50.003
Elements of Software Construction
Lecture 2

UML Use Cases

Use Case Description

Scope

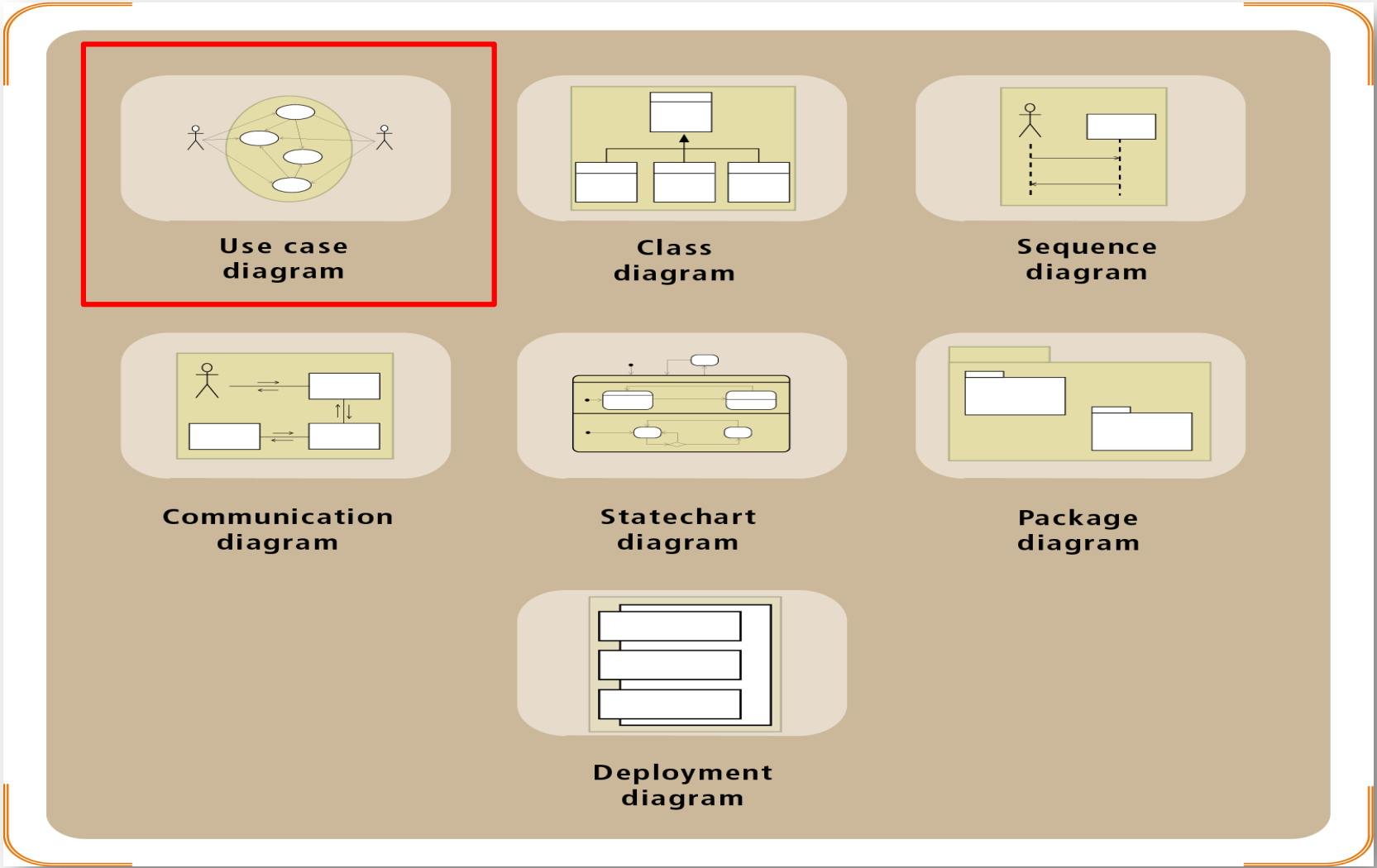
- Use cases
  - misuse cases
- Use case model
  - the components of a Use Case Diagram
  - Include and extend use cases
- Developing Use Case Model
- Use Case Description
  - parts of a Use Case Description
  - <<include>> and <<extend>> in Use Case Description
  - Guidelines for writing Use Case Description

Learning Outcomes

To be able to do the following:

- Explain what a Use Case is.
- Explain what misuse cases are
- Explain what a use case description and why it is needed
- Explain what use case scenarios are
- Write an acceptable Use Case Description
- Identify alternate flows in use case description
- Incorporate <<include> and <<extends>> relationships in use case description.

UML Diagrams used for Modelling



UML Use Case Diagrams

- Contribute to understanding system behaviours
- Provide a high-level overview of the requirements for a system
  - Captures the functional requirements
- Do not delve into detailed step-by-step processes
- Components of a Use case diagram
  - Use cases
  - Actors
  - Associations
  - System boundary
  - packages

