

DIAGRAMME DE CAS D'UTILISATIONS

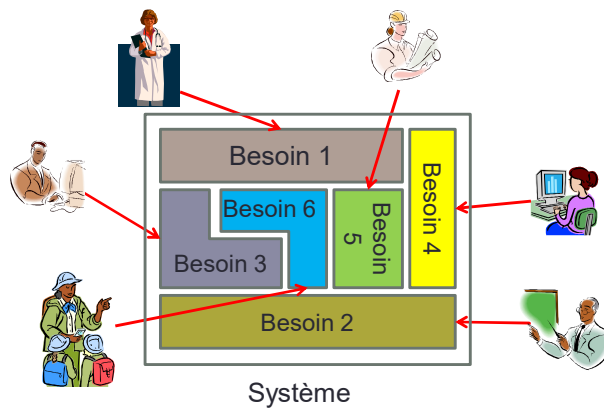
Chapitre 2

Introduction

- ❑ Les cas d'utilisations permettent de recueillir, d'analyser et d'organiser les besoins, et de recenser les grandes fonctionnalités d'un système.
- ❑ Un diagramme de cas d'utilisation capture le comportement d'un système du point de vue de l'utilisateur.
- ❑ Ils décrivent le comportement du système sous forme d'actions et de réactions.

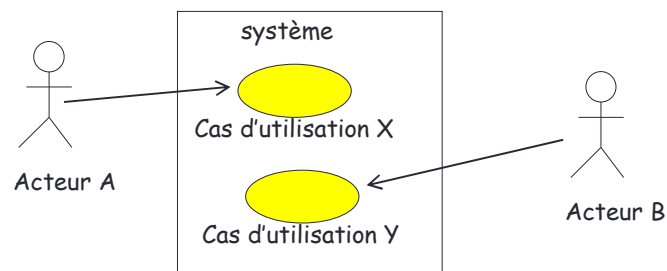
L'utilité

- ❑ Les cas d'utilisations partitionnent l'ensemble des besoins d'un système.



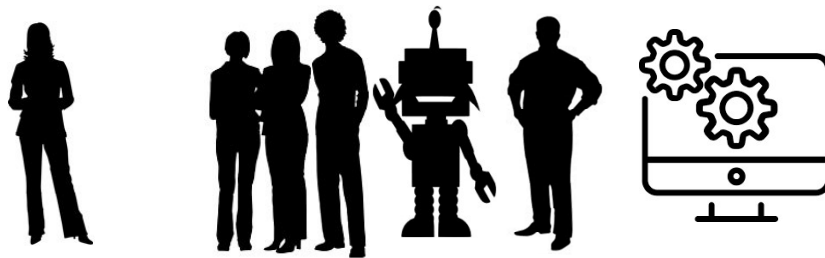
Le modèle des cas d'utilisations

- ❑ Le modèle des cas d'utilisation se compose de:
 - Acteurs: représenté par un petit bonhomme.
 - Système : représenté par un rectangle.
 - Cas d'utilisation eux-mêmes: représentés par des ovales.



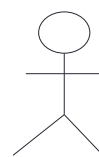
Notion d'acteur

- ❑ Un acteur joue le rôle d'une personne, d'un système ou d'un autre logiciel qui interagit avec le système en cours de conception.



Notion d'acteur

- ❑ Il se représente par un petit bonhomme avec son nom inscrit dessous.
- ❑ Il est également possible de représenter un acteur sous la forme d'une classe avec le stéréotype << actor >>
- ❑ **Remarque :**
 - un stéréotype est une annotation qui permet d'adapter un concept UML pour une situation particulière



Client

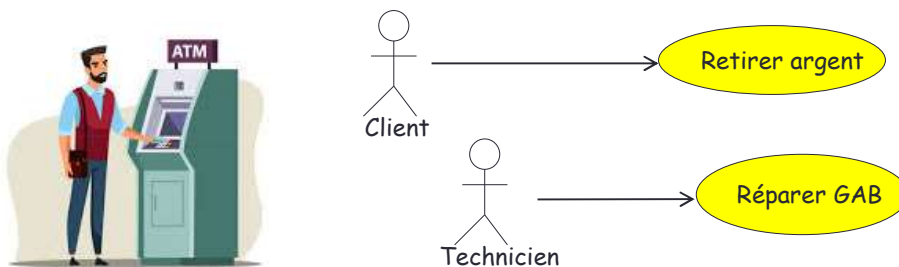


Notion d'acteur

- ❑ Il se détermine en observant les utilisateurs directs du système :
 - Ceux responsables de son exploitation ou
 - De sa maintenance ainsi que
 - Les autres systèmes qui interagissent avec le système.

