

## Use Case



# **ADRAR** **DIGIT@L ACADEMY**

PÔLE NUMERIQUE DU CENTRE DE FORMATION ADRAR

- > SUPPORT, ADMINISTRATION SYSTEMES & RESEAUX
- > DEVELOPPEMENT D'APPLICATIONS WEB & MOBILES
- > TRANSFORMATION NUMERIQUE DES ENTREPRISES

<http://www.adrar-numerique.com>

## Use Case : Définition

Le diagramme de cas d'utilisation, ou « Use case » en anglais, est la première phase d'analyse d'un projet. C'est la modélisation du métier.

Elle permet de connaître les différents processus mis en œuvre dans l'entreprise.

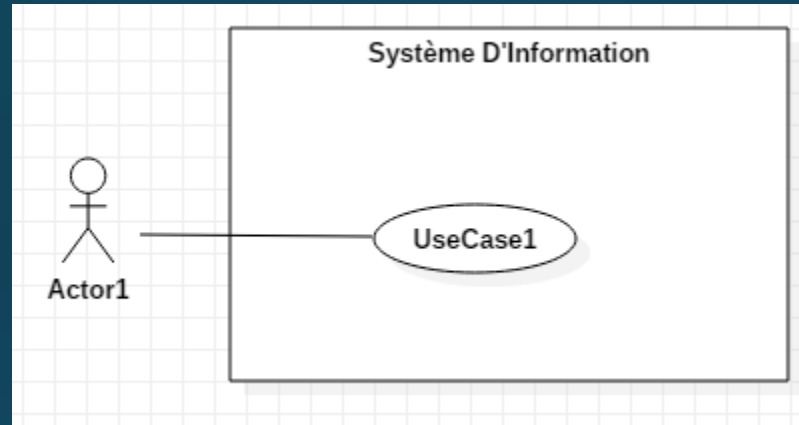
Ensuite, la deuxième phase est l'analyse du besoin.

C'est la modélisation de l'expression du besoin par rapport à un système d'information.

Nous passons donc sur la réalisation du système.

Nous allons trouver les acteurs principaux et leur demander quels sont les fonctionnalités qu'ils attendent du système informatique.