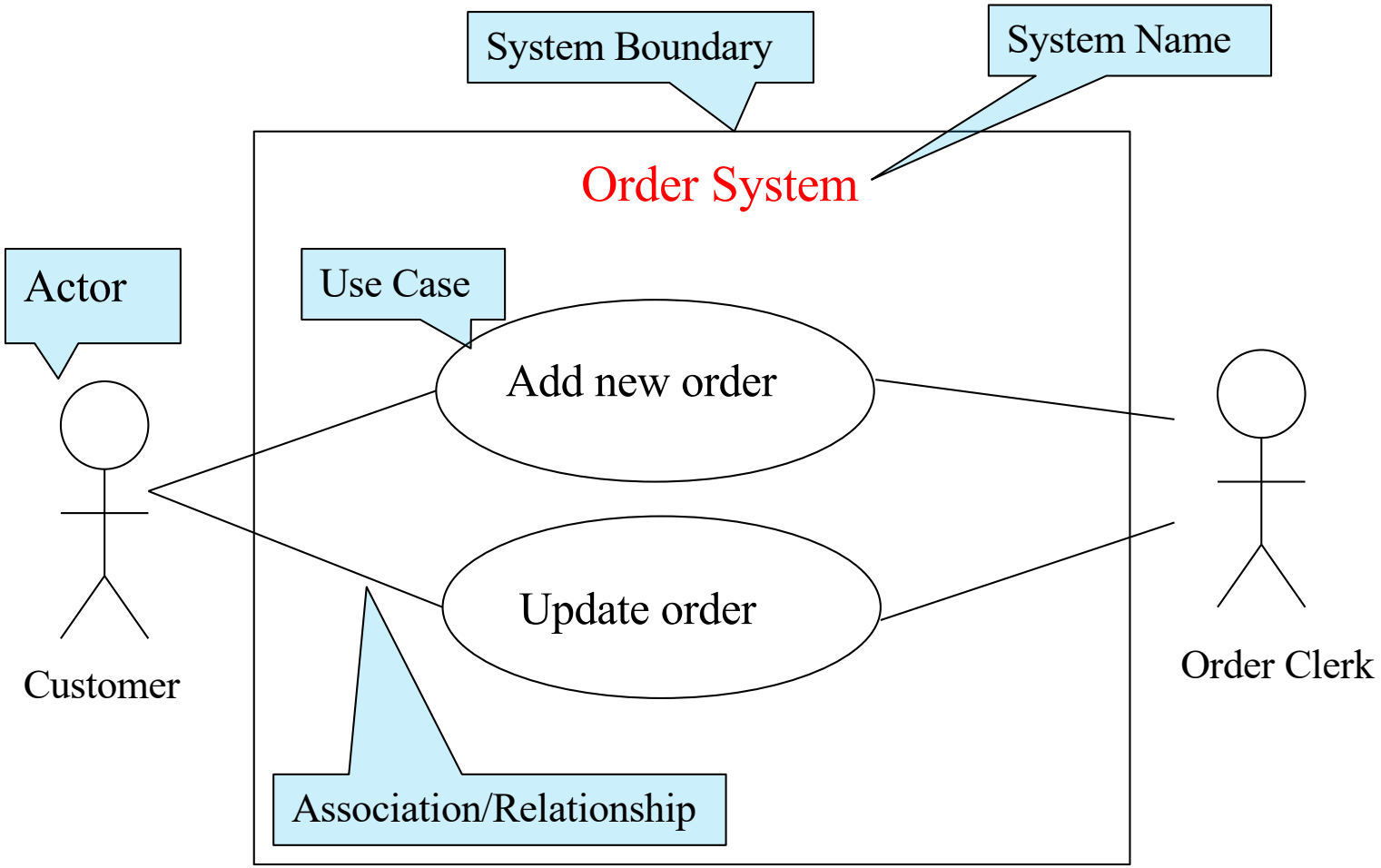
Components of a Use Case Diagram

- Use cases.
  - symbolized by an oval
  - describes a user goal (system function)
- Actors.
  - Simple stick figure
  - a person, organization, or external systems or devices that plays a role in one or more interactions with the system.
- Associations.
  - connecting lines that match actors to use cases
  - shows interactions between use cases and actors.

