



# ANALYSE ET CONCEPTION ORIENTEE OBJET

ISTA HAY RYAD  
2011/2012

## COURS 1

**Définir les besoins pour une solution logicielle**

Réalisé par :  
BOUROUS Imane

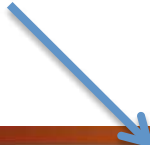


## PORTAIL DE LA FORMATION PROFESSIONNELLE AU MAROC

Télécharger tous les modules de toutes les filières de l'OFPPT sur le site dédié à la formation professionnelle au Maroc : [www.marocetude.com](http://www.marocetude.com)

Pour cela visiter notre site [www.marocetude.com](http://www.marocetude.com) et choisissez la rubrique :

### MODULES ISTA



HOME LIVRES **MODULES ISTA** ANNUAIRE ECOLES DOCTORAT LETTRE DE MOTIVATION NOUS CONTACTER SE CONNECTER

*Maroc Etude.Com* Connaissance - Métier - Technique

Annonces Google Emploi Maroc Messagerie Telecharger Un Jeu Maroc Annonces

recherche...

Nous avons 14 invités en ligne

**Annonces Google**

[Annonces Emploi Maroc](#)  
[Jeux Telecharger Gratuit](#)  
[Jeux PC En Ligne](#)

**Connexion**

Identifiant  
sniper

Mot de passe  
.....

Se souvenir de moi

**Connexion**

[Mot de passe oublié ?](#)  
[Identifiant oublié ?](#)

Notre Bibliothèque que ...Livres à Télé charger Gratuitement

**MacKeeper**

**-20%**

Complete your Purchase Now and save 20% Guaranteed with this Coupon Code

Apply Discount Automatically

"On ne jouit bien que de ce qu'on partage" [Madame de Genlis]

**Annonces Google**

[Jeu De Jeux](#)  
[Jeux Sur Internet](#)  
[Ecole Ingénieur](#)

**Dépanner et configurer votre réseau à domicile**

(Outil de Diagnostic)  
WI-FI / Ethernet  
Console de jeu  
Imprimante  
Messagerie

# Déterminer les exigences fonctionnelles

- En ingénierie, et plus particulièrement dans les procédures d'appel d'offres publiques et privées, *les exigences sont l'expression d'un besoin documenté sur ce qu'un produit ou un service particuliers devraient être ou faire.* Elles sont le plus souvent utilisées dans un sens formel dans l'ingénierie des systèmes et dans l'ingénierie logicielle.

# La phase exigence

La phase de développement des exigences peut avoir été précédée par une étude de faisabilité, ou une phase d'analyse conceptuelle du projet. La phase d'exigences peut être décomposée en :

- Mettre à jour les exigences: rassembler les exigences des parties prenantes;
- Analyser : vérifier la cohérence et l'exhaustivité ;
- Définir: écrire les exigences sous une forme aisément compréhensible pour les utilisateurs et les développeurs ;
- Spécifier: créer une interaction initiale entre les exigences et la conception.

# Exigences produit & exigences de processus

- Les projets sont soumis à trois sortes d'exigences :
- **Les exigences métier** qui décrivent le quoi dans les termes du métier. Elles décrivent ce qui doit être fourni ou réalisé pour produire de la valeur.
- **Les exigences produit** qui décrivent le produit ou le système à un haut niveau. Elles répondent aux exigences métier et sont couramment formulées comme les fonctionnalités que le système doit réaliser. On les appelle également exigences fonctionnelles ou spécifications fonctionnelles.
- **Les exigences de processus** qui décrivent le comment. Ces exigences prescrivent les processus que l'on doit suivre et les contraintes auxquelles on doit se conformer pour la réalisation du système. Dans ce cas, on trouve par exemple des exigences de sécurité, d'assurance qualité, ou de management.

# Processus de développement des exigences

## Rédaction des exigences

- Les exigences doivent être écrites de telle manière qu'elles orientent la création et la modification d'un système selon les règles métier (ou règles de gestion) appropriées au contexte et au domaine et dans lequel le système doit être utilisé.
- Les systèmes doivent normalement se conformer au domaine d'activité dans lequel ils sont exploités.

# Processus de développement des exigences

## Analyse des exigences

- Les exigences sont sujettes à des problèmes d'ambiguïté, d'imperfections, et d'incohérence. Des techniques telles qu'une inspection de logiciel rigoureuse ont été présentées pour aider à traiter de tels problèmes. Lorsque les ambiguïtés, les imperfections, et les incohérences sont résolues dans la phase d'exigences, l'ordre de grandeur du coût de correction est moins élevé que lorsque ces mêmes problèmes se retrouvent dans des étapes ultérieures de développement du produit. L'analyse des exigences s'efforce de résoudre ces problèmes.



# Les intervenants d'un projet informatique

- Plusieurs intervenants participent à l'analyse, la conception et le développement d'un projet informatique, parmi lesquels :
- **Le client**: Le client est l'organisme auquel est destiné le projet, c'est celui le donneur d'ordre et le payeur de la prestation. Le client peut être une entreprise qui fait appel à une SSII pour réaliser le projet ou le service de l'entreprise qui fait appel à la direction informatique
- **Le prestataire** Le prestataire est l'organisme qui réalise le projet. Le prestataire peut être une entreprise externe spécialisée (SSII) ou le service informatique de l'entreprise.



# *La Maîtrise d'œuvre*

- La Maîtrise d'œuvre est la responsabilité de l'exécution du projet. Elle **représente le prestataire** tout au long du projet.
- La Maîtrise d'œuvre est le garant du respect des engagements pris notamment sur les délais et les contenues des fournitures.
- Il assure le pilotage technique du projet, la gestion de l'équipe de production l'affectation des tâches et la mise en œuvre des dispositions d'assurance qualité.