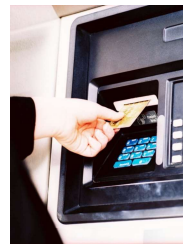
Notion d'acteur

- ❑ Remarque :
 - La même personne physique peut jouer le rôle de plusieurs acteurs (vendeur, client)
- ❑ Exemple :



Notion de Cas d'utilisation

- ❑ Un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur.
- ❑ Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie.
- ❑ Exemple d'un système gérant une banque:
 - Retirer argent
 - Effectuer virement
 - Consulter compte



Notion de Cas d'utilisation

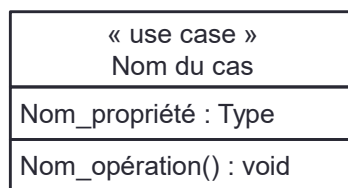
- ❑ Un cas d'utilisation se représente par une ellipse contenant le nom du cas (un verbe à l'infinitif), et optionnellement un stéréotype.



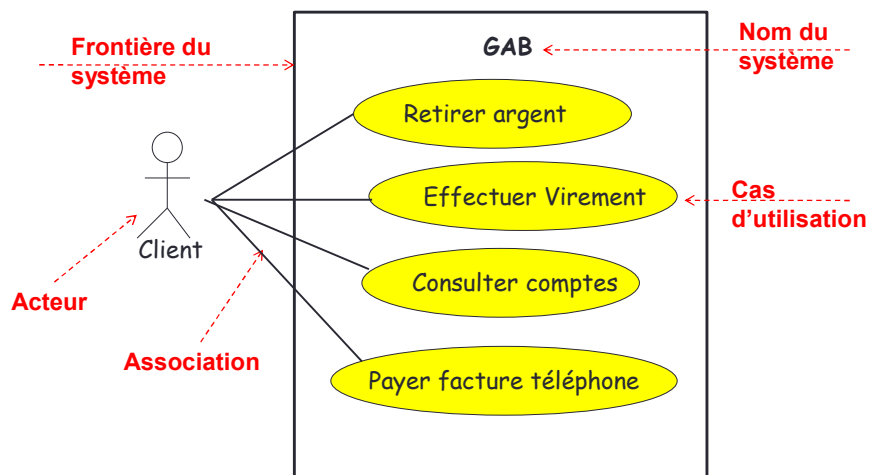
Notion de Cas d'utilisation

Remarque :

- Un cas d'utilisation peut éventuellement être représenté sous la forme d'une classe avec le stéréotype << use case >> .
- Dans ce cas certains attributs et opérations peuvent être affichés.



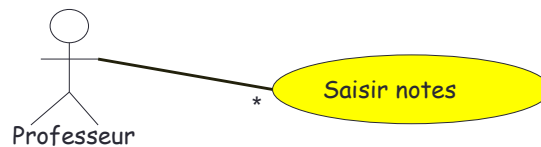
Représentation du diagramme



Relations Acteur - cas d'utilisation

❑ La relation d'association :

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



- Remarque :
 - Le symbole « * » signifie que l'acteur peut interagir plusieurs fois avec le cas d'utilisation : c'est une multiplicité

Relations Acteur - cas d'utilisation

▪ Exemples de multiplicités :

- * : plusieurs
- n..m : de n à m fois
- N : n fois exactement
- 1..* : au moins une fois

Relations Acteur - cas d'utilisation

- Catégories d'acteurs :

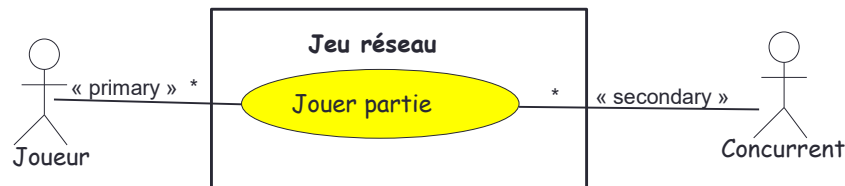
- Lorsqu'un cas d'utilisation rend service principalement à un acteur alors cet acteur est qualifié de principal :

→ stéréotype « **primary** »

- Les autres acteurs sont alors qualifiés de secondaires :

→ stéréotype <<

secondary >>



Relations Acteur - cas d'utilisation

- Remarques :

- Un cas d'utilisation a au plus un acteur principal.
- Un acteur principal obtient un résultat observable du système tandis qu'un acteur secondaire est sollicité pour des informations complémentaires.
- En général, l'acteur principal initie le cas d'utilisation par ses sollicitations.