## Use Case : Définition



### Attention !

Le nom du UseCase devra commencer par un verbe à l'infinitif

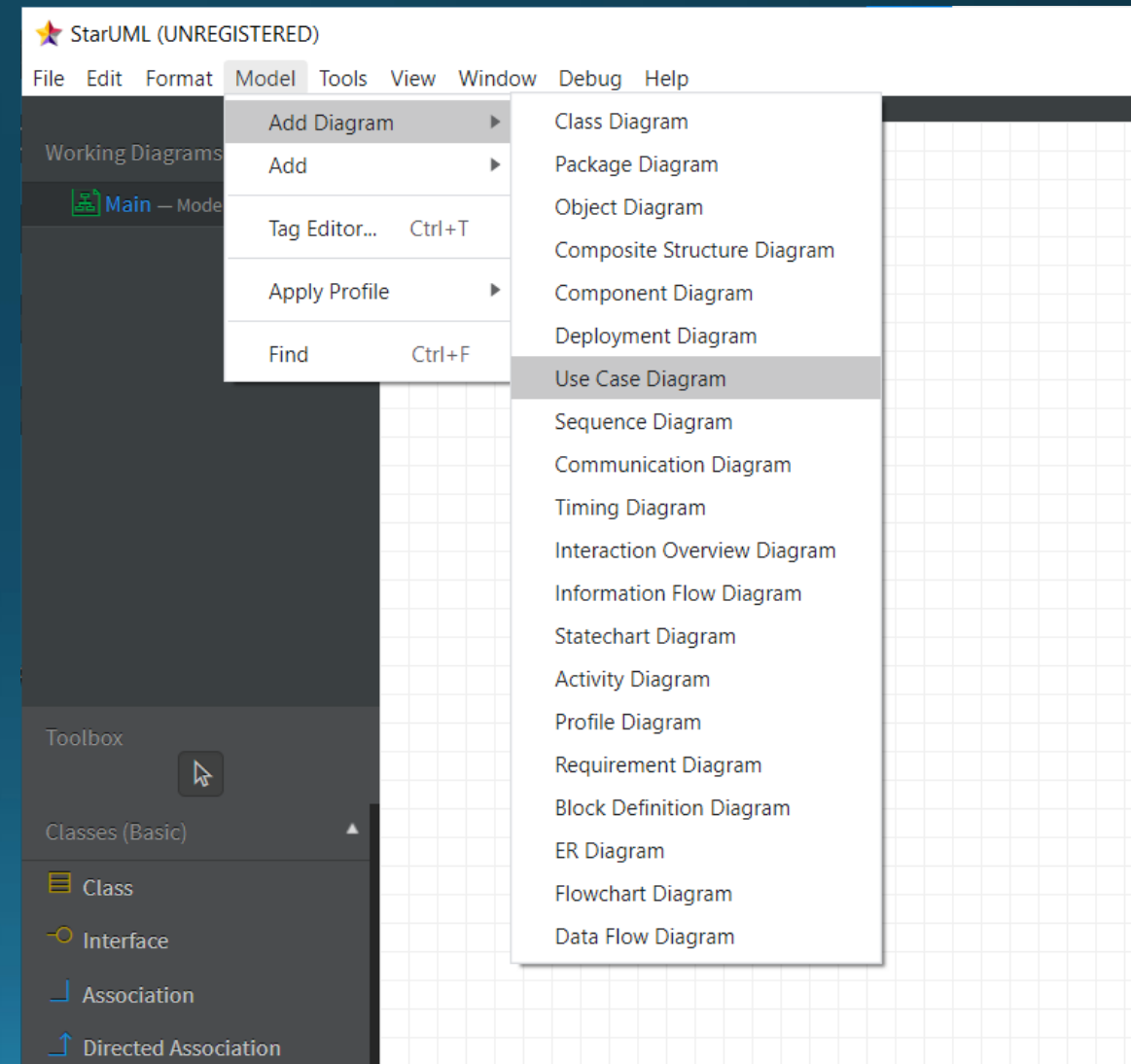
Le diagramme de cas d'utilisation est composé :

- D'un acteur, qui est une entité extérieur au système
- D'un système d'information
- D'un cas d'utilisation, use case, qui une fonctionnalité attendue

## Use Case : Star UML

Dans StarUML, il faudra faire:

Model / Add Diagram / Use Case diagram



## Use Case: Exemple

Une compagnie aérienne souhaite proposer sur un site Web :

- Un catalogue de vols
- La vente de vols

Le responsable de la compagnie aérienne vous donne les fonctionnalités qu'il souhaite avoir sur le site Web.

Les clients accèderont à l'application par Internet.