# La maîtrise d'ouvrage

La maîtrise d'ouvrage assure la conformité du projet vis-à-vis de la demande du client. Elle **représente le client** tout au long du projet, elle a pour rôle de :

- Veiller au respect des objectifs généraux du projet
- Assurer la conduite générale du projet
- Gérer les enveloppes financières
- Valider les documents relatifs au projet ainsi que les maquettes
- Éventuellement, préparer et exécuter les tests de réception des applications
- Prononcer les recettes

C'est au sein de la maîtrise d'ouvrage que l'on trouve les experts métier et les groupes de validation. Lorsqu'il existe un service d'organisation dans l'entreprise, celle-ci fréquemment chargée de la maîtrise d'ouvrage, à défaut elle peut représenter les utilisateurs auprès de celle-ci.

# Le directeur du projet ou chef de projet

- Le Directeur ou chef de projet est le responsable de la mise en œuvre du projet dans le cadre du cahier des charges établi.
- Il est chargé d'étudier les besoins des utilisateurs, de définir des solutions adaptées et après validation de les mettre en œuvre avec les outils informatiques retenus.
- Il s'appuie sur le Groupe de Pilotage et travaille en étroite collaboration avec le responsable utilisateur.
- Il dirigera l'équipe affectée au projet. Il veillera au respect des délais, à la qualité du travail et à l'établissement des critères de réception du projet.
- Il a pour rôle d'assurer la coordination de l'ensemble des acteurs du projet
- On désigne généralement le maître d'œuvre comme directeur du projet, parfois le maître d'ouvrage.

# Le responsable qualité

Le responsable qualité est choisi en commun accord entre le maîtrise d'oeuvre et la maîtrise d'ouvrage .Il a le rôle de :

- Définir les dispositions d'assurance qualité formalisées dans le plan d'assurance qualité.
- Veiller à la mise en application de ces dispositions.
- Définir les actions correctives si les dispositions ne sont appliquées.
- Vérifier et rendre compte de la mise en application de ces actions.

# Les « utilisateurs »

Les utilisateurs sont les destinataires finaux du projet. Ils participent au projet sous la responsabilité du maître d'ouvrage. Le rôle des utilisateurs est important en particulier au niveau de :

- L'expression des besoins.
- Les tests de recette.
- La mise en service du projet.

# Les fournisseurs

Un certain nombre d'éléments indispensables à l'exécution du projet peuvent être obtenus auprès des fournisseurs autres que le prestataire.

Ces fournisseurs peuvent fournir des matériels, logiciels, des ressources humaines et des services.

Il est recommandé de définir :

- Les relations contractuelles avec les fournisseurs
- L'entité qui porte la responsabilité du choix du fournisseur
- L'entité qui porte la responsabilité du contrôle de l'exécution du contrat
- Les dispositions financières associées

# La modélisation du système





# Introduction à la modélisation

- Pour modéliser un système informatique on se sert d'un langage de modélisation. Comme c'est souvent le cas en informatique plusieurs choix apparaissent sur le marché. L'un d'eux s'est cependant nettement démarqué depuis les années 90 : il s'agit de UML (Unified Modeling Language) né de la fusion de trois méthodologies de modélisation.
- UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue.

# UML

- UML unifie à la fois les notations et les concepts orientés objet. Il ne s'agit pas d'une simple notation, mais les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage.
- UML a une dimension symbolique et ouvre une nouvelle voie d'échange de visions systémiques précises. Ce langage est certes issu du développement logiciel mais pourrait être appliqué à toute science fondée sur la description d'un système. Dans l'immédiat, UML intéresse fortement les spécialistes de l'ingénierie système.

# Introduction

## ◆ UML ?

UML = Langage de modélisation

## ◆ A quoi sert?

Définir un modèle logique sur lequel se fonde le projet

## ◆ Types de diagrammes en UML?

- Statique ( ce que le système est)
- Dynamique (comment le système évolue)
- Fonctionnel (ce que le système fait)

# Introduction

## Statique (ce que le système EST)

- diagramme de classes
- diagramme d'objets
- diagramme de composants
- diagramme de déploiement

## Dynamique

(comment le système  
EVOLUE)

- diagramme de séquence
- *diagramme de collaboration*
- diagramme d'états-transitions
- diagramme d'activités

## Fonctionnel

(ce que le système FAIT)

- diagramme de cas d'utilisation
- *diagramme de collaboration*

# Plan

## Définition Diagramme de cas d'utilisation

- Définition
- Intérêt

## Éléments de base

- Acteurs du système
- Cas d'utilisation

## Relations entre éléments de base

- Entre Cas d'utilisation
- Entre Cas d'utilisation et acteurs
- Entre acteurs

# Définition Diagramme de cas d'utilisation

- ❖ Expression du comportement du système (actions et réactions), selon le point de vue de l'utilisateur
- ❖ Décrit le système et les relations entre le système et l'environnement
- ❖ Intérêts:
  - Permettent de délimiter les frontières du système
  - Constituent un moyen d'exprimer les besoins d'un système
  - Utilisés par les utilisateurs finaux pour exprimer leurs attentes et leurs besoins
  - Permettent d'impliquer les utilisateurs dès les premiers stades du développement
  - Constituent une base pour les tests fonctionnels

# Définition Diagramme de cas d'utilisation

- **Cas d'utilisation (principes)**

Ce que le système doit faire (comportement souhaité)

- **Mais pas comment réaliser ce comportement**

Pas de détails de programmation, mise en œuvre, etc.

