

Abdelhafid Boussouf
University Center
Mila

Faculty of Math and Computer
Science

Department of
Computer Science

Software engineering

Chapter 3

Use Case Diagram

Mrs. S.HEDJAZ

II. Use Case Diagram



01

Introduction

02

The elements of the use case diagram

03

Description of use cases

04

Example



Introduction

1- Objective:

- Allows you to develop the specifications or the document specifying the software's needs
- Allows you to structure the needs of users, the corresponding objectives of a system
- Determines the functional needs of each actor (the Who? and the What?)
- They identify the users of the system and their interaction with it
- They make it possible to define the limits of the system and the relationships between the system and its environment



Introduction

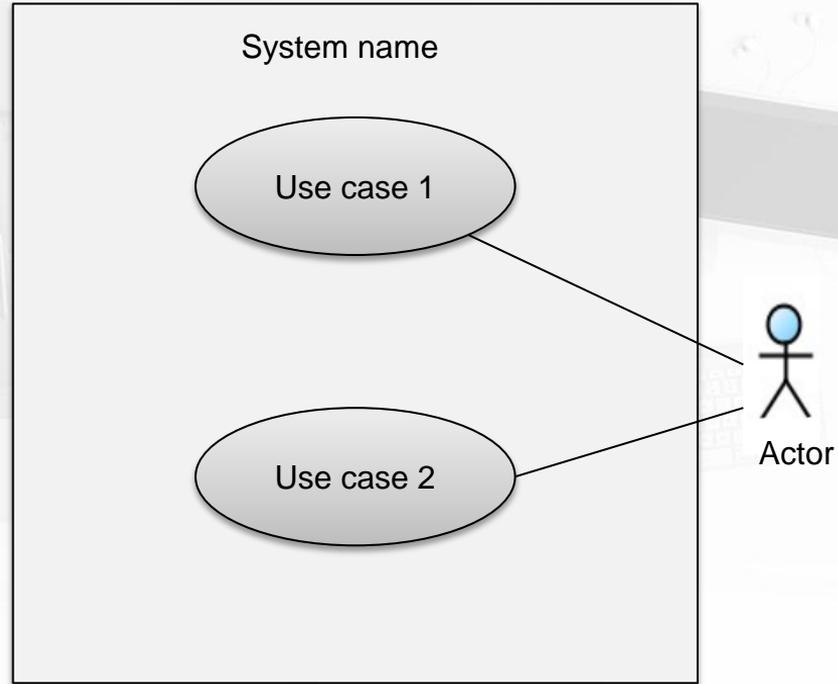
2- Definition:

- Constitutes the first step of the UML analysis by:
 - ✓ Modeling user **needs**
 - ✓ Identifying the system's key features and limitations
 - ✓ Representing the **interactions** between the **system** and its **users**
- The use case diagram describes the functionality of a system from a **user perspective**
- The use case diagram lists only **general essential and main functions** without going into detail



Introduction

3- Notation:





The elements of the use case diagram

1- Actor:

- Represents a role played by an external entity (human user, hardware device, or other system) that directly interacts with the system.
- An actor can consult and/or modify the state of the system by sending and/or receiving messages likely to carry data

1.1. How to present them?:

Actors represent themselves in the form of a small character (stick man) or in the form of a rectangular box (called a binder) with the keyword "actor". Each actor has a name.

We use:

- the stick man if the actor is human
- the binder if the actor is hardware or another system.





The elements of the use case diagram

1.2. Main Actor:

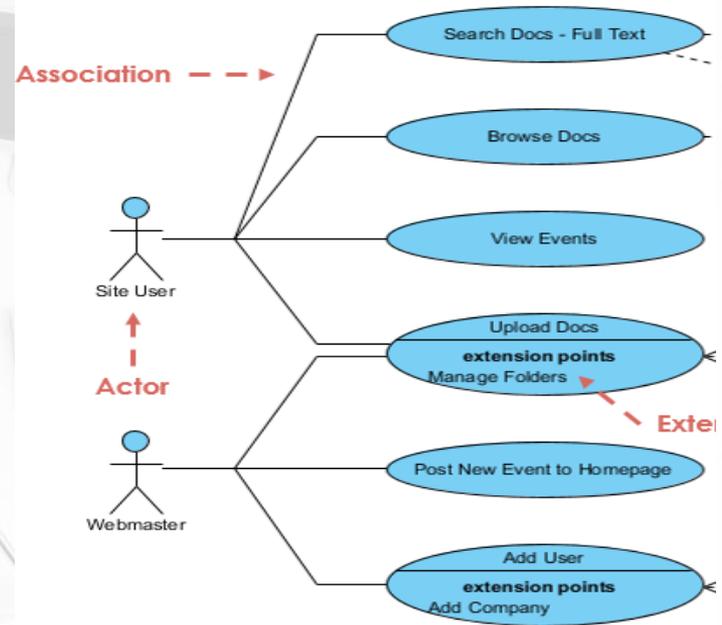
- An actor is referred to as a principal for a use case when that case is of service to that actor.
- People using the main functions of the system (human user)