Relations Acteur - cas d'utilisation

- Cas d'utilisation interne :
 - Quand un cas n'est pas directement relié à un acteur, il est qualifié de cas d'utilisation interne.

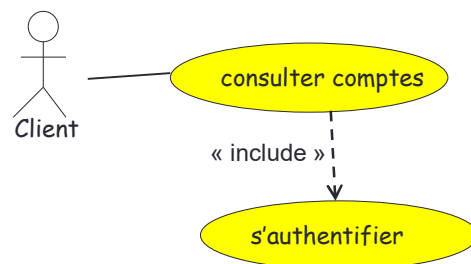
Relations entre cas d'utilisation

- ▣ Relation d'inclusion :
 - Un cas A inclut un cas B si le comportement décrit par le cas A inclut le comportement du cas B : le cas A dépend de B.
 - Lorsque A est sollicité, B l'est obligatoirement, comme une partie de A.
 - Cette dépendance est symbolisée par le stéréotype « **include** »

Relations entre cas d'utilisation

❑ Relation d'inclusion :

- **Exemple** : pour consulter ses comptes bancaires sur Internet , un internaute doit s'authentifier



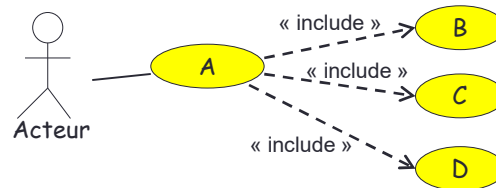
Relations entre cas d'utilisation

❑ Relation d'inclusion :

- Remarque 1:
 - Les deux cas d'utilisation précédentes peuvent être rassemblés en une Mais on les a séparé pour pouvoir réutiliser l'authentification pour d'autres cas qui en auraient besoin.
- Attention :
 - Les cas d'utilisation ne s'enchaînent pas, puisqu'il n'y a aucune représentation temporelle dans un diagramme de cas d'utilisation.

Relations entre cas d'utilisation

- Remarque 2:
- Soit le diagramme suivant :

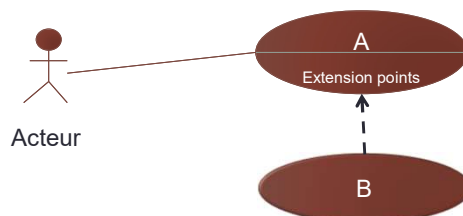


- Ce diagramme ne nous informe pas sur l'enchaînement des sous-cas d'utilisation B,C,D.
- Pour indiquer cet enchaînement il faut lui associer un diagramme d'activité.

Relations entre cas d'utilisation

Relation d'extension

- Un cas d'utilisation B étend un cas d'utilisation A lorsque B est éventuellement appelé à la suite de l'exécution du cas A.

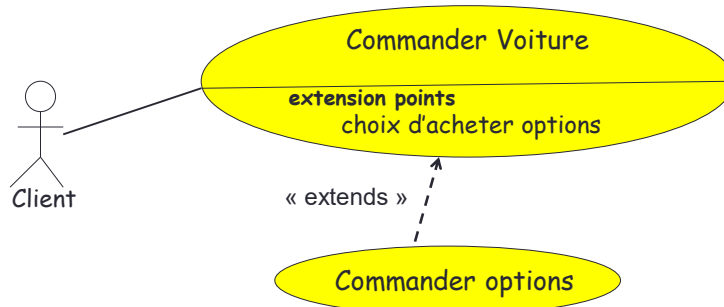


- Une extension est souvent soumise à condition à un point précis du cas étendu qu'on appelle point d'extension.