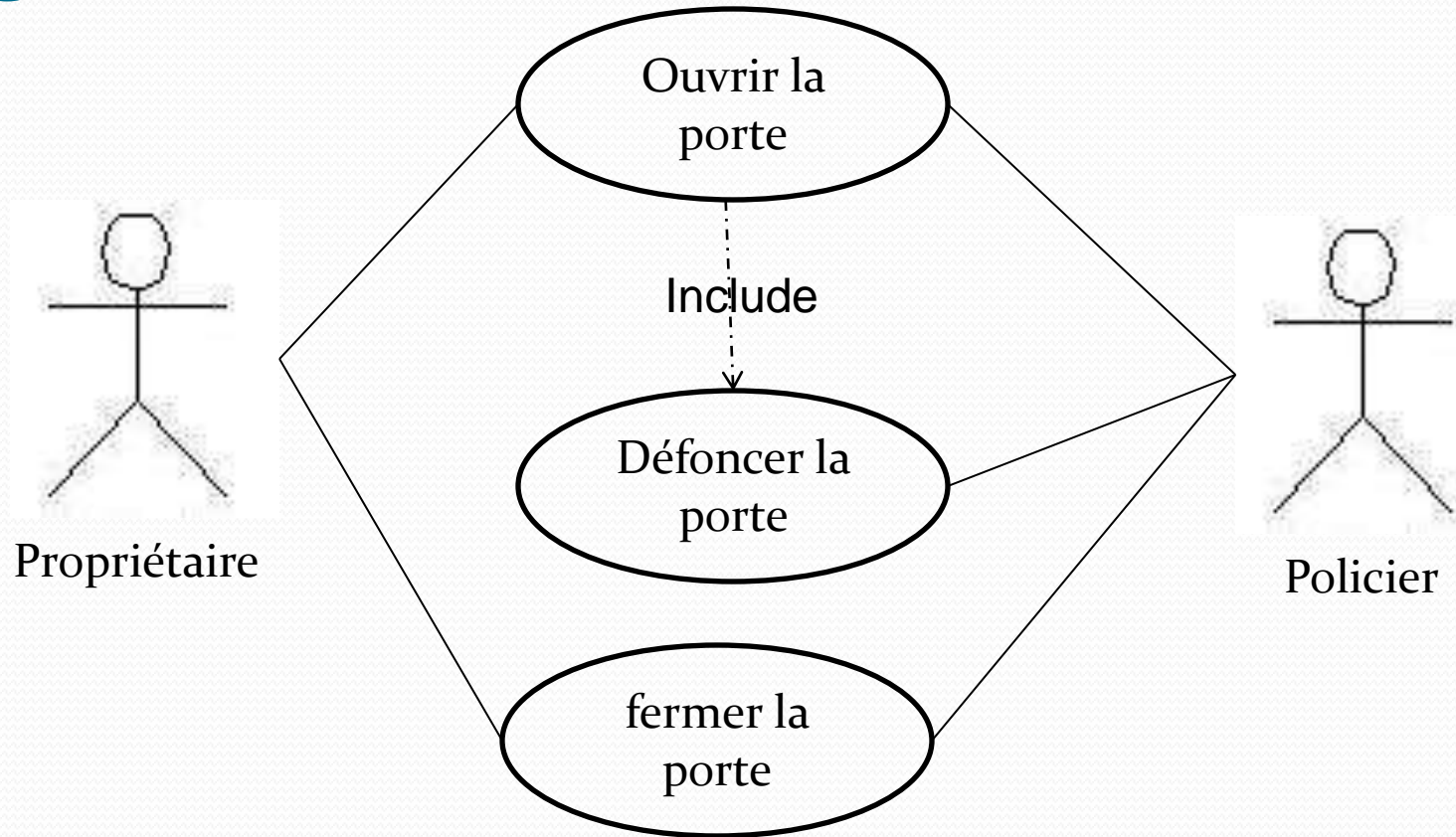
Indépendant de la réalisation

- **Un outil pour communiquer**

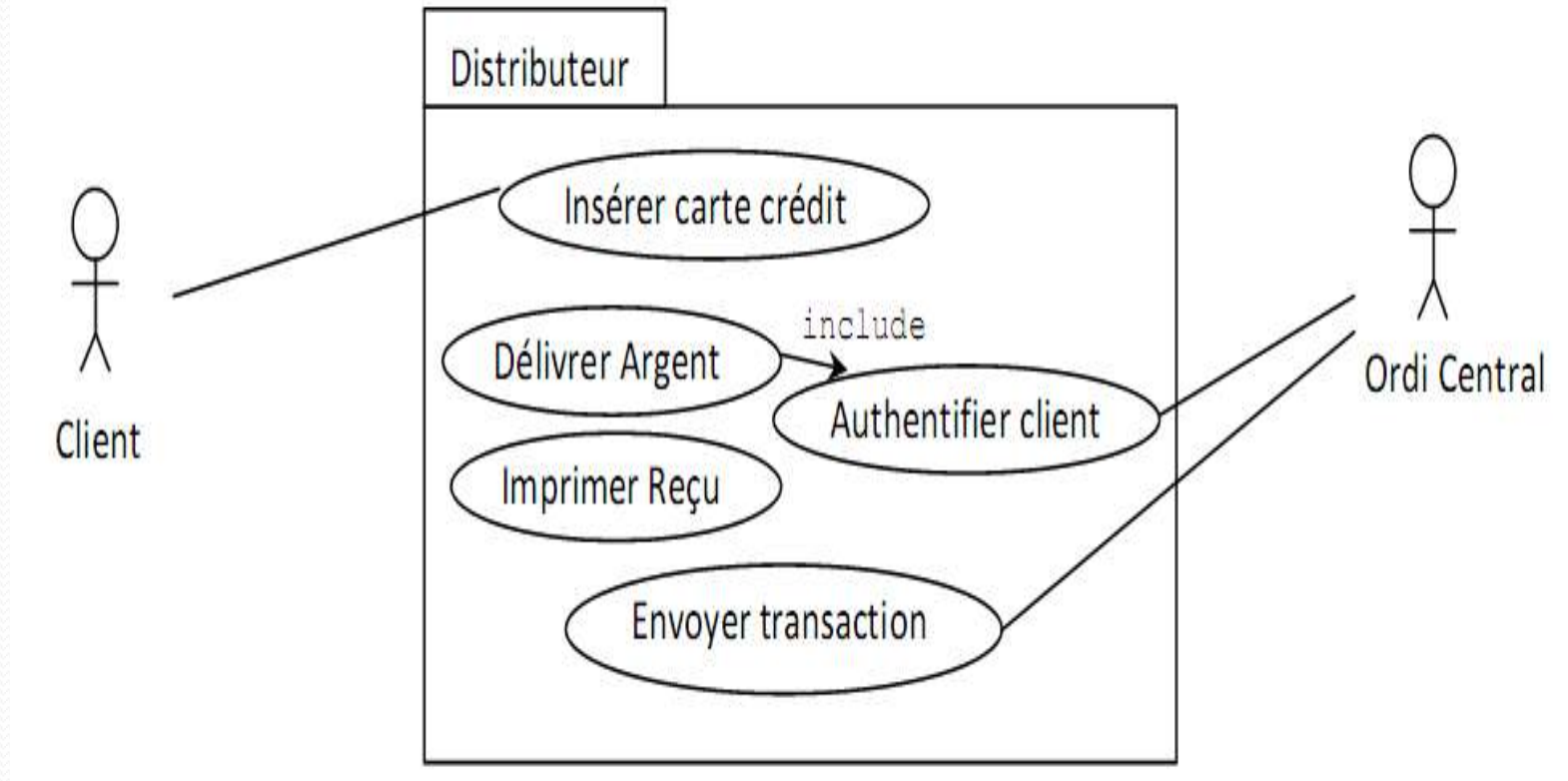
Utilisateur final / expert du domaine <---> concepteur / développeur