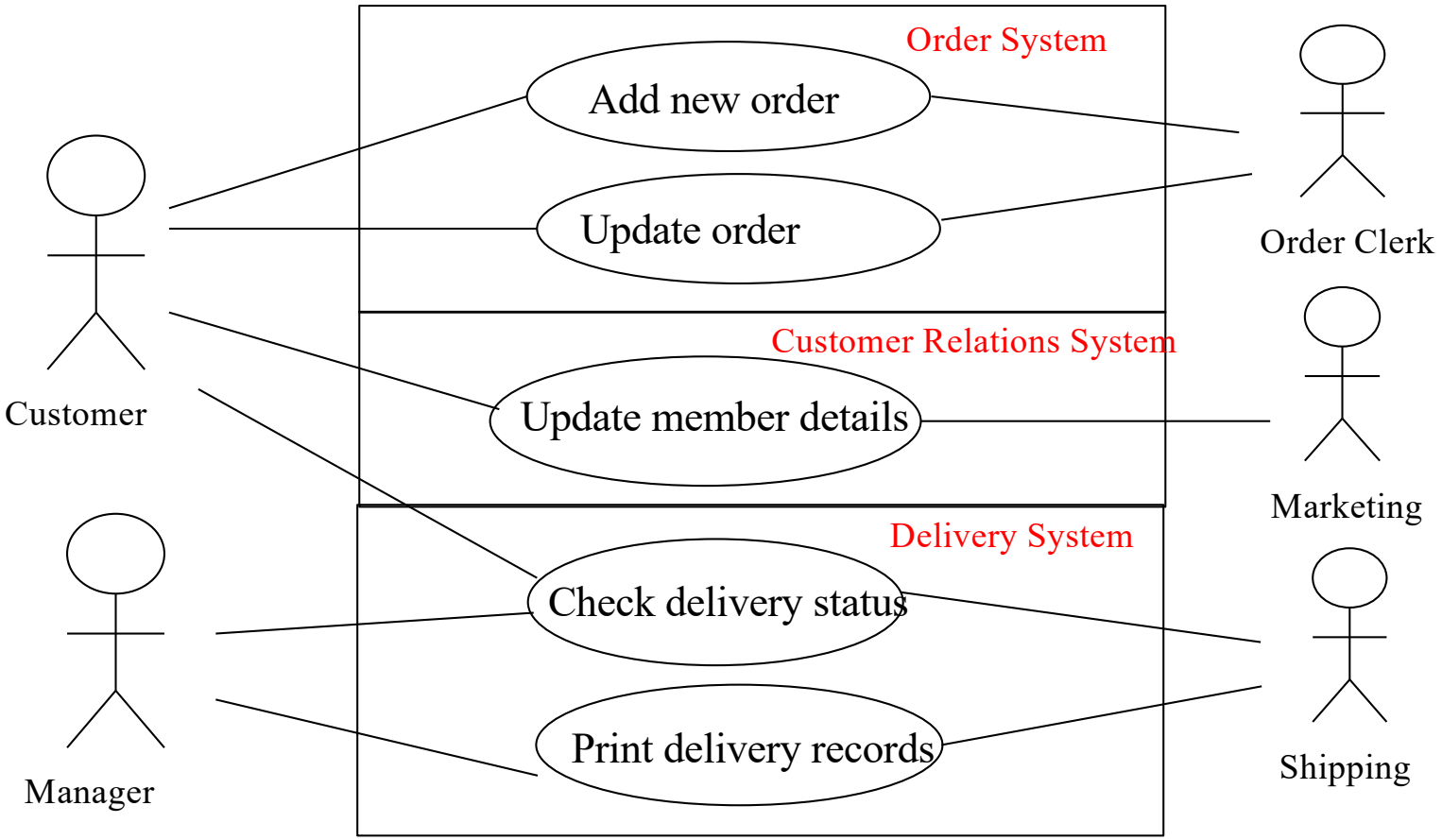
Components of a Use Case Diagram

- System boundary
  - a box and label to draw the system boundary and provide the name of the system being modelled.
  - indicates the scope of your system.
  - Anything within the box represents functionality that is in scope and anything outside the box is not.
- Packages (optional).
  - UML constructs that enable you to organize model elements (such as use cases) into groups

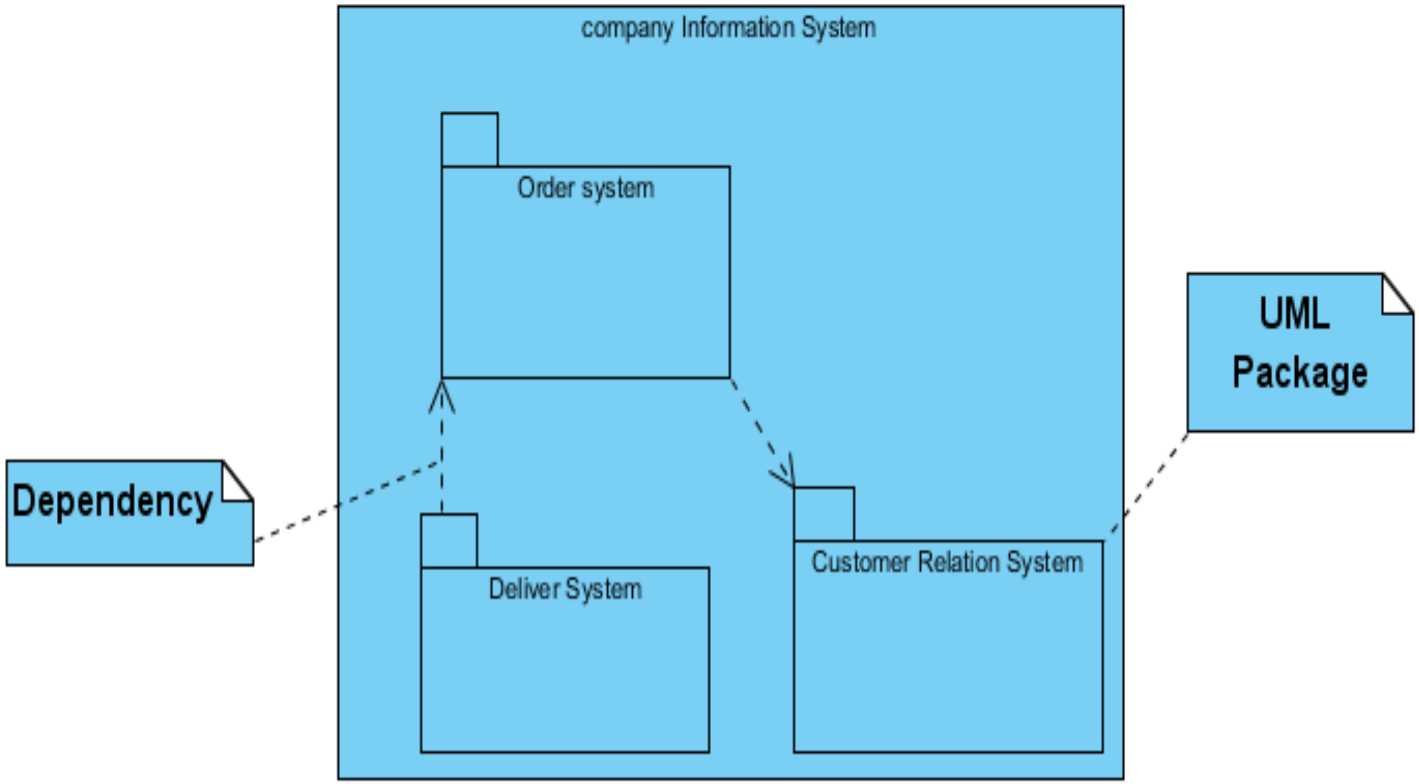
A Use Case Diagram of the Order System with System Boundary



A Use Case Diagram of multiple subsystems – using boundary



A Use Case Diagram of multiple subsystems – using packages



Identifying Use Case

- **Use Case**
  - Something that the actors wants the system to do
  - Always triggered by actor(s)
  - Identified from the point of view of the actor
- **Use Case Name**
  - A short descriptive name that is a verb phrase
  - E.g. Add orders, Cancel membership, Print reports

Example: *A online bookstore allows customer to buy books and payment is done through an external payment system (e.g. paypal).*

What is the use case?

