *extend relationship*, you are linking an **optional** use case to a standard use case.
  - *Register course* (std) may have *Register for Special Class* (extend use case). It may be a class for non-standard students, in unusual times, special topics, etc.
  - The *optional* extends the standard.

Example


Customer
Search Product
Navigate Deals
Checkout
Handle Order Status
View Product Details
After page generation
Rank Supplier
Write Revie w
Add to cart
Login
Register
«include»
«include»
«include»
«include»
«extend»
«extend»
«extend»
«extend» user is not a member

Example


cellular network
user
place phone call
<<extend>>
place conference call
receive phone call
<<extend>>
receive additional call
use scheduler
Cellular Telephone

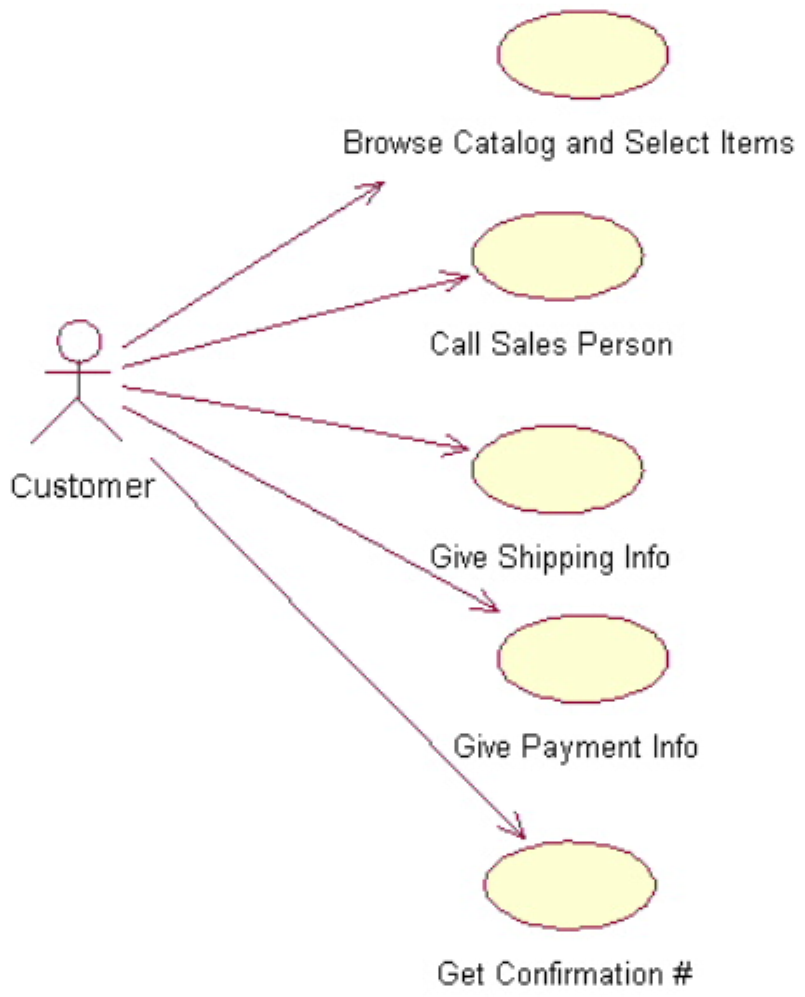
Example

- First we list the main system functions (use cases).
  - Business events demanding system's response
  - users' goals to be accomplished
  - CRUD data tasks
  - Naming use cases
- Draw actors and use cases and connect them
- Specify include and extend relationships