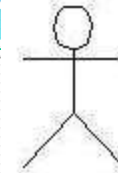
# Diagramme de cas d'utilisation



# Diagramme de cas d'utilisation



# Éléments de base



## Acteurs du système

- Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié.
- Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données.

# Éléments de base

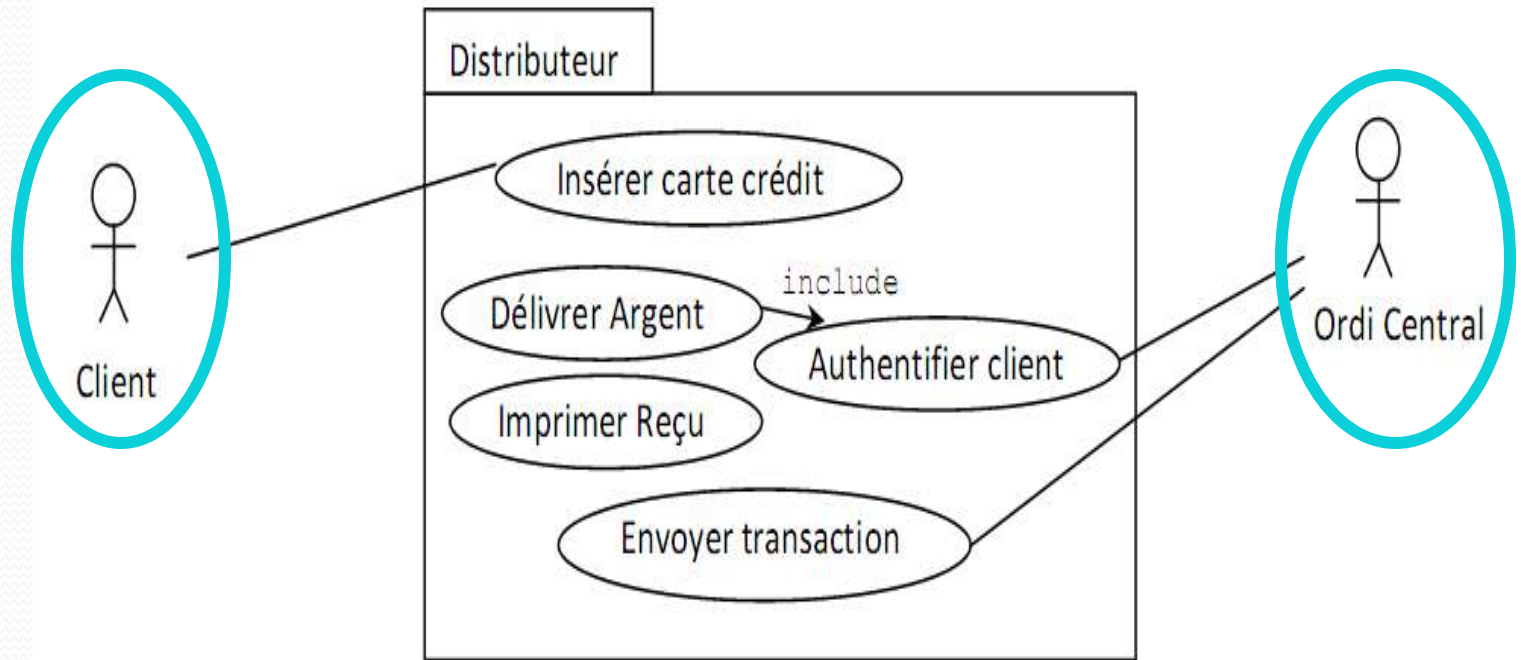
## Acteurs du système

- Quels sont les utilisateurs qui ont besoins du système pour réaliser le travail ?
- Quels sont les utilisateurs qui vont effectuer les fonctions principales du système ?
- Quels sont les utilisateurs qui vont exécuter les fonctions principales du système (maintenance et administration) ?
- Est-ce que le système interagit avec du matériel ou d'autres logiciels ?

# Éléments de base

## Acteurs du système

- Exemple distributeur



# Éléments de base

## Cas d'utilisation (use case)

- Un cas d'utilisation (use case) représente un ensemble de séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier.
- Un cas d'utilisation modélise un service rendu par le système. Il exprime les interactions acteurs/système et apporte une valeur ajoutée « notable » à l'acteur concerné.

# Éléments de base

Cas d'utilisation (use case)