Relations entre cas d'utilisation

Exemple :

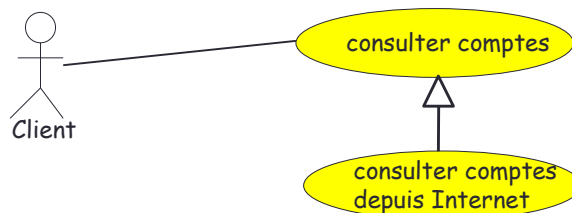
- Un client qui désire commander une voiture peut choisir de commander les options aussi ou non.



Relations entre cas d'utilisation

Relation de généralisation :

- La relation de généralisation correspond au concept d'héritage dans les langages orientés objet.
- Exemple :

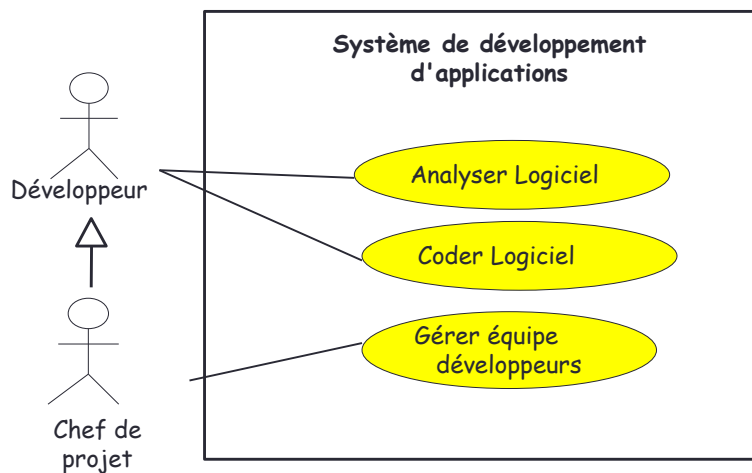


Relations entre acteurs

❑ La relation de généralisation :

- C'est la seule relation possible entre deux acteurs
- 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.

Relations entre acteurs



Identification des Acteurs



- ❑ Les acteurs d'un système sont les entités externes à ce système qui interagissent avec lui.
- ❑ Ils peuvent être :
 - Humains,
 - Matériels
 - Logiciels.
- ❑ Les acteurs identifiés doivent être décrits d'une manière claire et concise en 3 à 4 lignes max.

Identification des Acteurs



- ❑ Pour trouver les acteurs d'un système, il faut :
 - identifier les différents rôles que vont devoir jouer ses utilisateurs (ex : responsable clientèle, responsable d'agence, administrateur, approbateur, ...).
 - s'intéresser aux autres systèmes avec lesquels le système va devoir communiquer.

Identification des cas d'utilisation

- ❑ Un cas d'utilisation correspond à une fonction métier du système, selon le point de vue de l'acteur
- ❑ Les cas d'utilisation se déterminent en observant et en précisant, acteur par acteur, les séquences d'interaction - les scénarios – du point de vue de l'utilisateur.
- ❑ Les cas d'utilisation sont des abstractions du dialogue entre les acteurs et le système.



Identification des cas d'utilisation

- ❑ Pour identifier les cas d'utilisation, il faut se placer du point de vue de chaque acteur et déterminer comment et surtout pourquoi il se sert du système.
- ❑ Exemple :
 - Le cas d'utilisation « retirer argent » est correct
 - Le cas d'utilisation « ~~distribuer argent~~ » est incorrect (car point de vue système)



Description des cas d'utilisation

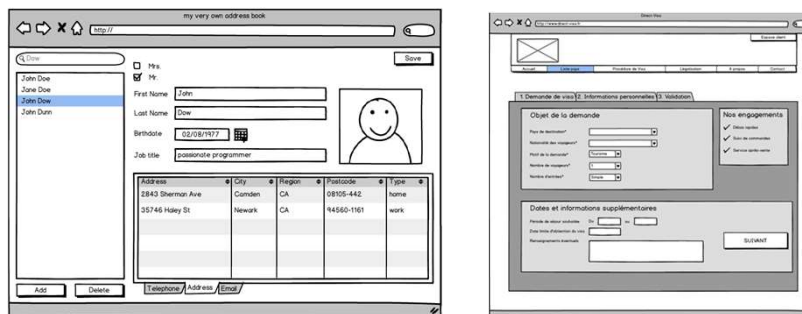
- Une description textuelle couramment utilisée se compose de trois parties :
 - Partie 1 : Informations sur le cas d'utilisation
 - **Nom** : Utiliser une tournure à l'infinitif (ex : Réceptionner un colis).
 - **Objectif** : Une description résumée permettant de comprendre l'intention principale du cas d'utilisation.
 - **Acteurs principaux** : Ceux qui vont réaliser le cas d'utilisation
 - **Acteurs secondaires** : Ceux qui ne font que recevoir des informations à l'issue de la réalisation du cas d'utilisation
 - **Dates** : La date de création ou de la mise à jour.
 - **Responsable** : Le nom des responsables.
 - **Version** : Le numéro de version.