Identifying Actors

- **Source**
  - Person or thing initiating the business event
  - Initiates the trigger for the event
    - E.g. Customer calls customer service officer to update address for credit card

Source: Customer

Actor: Customer service officer

- **Actor**
  - Person or thing that directly interacts with the system
  - Note the difference between Actor and Source

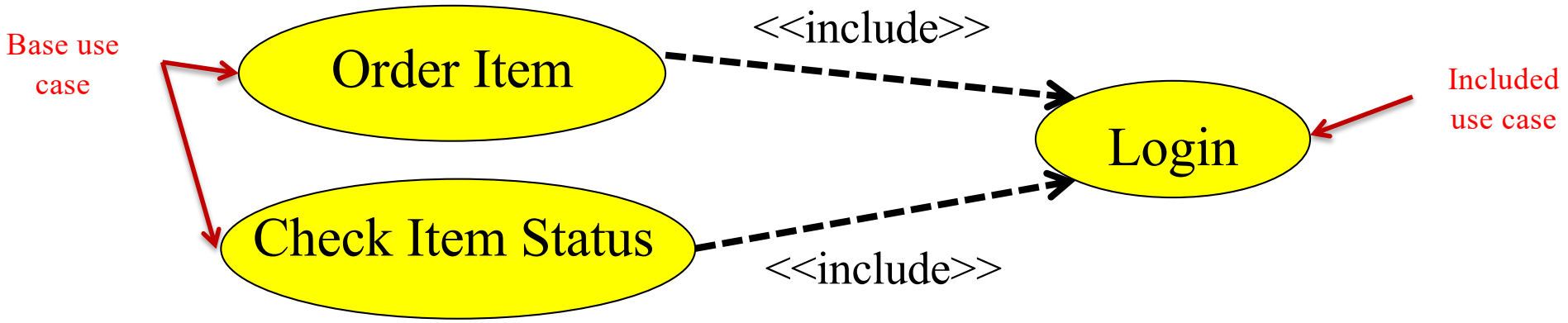
Example: *A online bookstore allows customer to buy books and payment is done through an external payment system (e.g. paypal).*

Who is/are the actor(s)?

Use Case Inclusion <<include>>

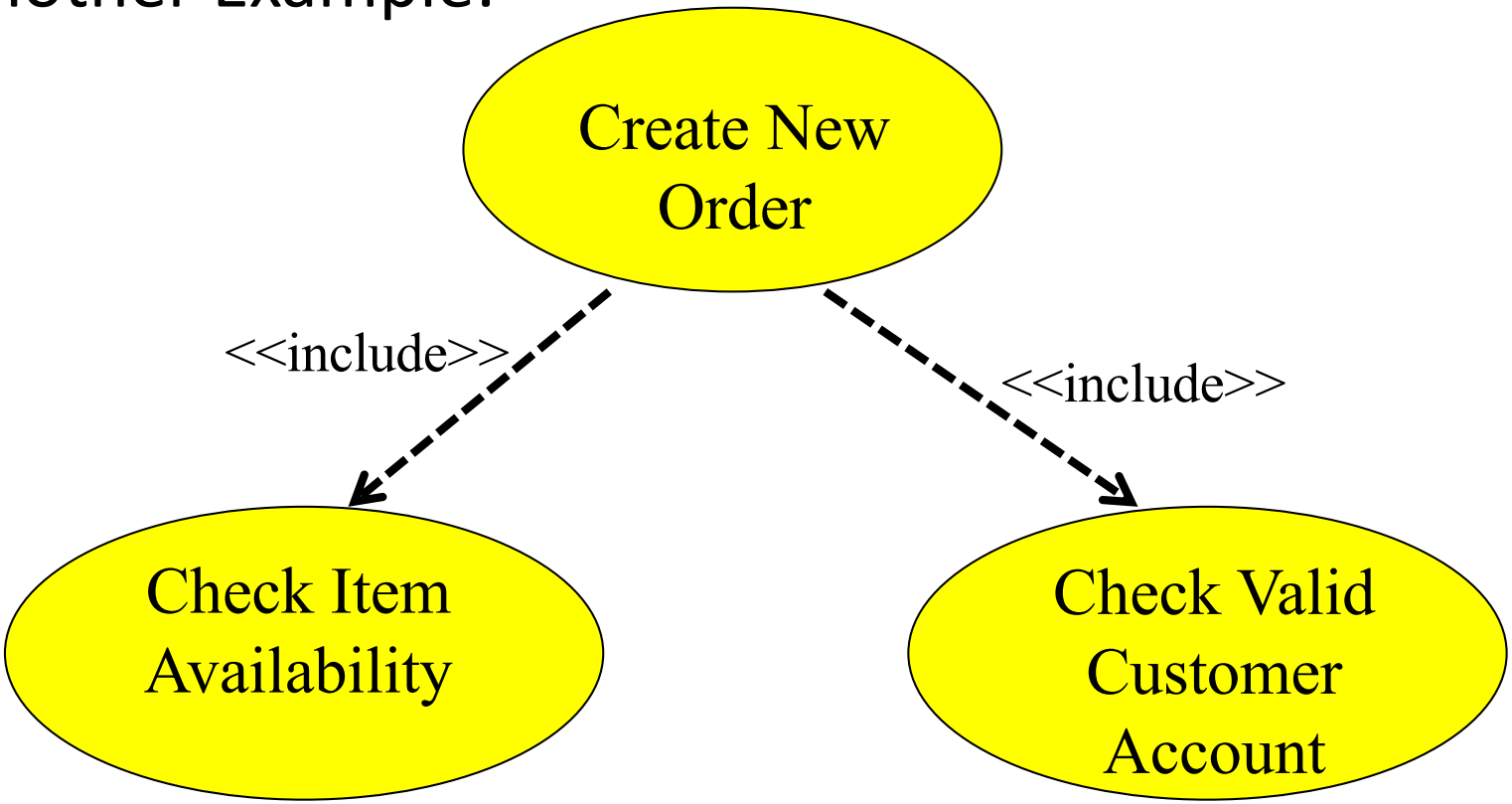
- Use case inclusion representation 

- Additional Use cases can be identified when there are common functionality from several use cases.
- Base use case is not complete without all its inclusion use cases