## Identification des « use cases » :

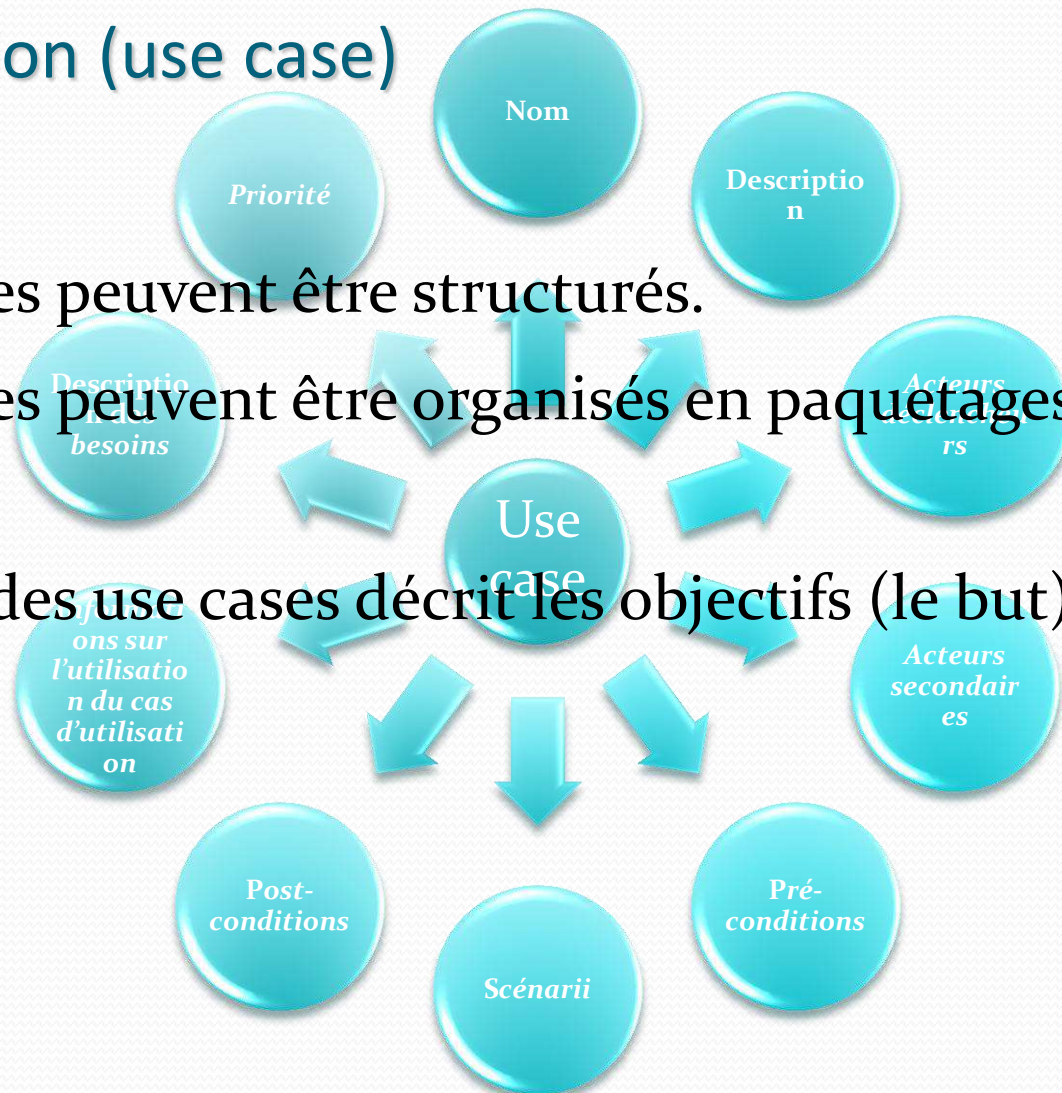
- Quelles sont les fonctions demandées par l'acteur du système?
- Le système mémorise t-il de l'information?
- Quel acteur créera, lira ou mettra à jour l'information ?
- Le système doit il informer un acteur extérieur que son état interne a changé?
- Y a t-il des événements externes que le système doit connaître? Quel acteur informe le système de ces événements?



# Éléments de base

## Cas d'utilisation (use case)

- ◆ Les uses cases peuvent être structurés.
- ◆ Les uses cases peuvent être organisés en paquetages (packages).
- ◆ L'ensemble des use cases décrit les objectifs (le but) du système.