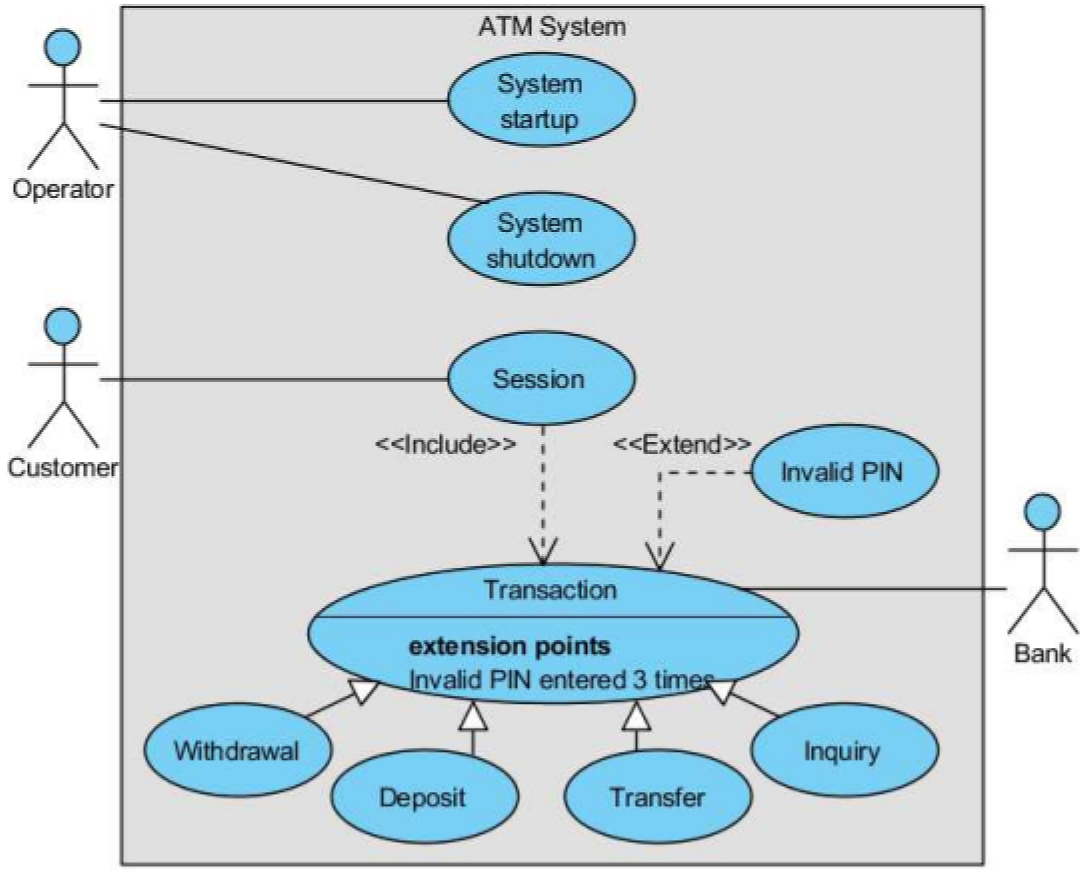
Example > cont.

- First, we begin with the use case.
  - A user placing an order.
    - Browse catalog and select items
    - Call sales representative
    - Supply shipping information
    - Supply payment information
    - Receive confirmation number
- Then we translate the *use case sequence* into diagram.

Example > cont.



Example (ATM)


ATM System
Operator
System startup
System shutdown
Customer
Session
<<Include>>
<<Extend>>
Invalid PIN
Bank
Transaction
extension points
Invalid PIN entered 3 times
Withdrawal
Deposit
Transfer
Inquiry

References

- https://courses.cs.washington.edu/courses/cse403/15sp/lectures/L4.pdf
- https://www.inf.ed.ac.uk/teaching/courses/seoc/2004_2005/notes/LectureNote03_UseCases.PDF
- http://csis.pace.edu/~marchese/CS389/L9/Use Case Diagrams.pdf
- https://warwick.ac.uk/fac/sci/physics/research/condensedmatt/imr_cdt/students/david_goodwin/teaching/modelling/l1_introuml.pdf
- https://www.cs.sjsu.edu/~pearce/modules/lectures/uml/class/index.htm