Use Case Inclusion <<include>>

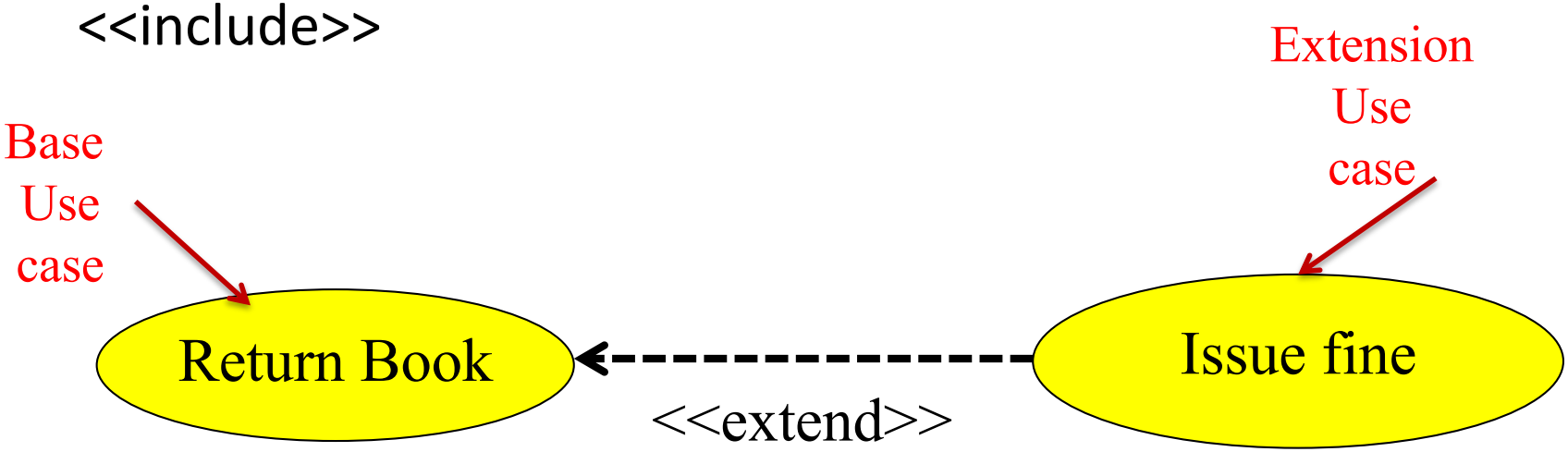
- Another Example:



Use Case Extension <<extend>>

- Use case extension representation 

- Additional Use cases can be identified when additional functionality is required for a base use case
- Base use case is complete without its extensions unlike in <<include>>



*Notice the arrowhead points to the base use case

Misuse Cases

- opposite of use cases
- scenarios that describe how a system can be abused or attacked
- identify potential security threats and vulnerabilities
- guide the countermeasures development to protect the system
- proactively build more robust systems.

Misuse Cases

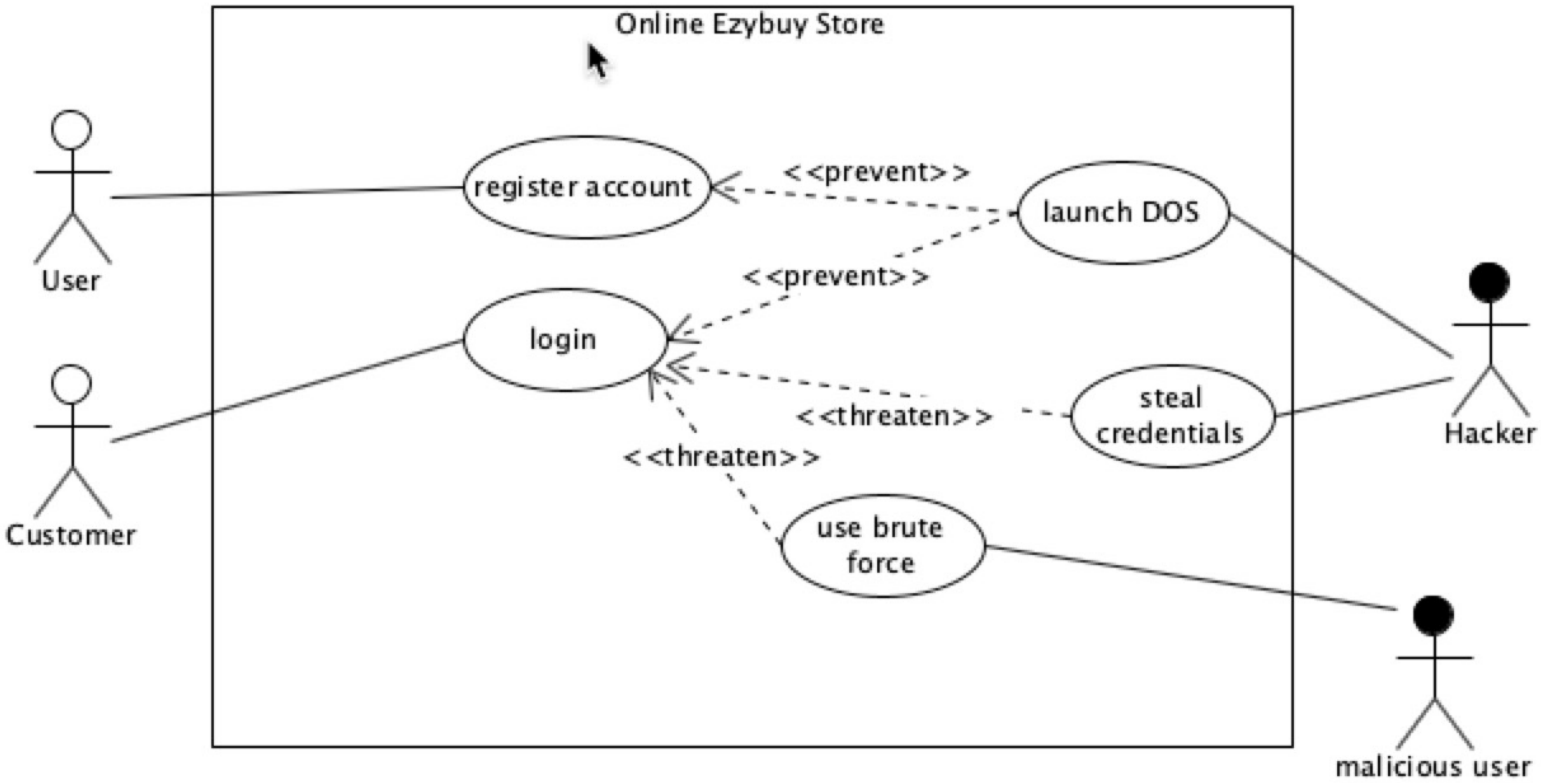
Examples:

1. Unauthorized Access
2. Data Tampering
3. Denial of Service (DoS)
4. Eavesdropping
5. Privilege Escalation

Representing Misuse Cases

- Extension of the Use Case diagram
- Identify Threat Actors
- Convert Use Cases to Misuse Cases
- Use <<extend>> notation and change stereotype name to risk posed to base use cases

Representing Misuse Cases



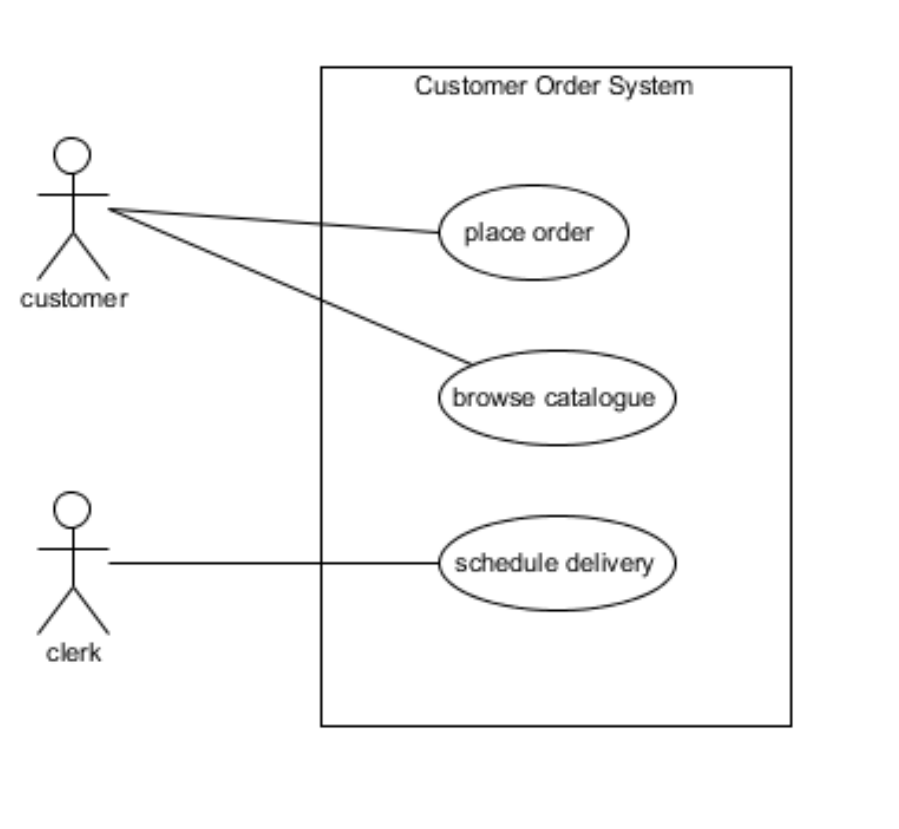
Developing a Use Case Diagram

- Developing a Use Case diagram is an **Iterative process**
  - [1] Identify the actors and roles
  - [2] Extract system capabilities
  - [3] For each actor map the relationship lines

  **Sometimes you will uncover new system capabilities*
  - [4] For system capabilities map the relationship lines to the actors(s)

  **Sometimes you will uncover new actors*
  - [5] Iterative process till model of the system stabilizes

Is a picture worth a thousand Words?

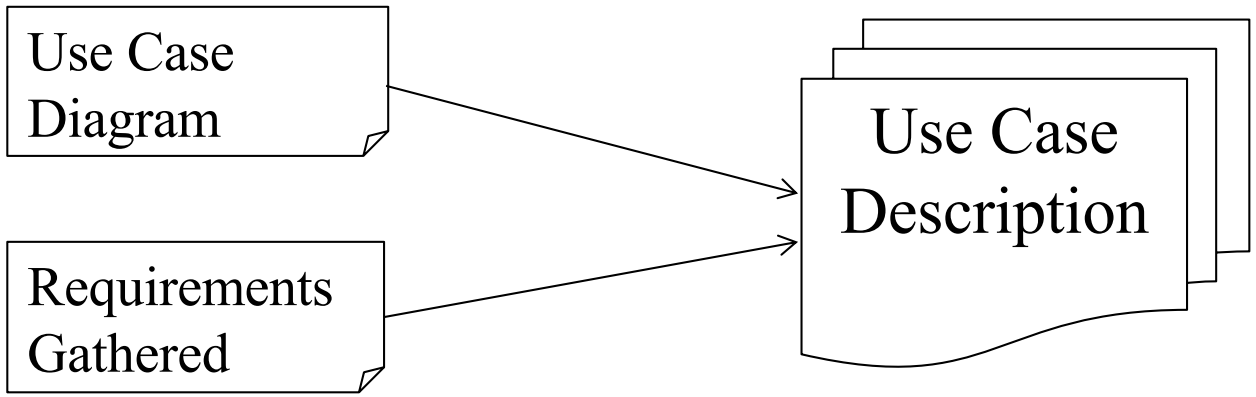


- Model system functional requirements
  - Users' goals
- What does this diagram don't tell you?
  - how the goal is achieved
  - what the steps involved
  - what data are required
- For additional details
  - an accompanying textual description
  - use case description

Use Cases → **Use Case Description**

Use Case Descriptions

- Each use case is accompanied with a detailed use case description



Use Case Descriptions

- Use cases have **internal complexity:**
  1. Sequence of steps to execute business process to achieve the user's goal
  2. Several variations may exist within single use case - eg: "place order" steps varies : is it customer a registered member entitled to privileges?
  3. Each variation is called a scenario
     a. Represents a possible situation that may arise
  4. Important to identify all possible scenarios
     a) System must be able to handle all these scenarios

Use Case Description Components

1. Use case name/title
   - results-oriented name for the use case as it appears in the use case diagram
2. Use case ID
   - a unique identifier, as specified by the organisation so that functional requirements can be tracked.
3. Use case description
   - a description of the reason for and outcome of this use case,
   - a high-level description of the sequence of actions
   - the outcome of executing the use case
   - include input data required and their description, business rules, constraints.