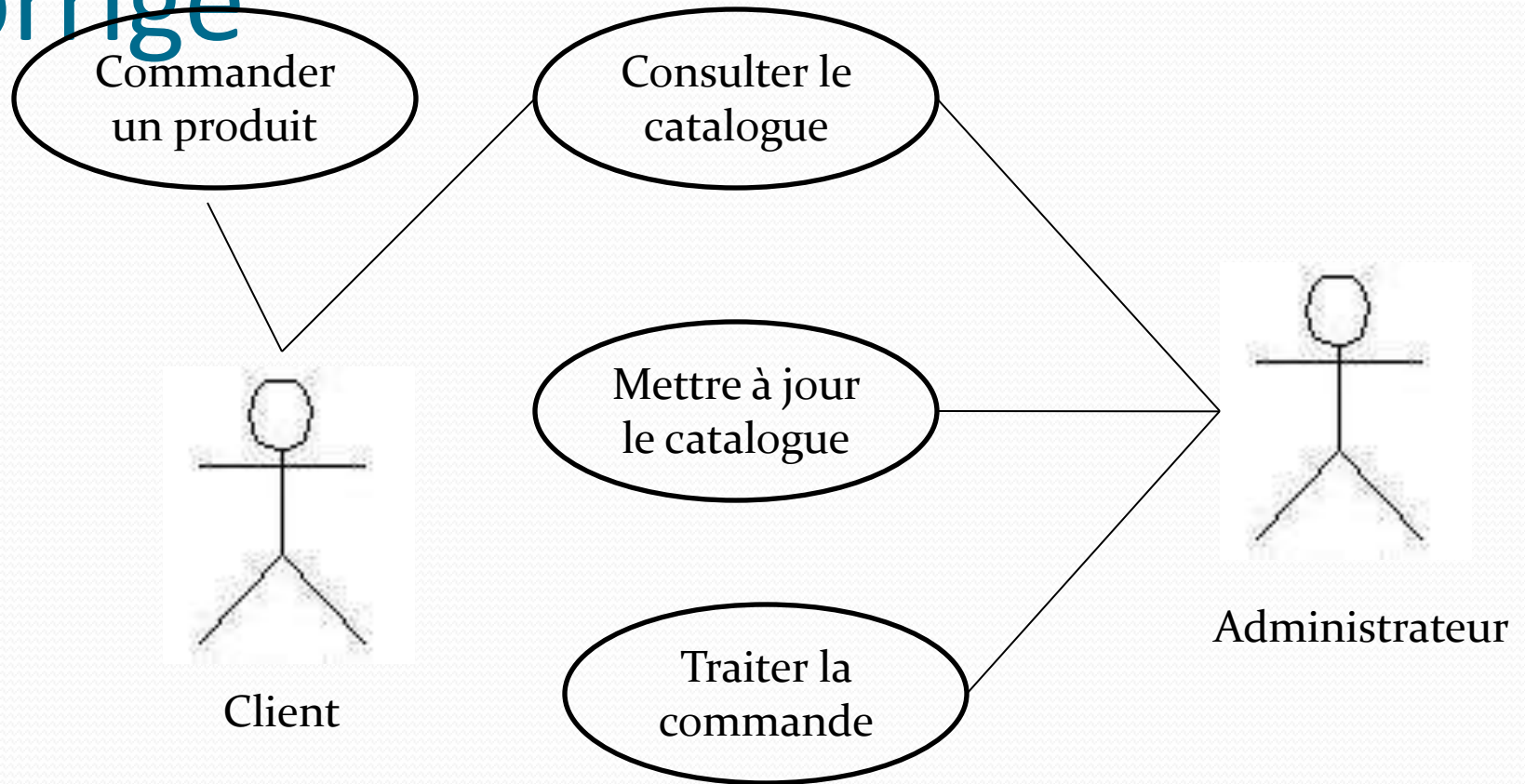
# Exercice d'application

- France Télécom désire disposer d'une plateforme d'achat en ligne où les clients peuvent consulter ou/et commander les produits qui sont disponibles sur un catalogue.
- Les administrateurs de FT sont responsables de la mise à jour des produits et du traitement des commandes effectués par les clients.

# Corrigé

- Identification des acteurs :
  - Administrateur
  - Client
- Identification des cas d'utilisations :
  - Consulter le catalogue
  - Commander un produit
  - Mettre à jour le catalogue
  - Traiter une commande

# Corrigé



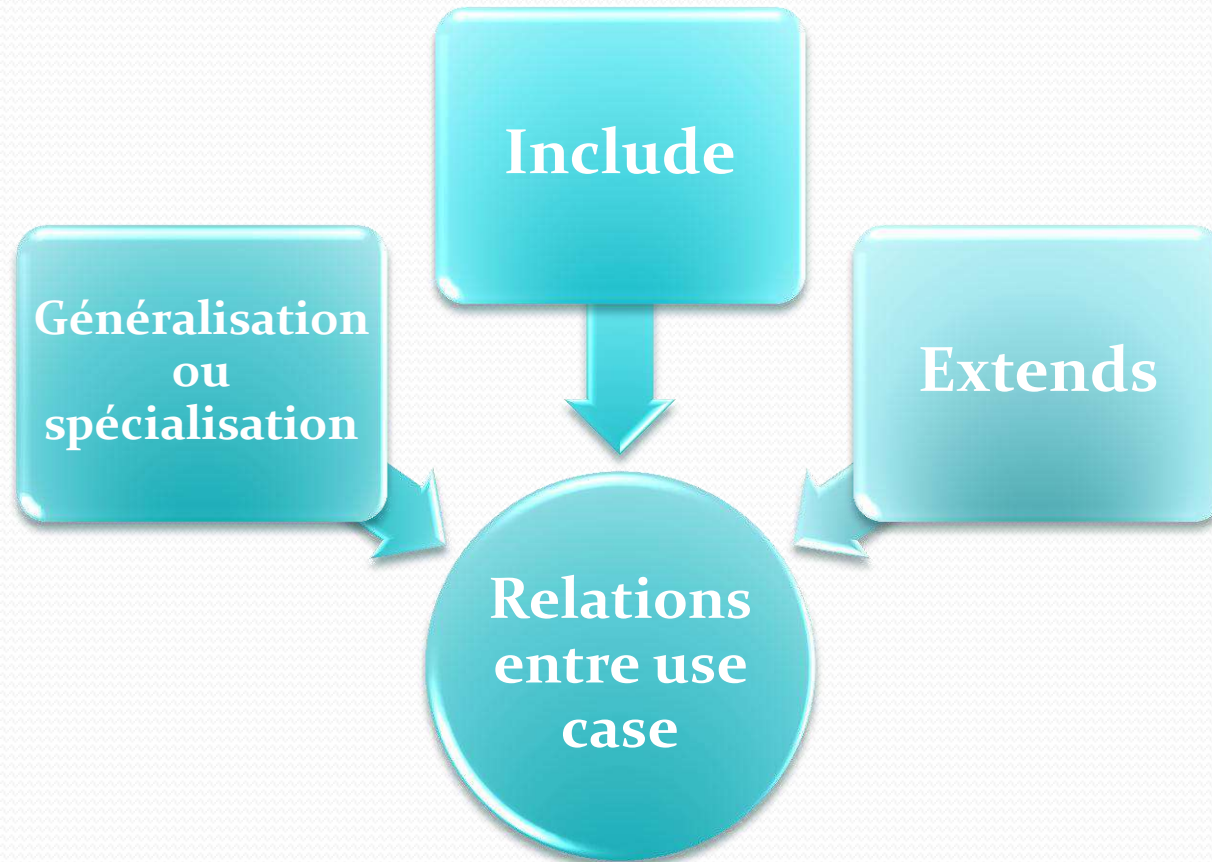
# Corrigé

Description textuelle du cas d'utilisation

- **Nom du cas d'utilisation : Mettre à jour le catalogue**
- **Acteurs déclencheurs : Administrateur.**
- **Contexte de déclenchement: l'acteur sélectionne la mise à jour d'un catalogue.**
- **Pré-conditions : l'acteur s'est connecté au système et choisit un catalogue**
- **Description du UC : permet à l'acteur de gérer un catalogue à savoir : l'ajout, la modification et la suppression du catalogue.**
- **Post-conditions : le catalogue est mise à jour.**
- **Scénario nominal :**
  1. Connexion au système ;
  2. Sélection d'un catalogue
  3. Mise à jour du catalogue.