1.3. Secondary actor:

- Individuals who perform administrative or maintenance tasks (human user, hardware device, other software)

1.4. Relationship between the actors: generalization:

- The point of generalization is to show that some actors inherit all the use cases of other actors, and that they also have their own specific use cases

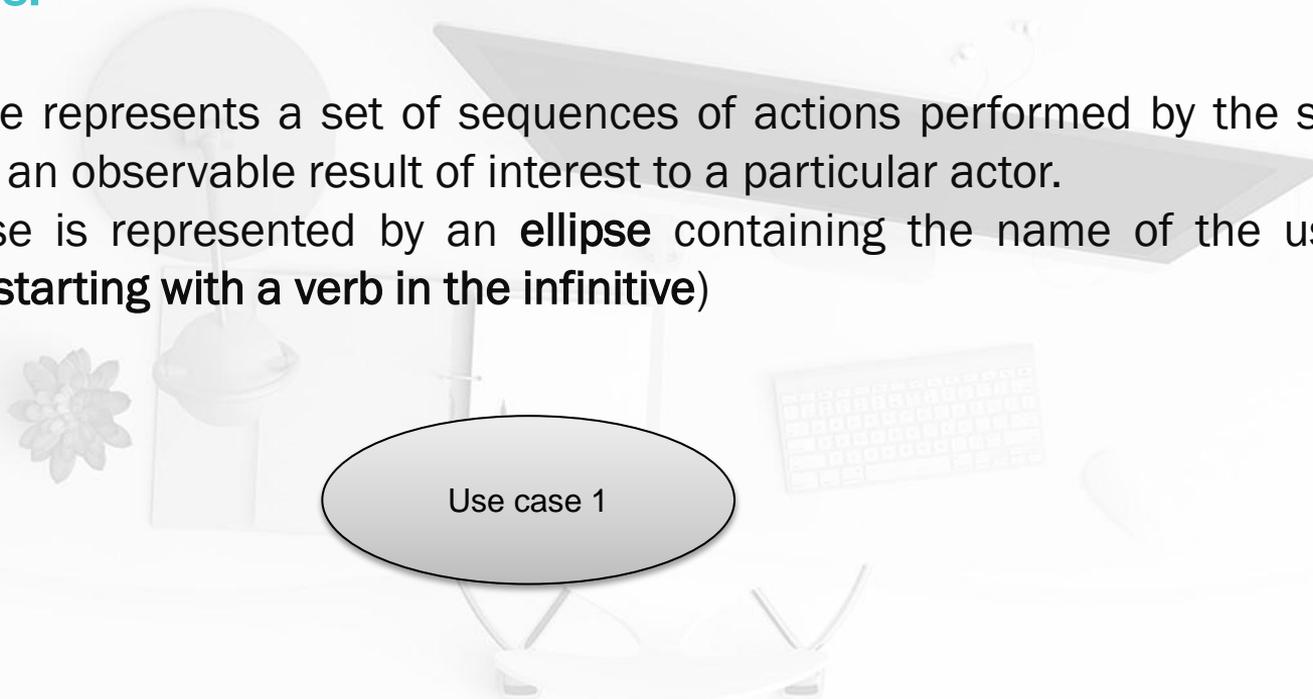




The elements of the use case diagram

2- Use Cases:

- A use case represents a set of sequences of actions performed by the system and producing an observable result of interest to a particular actor.
- A use case is represented by an **ellipse** containing the name of the use case (a **sentence starting with a verb in the infinitive**)