Description des cas d'utilisation

- Partie 2 :
 - **Les pré-conditions** : l'état dans lequel le système doit être (l'application) avant que ce cas d'utilisation puisse être déclenché.
 - **Des scénarios** : Ces scénarios sont décrits sous la forme d'échanges d'évènements entre l'acteur et le système. On distingue :
 - **Scénario nominal** : se déroule quand il n'y a pas d'erreur,
 - **Scénarios alternatifs** : sont les variantes du scénario nominal,
 - **Scénarios d'exception** : décrivent les cas d'erreurs.
 - **Des post-conditions** : Elle décrivent l'état du système à l'issue des différents scénarios.

Description des cas d'utilisation

- Partie 3 (optionnelle):
 - Spécifications non fonctionnelles (spécifications techniques, ...).
 - Description des besoins en termes d'interfaces graphique (maquettes)



Logiciels pour réaliser les maquettes



Problème (1/5)

- ❑ Nous souhaitons construire une application informatique de gestion et de suivi des incidents techniques d'un parc informatique d'une grande société. Ce parc est géré par une équipe de techniciens qui sont chargés de la maintenance matérielle des équipements informatiques (ordinateurs, imprimantes,...) ainsi que de la maintenance logicielle (installer de nouveaux logiciels, les réinstaller, etc....). Cette équipe est gérée par un superviseur système qui travaille comme technicien et superviseur en même temps

Problème (2/5)

- ❑ Les processus métier associé à cette application sont comme suit : Lorsqu'un matériel informatique tombe en panne logicielle ou matérielle, l'opérateur qui l'utilise se connecte à notre application et signale un nouvel incident en fournissant des informations sur le matériel concerné (marque, type, numéro de série) ainsi qu'en fournissant une description de la panne recensée. L'incident est ajouté toute suite après à une liste des incidents techniques non encore traité avec l'état « non traité ».

Problème (3/5)

- Lorsque le superviseur se connecte au système, il consulte la liste des incidents, et les affecte aux différents techniciens, l'incident passe à l'état « affecté ». A son tour, lorsqu'un technicien se connecte au système, il peut consulter la liste des incidents qui lui sont affectés, en choisit un et décide de le résoudre en le marquant comme étant « en cours de traitement » en attendant de le résoudre physiquement.

Problème (4/5)

- Après avoir résolu un incident, le technicien entre les détails de son intervention et marque l'incident comme étant résolu en attente de Confirmation par l'opérateur. Un message d'information est alors généré et envoyé à l'opérateur pour l'inciter à vérifier la résolution de la panne. Après s'être assuré de la résolution de son problème, l'opérateur se connecte au système consulte la liste de ses incidents et confirme ceux parmi eux qui ont été résolu. Un incident dont la résolution a été confirmée passe alors à la liste des problèmes résolus définitivement. Si, après avoir vérifié son matériel, l'opérateur découvre que le problème existe toujours alors il se connecte au système et marque l'incident avec « problème persistant » l'incident repasse donc à la liste des incidents affectés au même technicien et qui sont en attente de traitement.

Problème (5/5)

- ❑ Le superviseur est chargé aussi de saisir tout nouveau matériel ajouté au parc (numéro de série, type, configuration,...etc). Il est chargé aussi de la gestion des comptes (login, mot de passe) pour les techniciens et les opérateurs (création, modification, activation, désactivation).
- ❑ Finalement, le superviseur, peu consulter les statistiques suivantes :
 - Nombre de pannes sur une période de temps.
 - Nombre de pannes par matériel.
 - Nombre de pannes résolus par technicien sur une période de temps.
 - Temps moyen de résolution d'un problème par technicien.

Questions

- 1) Fournir une description des acteurs (3 lignes max pour chacun)
- 2) Construire le diagramme de cas d'utilisations correspondant.
- 3) Fournir ensuite la description des scénarios d'échange relatifs à chaque cas d'utilisation.