# Relations entre éléments de base

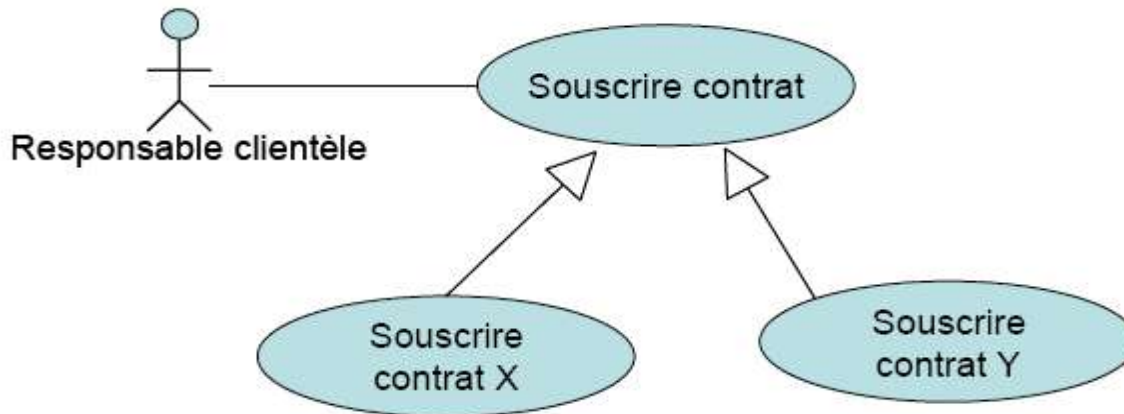
## Entre Cas d'utilisation



# Relations entre éléments de base

## Entre Cas d'utilisation

### ◆ Général



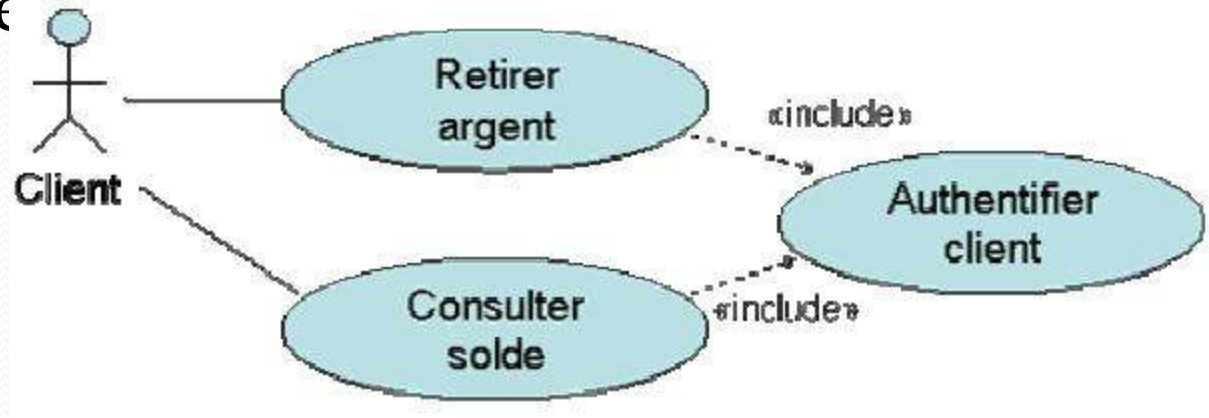
- Les cas d'utilisation descendants héritent de la description de leur parent commun. Chacun d'entre eux peut néanmoins comprendre des interactions spécifiques supplémentaires.



# Relations entre éléments de base

## Entre Cas d'utilisation

### ◆ Include

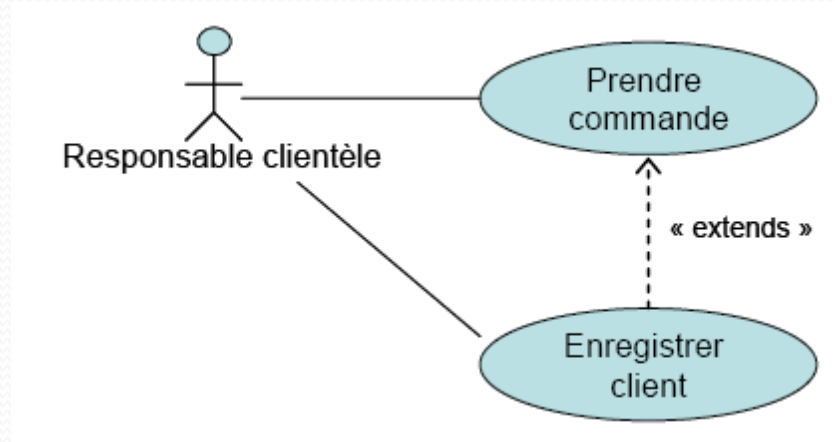


- Le cas d'utilisation contient un autre cas d'utilisation

# Relations entre éléments de base

## Entre Cas d'utilisation

### ◆ Extends :



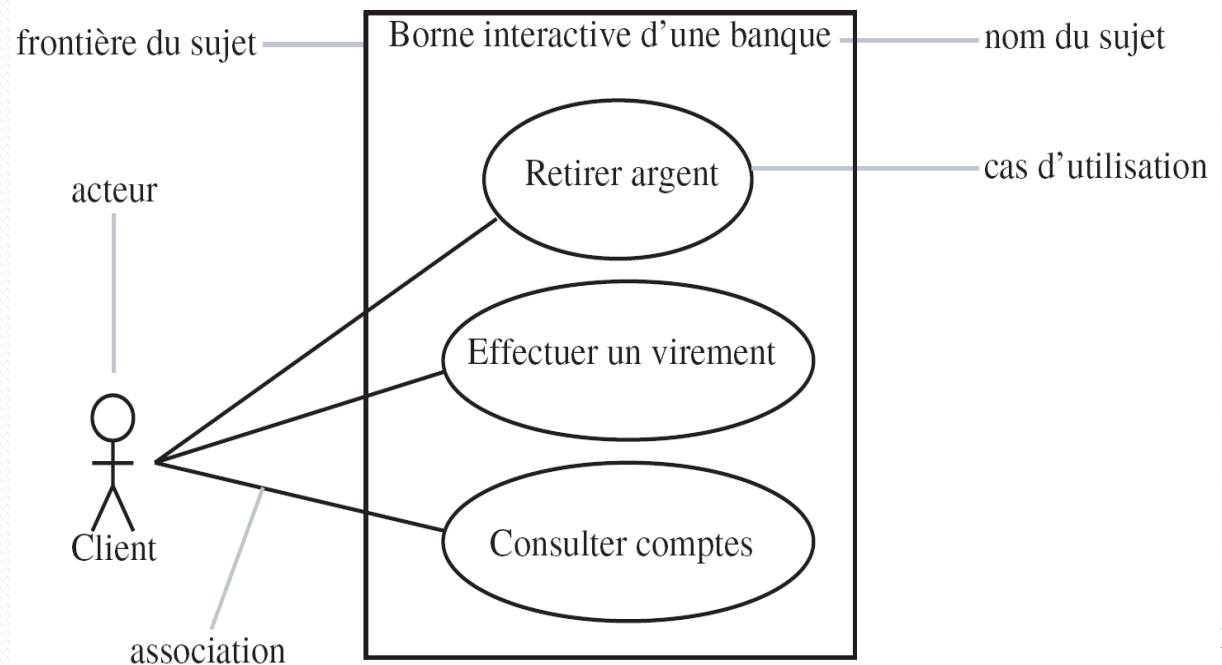
- Le cas d'utilisation étend (précise) les objectifs (le comportement) d'un autre cas d'utilisation.
- On dit qu'un cas d'utilisation X étend un cas d'utilisation Y lorsque le cas d'utilisation X peut être appelé au cours de l'exécution du cas d'utilisation Y.