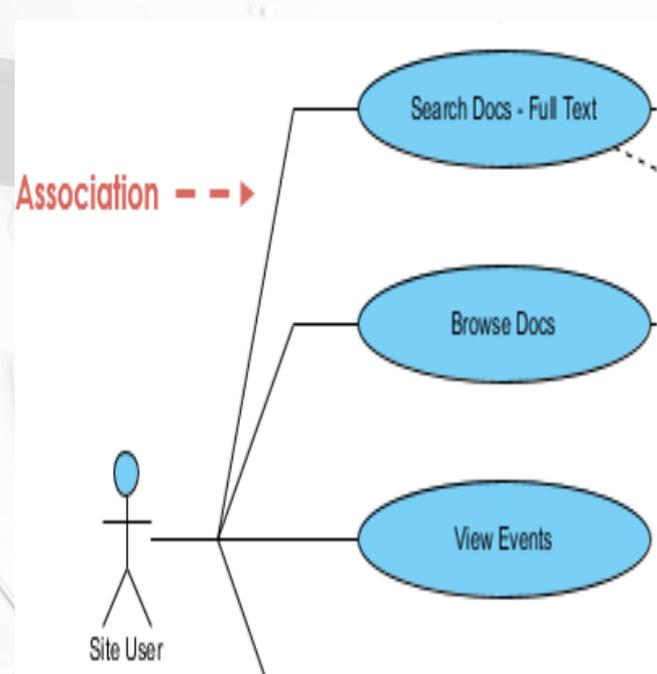
Use case 1



The elements of the use case diagram

2.1. Relationship between actors and use cases:

- Each actor is associated with one or more use cases, the association relationship.
- It is represented by a trait connecting the actor and the use case



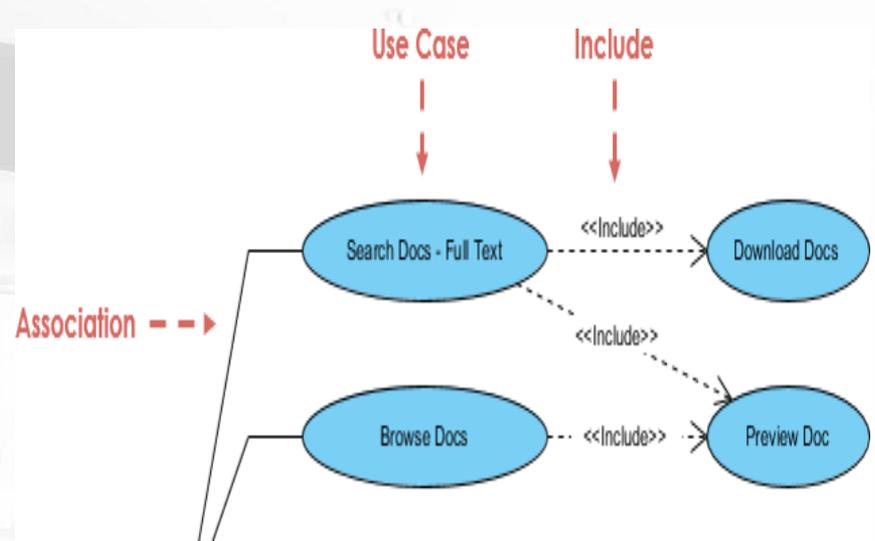


The elements of the use case diagram

2.2. Relationship between use cases:

a- Inclusion:

- The inclusion relationship is used to enrich one use case with another use case
- This relationship is represented by a dotted arrow connecting the two use cases with the "include" stereotype