Use Case Description Components

4. Primary Actor
   - Actors that triggers/initiates the use case (initiator)
5. Secondary Actor
   - Actors that interact with the use case after it is triggered (participator)
4. Preconditions
   - List any activities that must take place, or any conditions that must be true, **before** the use case can be started
7. Postconditions
   - List the state of the system **AT THE CONCLUSION** of the use case execution.

Use Case Description Components

8. Main flow
   - List the user actions and system responses that will take place during execution of the use case under normal, expected conditions.
   - The sequence of actions and responses will ultimately lead to accomplishing the goal stated in the use case name and description
9. Alternative flows
   - Document other, legitimate usage scenarios that can take place within the use case
   - Each flow should be preceded with a scenario name
   - List the user actions and system responses for the alternate scenario

Guidelines for use case description

- UML does not provide standard rules for writing detail use case descriptions
- There are guidelines that one can adhere to
- Use Simple Grammar, eg:
  - Librarian enters book ID
  - System displays book details
- Show clearly who is in control at point in time
  - Always start with:
    - The *actor* …. (replace actor with the actor's name in the use case)
    - The system …

Guidelines for use case description

- Eg:
  - The customer enters quantity to purchase
  - The system adds items into order list
- Do not impose constraints on user interface
  - User ~~clicks on OK button~~ to submit personal details ✗
  - User submits personal details to system ✓
- <u>User Input1 (by entering data):</u>
  1. The actor enters ….
  2. The actor submits …
  3. The system validates ….
  4. The system verifies …

Guidelines for use case description

- User Input2 (by selecting data):
  1. The system retrieves list of …
  2. The system displays list of …
  3. The actor selects …
  4. The actor submits ….
  5. The system validates ….
- System responses:
  - The system computes …
  - The system prints …
  - The system triggers …
  - The system :-
    - inserts, creates, updates, deletes

Use Case Description Template

Use Case ID:	
Use Case Name:	
Created By:	
Date Created:	

Description:	
Primary Actor:	
Secondary Actor:	
Preconditions:	1. 2.
Postconditions:	1. 2.
Main Flow:	1. 2. 3.
Alternative Flows:	2a ScenarioName1 1. 2. 4a ScenarioName2 1. 2.