# Relations entre éléments de base

## Entre Cas d'utilisation et acteurs

- **Relation d'association**

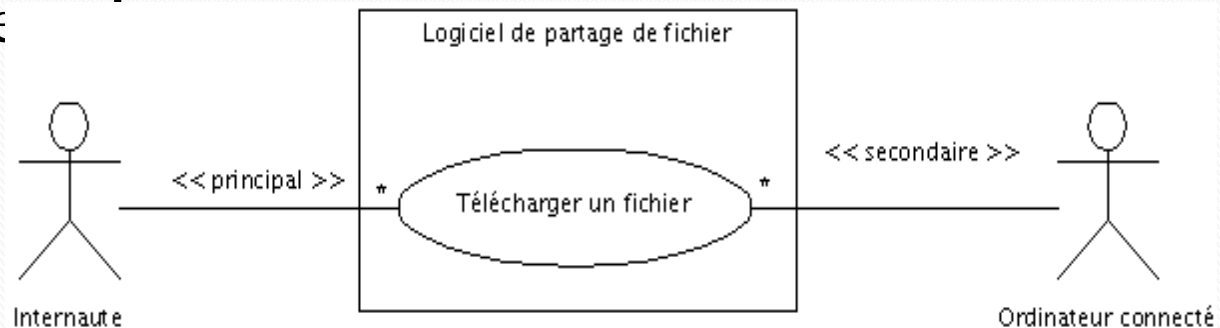
Une relation d'association est chemin de communication entre un acteur et un cas d'utilisation et est représenté par un trait continu



# Relations entre éléments de base

## Entre Cas d'utilisation et acteurs

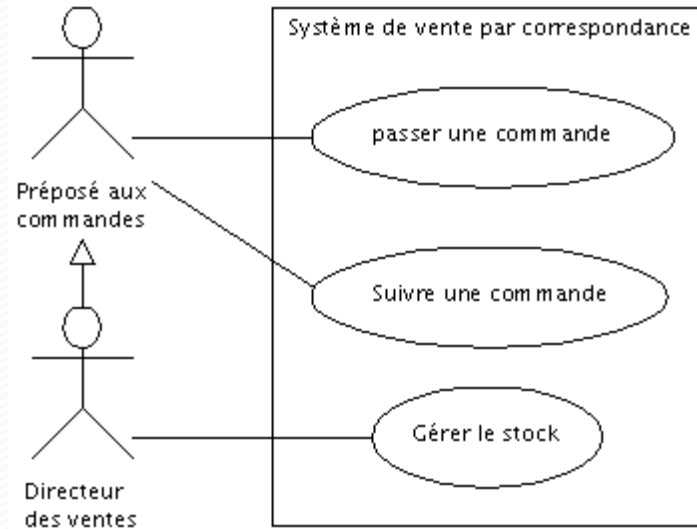
- Multiplicité: Lorsqu'un acteur peut interagir plusieurs fois avec un cas d'utilisation, il est possible d'ajouter une multiplicité sur l'association du côté du cas d'utilisation;
- Acteur principal: Le cas d'utilisation rend service à cet acteur.
- Acteur secondaire : est sollicité pour des informations complémentaires.



# Relations entre éléments de base

## Entre Cas d'utilisation et acteurs

- La Généralisation : un acteur A est une généralisation d'un acteur B si l'acteur A peut être substitué par l'acteur B. Dans ce cas, tous les cas d'utilisation accessibles à A le sont aussi à B, mais l'inverse n'est pas vrai.



# Remarques

## Concernant les relations dans les cas d'utilisation

- Il est important de noter que l'utilisation des relations n'est pas primordiale dans la rédaction des cas d'utilisation et donc dans l'expression du besoin.
- Ces relations peuvent être utiles dans certains cas mais une trop forte focalisation sur leur usage conduit souvent à une perte de temps ou à un usage faussé, pour une valeur ajoutée, au final, relativement faible.

# Remarques

## Concernant les cas d'utilisation

- les diagrammes de cas d'utilisation ne peuvent être qualifiés de modélisation à proprement parler. D'ailleurs, de nombreux éléments descriptifs sont en langage naturel. De plus, ils ne correspondent pas exactement à une approche objet.
- En effet, capturer les besoins, les découvrir, les réfuter, les consolider, etc., correspond plus à une analyse fonctionnelle classique.

# Conclusion

- L'objectif poursuivi par les cas d'utilisation est de permettre de décrire, dans des documents lisibles par tous, la finalité des interactions du système et de ses utilisateurs.

**Ils ne doivent pas chercher l'exhaustivité, mais clarifier, filtrer et organiser les besoins !**

- Après avoir rédigé les cas d'utilisation, il faut identifier des objets, des classes, des données et des traitements qui vont permettre au système de supporter ces cas d'utilisation.