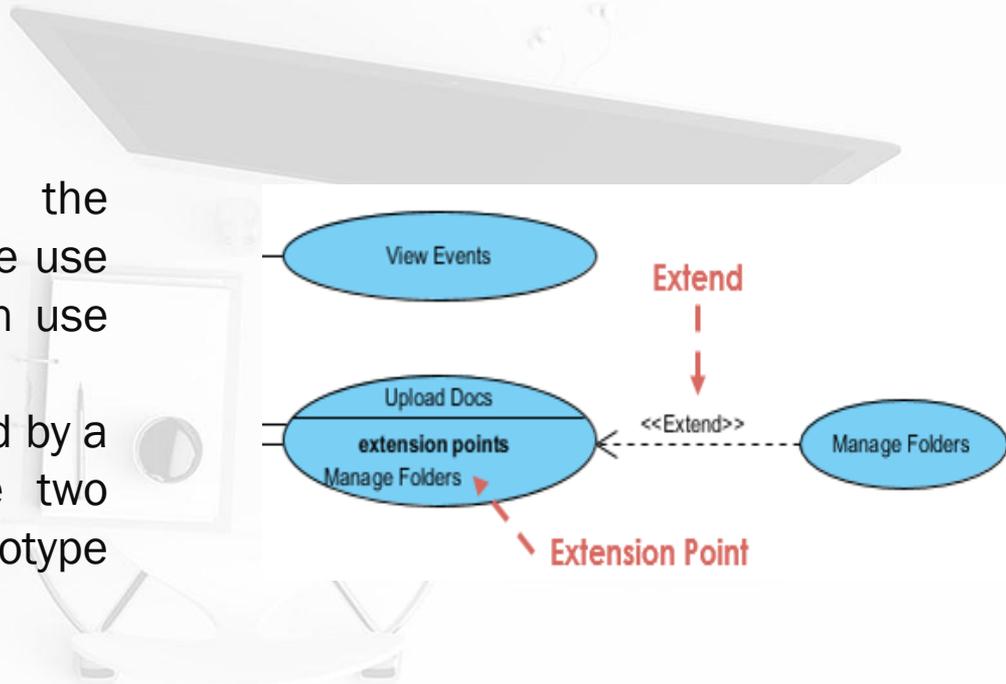


The elements of the use case diagram

2.2. Relationship between use cases:

b- Extension:

- Like the inclusion relation, the extension relation enriches one use case with another subfunction use case, but this one is **optional**.
- This relationship is represented by a dotted arrow connecting the two use cases and with the stereotype "**extend**"



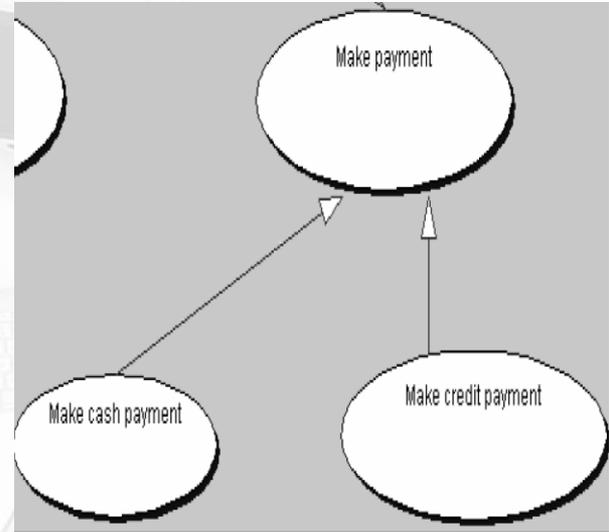


The elements of the use case diagram

2.2. Relationship between use cases:

c- Generalization or specialization:

- It is also possible to specialize one use case into another use case.
- The generalization relation is represented by an arrow with a triangular end





Description of use cases

1- Definition:

- The use case diagram describes the main functions of a system from the perspective of the actors, but does not detail the dialogue between the actors and the use cases.
- Each use case must be documented so that there is no ambiguity about how it will work and what exactly it covers.

1.1. Textual description:

A typical textual description consists of three parts: **identification, description of scenarios, and non-functional requirement**