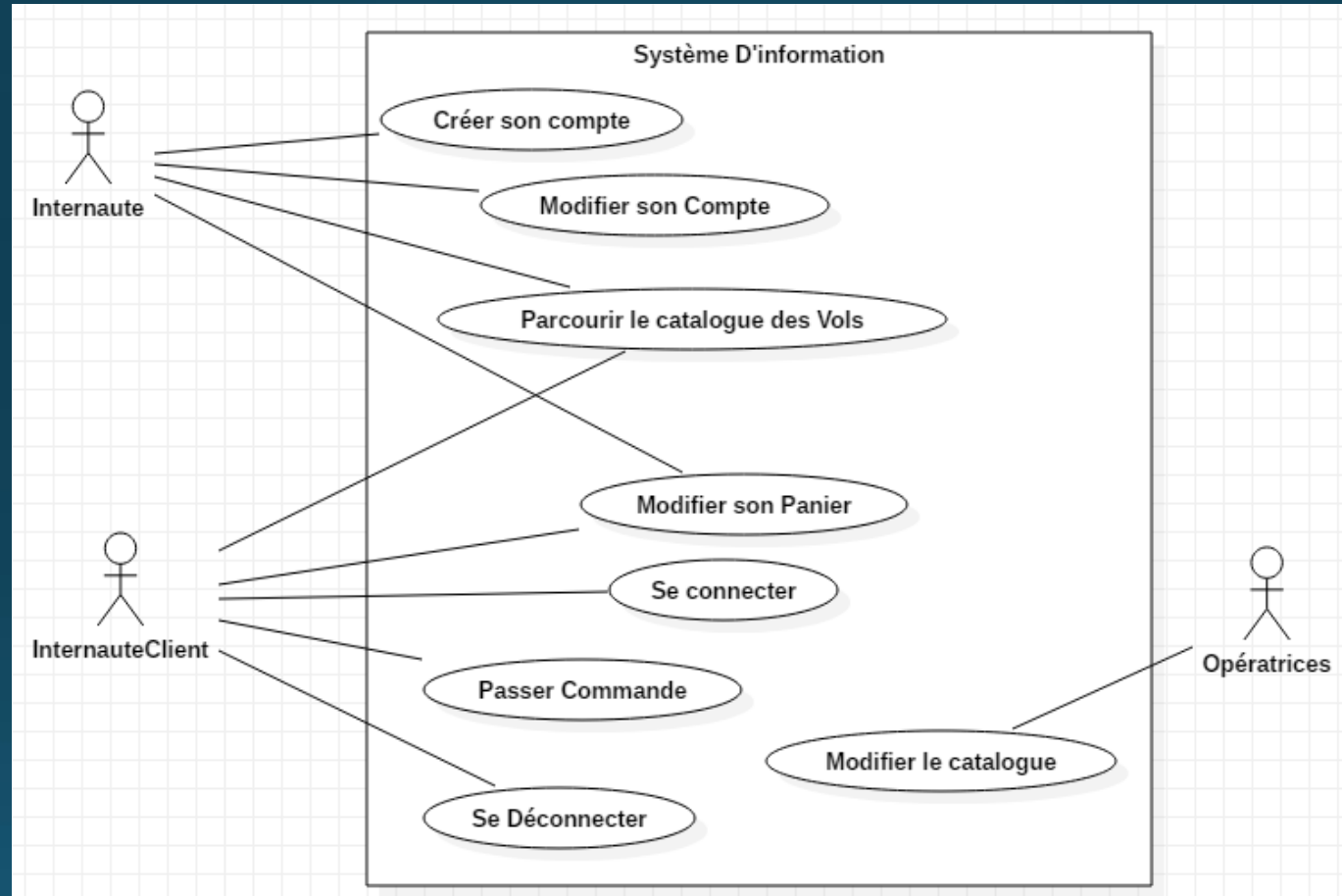
Ils pourront parcourir un catalogue de Vols proposés par la compagnie aérienne et mis à jour par les opératrices.

Le client pourra passer commande d'un vol en ligne s'il s'est authentifié.

S'il le souhaite il pourra modifier sa commande sous certaines conditions.



## Use Case: Exemple



Note: Un internaute client est différent d'un Internaute, car il aura passé une commande.

## Use Case: Notions avancées

Dans l'exemple précédent prenons les acteurs suivant :

- Internaute
- Internaute Client

Ils ont des actions en commun :