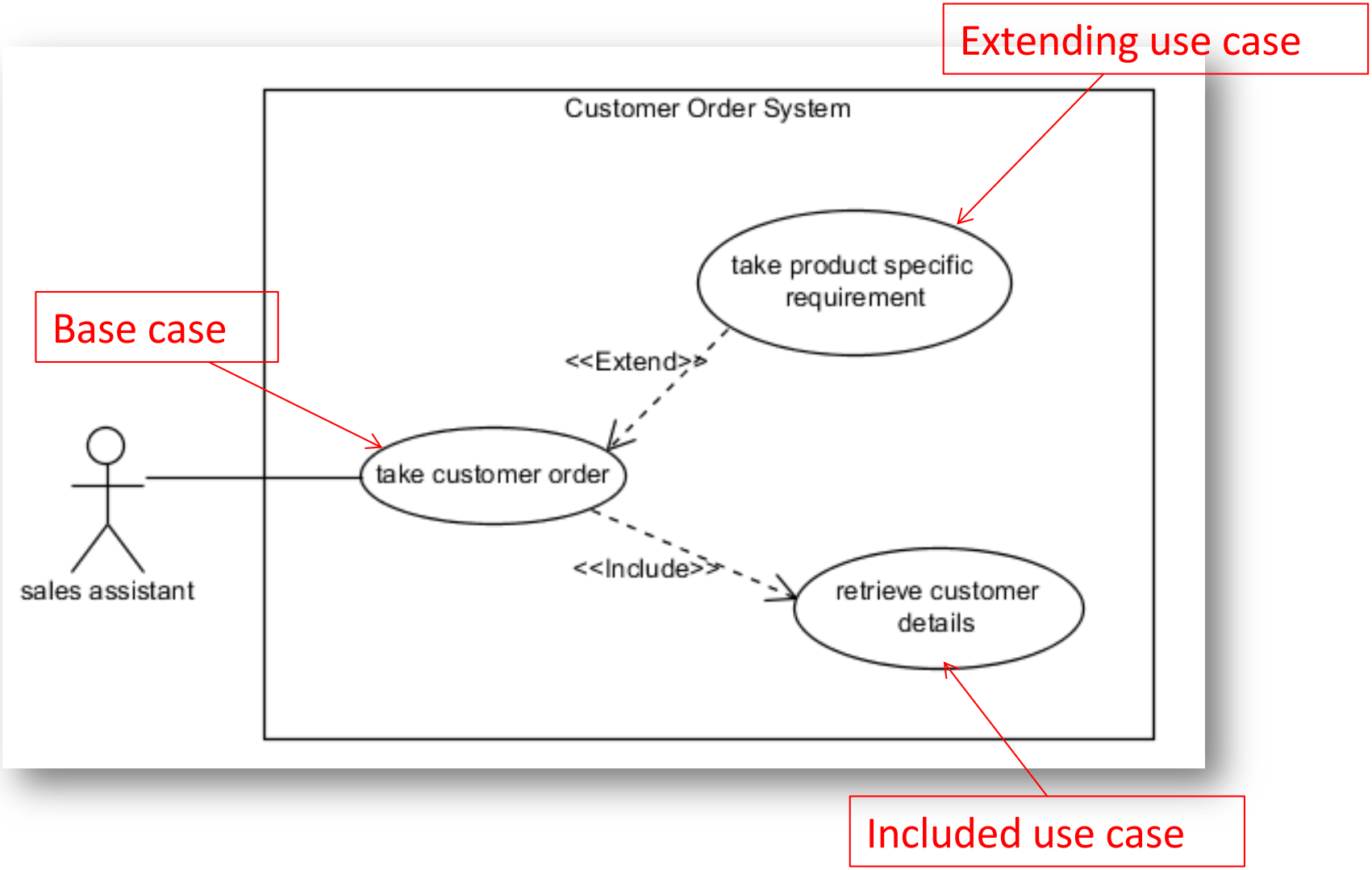
Use Case Description Example - login

Use Case ID:	ACC_UC_1
Use Case Name:	login
Created By:	Joe Blogs
Date Created:	1-1-2012
Description:	This use case allows user to login into the system to access the relevant functions according to the user's role. The various user roles are staff, admin staff, system administrator, manager and department head. To login to the system, all users have to enter their unique staff id which is their NRIC number. The users have a maximum of 3 attempts to login after which their account are locked and they will have to contact the system administrator to unlock their account upon successful login the system will display the relevant user's home page.
Primary Actor:	User
Secondary Actor:	None
Preconditions:	1. User has to have a valid account
Postconditions:	1. The system displays the relevant homepage

Use Case Description Example - login

Main Flow:	1. The user enters the staffID and password 2. The user submits the staffID and password 3. The system validates the staffID and password 4. The system verifies the staffID and password 5. The system displays the user's homepage 6. The use case ends
Alternative Flows:	3a Missing staffID and/or password 1. The system prompts for staffID and password 2. Use case resumes at main flow step 1 3b Maximum 3 attempts exceeded 1. The system displays "Maximum attempts exceeded, contact administrator" message 2. The system locks user account 3. The use case ends 4a Invalid staffID and/or password 1. The system displays "Invalid staffID and/or password" message 2. The system prompts for staffID and password 3. Use case resumes at main flow step 1

<<include>> & <<extends>> use case relationships



include base case in Main Flow

Use Case ID:	PDC_UC_2
Use Case Name:	Take customer order
Created By:	Jane Doe
Date Created:	1-1-2012
Description:	This use case allows … …
Primary Actor:	Sales assistant
Secondary Actor:	None
Include use cases:	1. **retrieve customer details**
Preconditions:	2. sales assistance must login successfully 3. customer must have an account
Postconditions:	1. The system creates a new customer order
Main Flow:	1. System **triggers retrieve customer details use case** 2. The sales assistant enters and submits product ID 3. The system validates product ID 4. The system retrieves and displays product details 5. The sales assistant enters and submits quantity 6. The system validates the quantity 7. **Repeats step 2 to step 5 for all products** 8. The sales assistance confirms customer order 9. The system creates a new customer order 10. The use case ends

At the point where the included use case Is needed, the system triggers it

For repetition of steps

extend base case description

Use Case ID:	PDC_UC_2
Use Case Name:	Take customer order
Created By:	Jane Doe
Date Created:	1-1-2012
Description:	This use case allows …… ….
Primary Actor:	Sales assistant
Secondary Actor:	None
Include use cases:	1. retrieve customer details use case
Preconditions:	1. Sales assistant must login successfully 2. customer must have an account
Postconditions:	1. The system creates a new customer order
Main Flow:	1. System triggers retrieve customer details use case 2. The sales assistant enters and submits product ID 3. The system validates product ID 4. The system retrieves and displays product details 5. **(Extension point: take product specific requirements)** 6. The sales assistant enters and submits quantity 7. The system validates the quantity 8. Repeats step 2 to step 5 for all products 9. The sales assistance confirms customer order 10. The system creates a new customer order 11. The use case ends

extending use case description

Use Case ID:	PDC_UC_4
Use Case Name:	Take product specific requirements
Created By:	Jane Doe
Date Created:	1-1-2012
Description:	This use case allows …… ….
Include Use Cases:	1. None
Extends Use Cases:	**1. take customer order**
Primary Actor:	Sales assistant
Secondary Actor:	None
Preconditions:	1. none
Postconditions:	1. The system computes additional product cost
Main Flow:	1. The system prompts for product dimensions 2. The sales assistant enters product dimensions 3. The sales assistant enters color choice 4. The sales assistance submits product specific requirements 5. The system validates product requirements details 6. The system computes and displays additional cost 7. The use case ends

Use Case Description Template (With Extends and Include)

Use Case ID:	
Use Case Name:	
Created By:	
Date Created:	
Description:	
Primary Actor:	
Secondary Actor:	
Include use cases:	1. . 2.
Extends use cases:	1. 2.
Preconditions:	1. 2.
Postconditions:	1. 2.
Main Flow:	1. 2. 3.
Alternative Flows:	2a ScenarioName1 1. 2. 4a ScenarioName2 1. 2.

Summary

- The concept of UML use cases
- Relationships between use cases
  - include
  - extend
  - misuse cases
- Drawing a use case diagram
- identifying use case scenarios
- Writing use case description

Cohort Exercise – Case Study

Global Knowledge Pte Ltd has a resource center to loan the software items to its internal staff within the company. A web application is to be developed with the functions described below.

The system allows staff to submit loan requests by specifying the purpose and the duration of the software items to borrow. The system will route the requests to the resource manager for approval and the staff will be notified by email on the outcome of the request. The system will allow the staff to check-out the items with a loan transaction created.

The software must be returned by the due date. The return date will be recorded, and the loan record status will be updated. If the loan is overdue, a reminder email will be sent to the relevant staff.

The system also allows the Resource Manager to generate a summary report to consolidate all the loan transactions at any point in time.

Cohort Exercise – Case Study

Form a team of between 7-5 members

- Identify the functional and non-functional requirements in the above case scenario.
- Construct a use case diagram illustrating the main functionality of the web application.
  - Identify and include potential misuse cases in your use case diagram.
- Each member of the team should choose a use case and write an accompanying use case description