Description of use cases

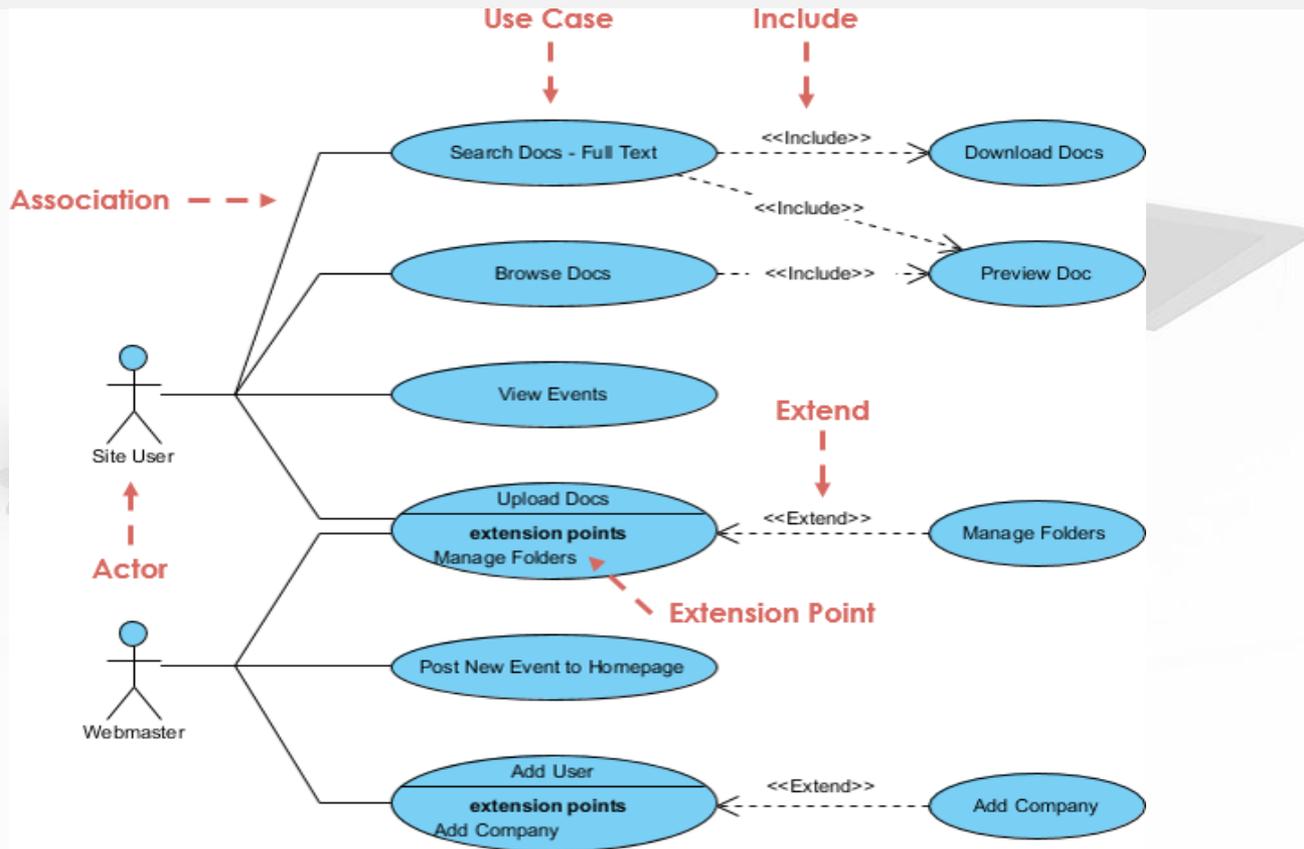
Use case	Use case name
Main Actor	Main actors.
Secondary Actor	Secondary actors.
Goal	Use case goal
Pre condition	Define what must be true before the use case can start.
Post conduction	The post conditions define what must be true when the use case ends.
Main Scenario	The one that satisfies the actors' objectives through the most direct path to success.
Alternative Scenario	Include all other scenarios, whether successful (normal completion) or failed (error).
Additional requirements	These may concern performance, security, or usability. For example, the description of the scenarios can be supplemented with screenshots of the prototype.



Description of use cases

1.2. Example:

Actor	Customer.
Objectif	Add an order.
Pre condition	The use case begins when the user is authenticated.
Post condition	The customer places an order
Nominal scenario	<ol style="list-style-type: none">1. The customer asks to fill in the order form2. The system displays the form.3. The customer enters the necessary information (surname, first name, address, etc.).4. The system records the information, it returns a success message.
Error scenario (alternative)	4-a- The customer has not filled in a required field or has entered invalid information. The system indicates to the customer "the unaccepted field with a red color and the scenario resumes from 2".
Additional Requirements	/



Bibliographies

- **Uml 2 pratique de la modélisation**, Benoît Charroux, Yann Thierry-Mieg, Aomar Osmani
Ni <https://fr.slideshare.net/nassimamine3994/uml-2-pratique-de-la-modlisation>
- **Uml 2 par la pratique**, Pascal roques
- **Les cahiers du programmeur**, Pascal roques
- **Uml en action**, Pascal roques
-