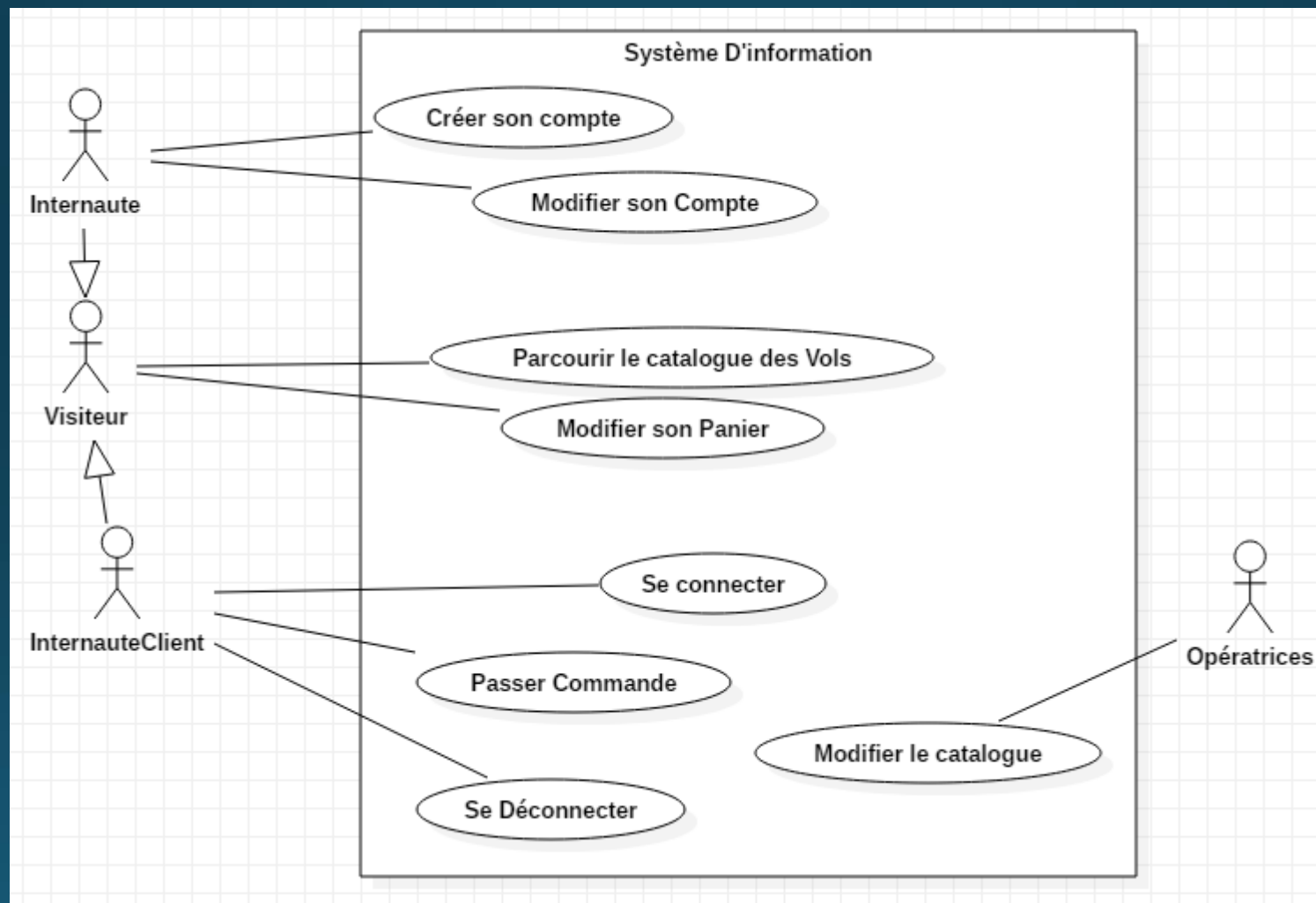
- Parcourir le catalogue des vols
- Modifier le panier

Si le diagramme de cas d'utilisation devient complexe avec beaucoup d'acteur et d'actions communes, le diagramme deviendrait illisible.

Nous avons donc la possibilité de créer un acteur plus généraliste, le Visiteur par exemple qui réalisera les actions communes.

Puis nous préciserons que Visiteur est la généralisation d'Internaute et Internaute Client.

## Use Case: Notions avancées





## Use Case: Notions avancées

C'est ce qu'on appelle la « Generalization », aussi appelé l'Héritage.

Internaute et InternauteClient héritent de visiteur et donc des actions :

- Parcourir le catalogue
- Modifier le panier

Nous pourrions aussi travailler les UsesCase avec cette généralisation

Exemple : Pour l'InternauteClient, nous allons lui donner la possibilité de payer avec plusieurs moyens de paiement.

## Use Case: Notions avancées

Dans le cas où il paye par carte bancaire, un nouvel acteur intervient « Partenaire Bancaire ».

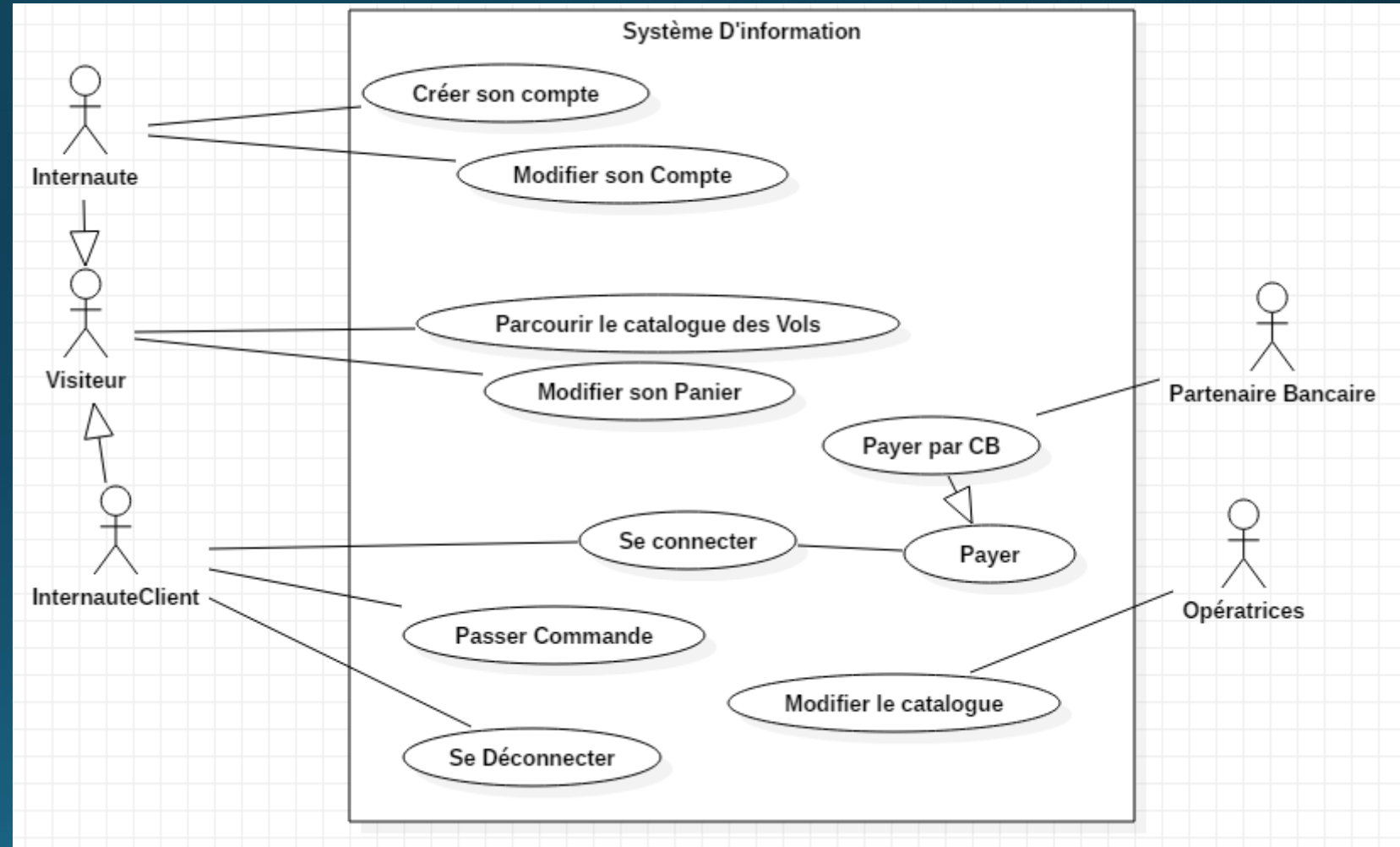
Mais il intervient que si le paiement est par carte bancaire.

Nous allons donc créer un nouveau UseCase qui sera un sous UseCase de Payer, c'est-à-dire un héritage entre UseCase

## Use Case: Notions avancées

Payer par CB est un UseCase Concret – C'est un processus spécifique

Payer est un UseCase Abstrait



## Use Case: Notions avancées

Include (inclusion) : Relation d'obligation entre les Cas d'utilisation.

- Exemple : Pour passer la commande l'InternauteClient est obligé de payer.

Extend (exclusion) : Relation Optionnelle entre les Cas d'utilisation.

- Exemple : Quand l'InternauteClient met à jour le panier il peut ou non parcourir le catalogue