



Génie Logiciel 1

Cas d'utilisation



Ecole d'ingénieurs et d'architectes de Fribourg
Hochschule für Technik und Architektur Freiburg

Télécom 2ème année
Semestre d'automne
2018-2019

.....
Hes-SO FRIBOURG
FREIBURG
Haute Ecole Spécialisée
de Suisse occidentale
Fachhochschule Westschweiz



Une Démarche Orientée Objet avec UML

.....
Hes-SO FRIBOURG
FREIBURG
Haute Ecole Spécialisée
de Suisse occidentale
Fachhochschule Westschweiz
University of Applied Sciences
Western Switzerland
.....

• ANALYSE

- Définitions des besoins
 - Acteurs, cas d'utilisation
 - relations entre acteurs et cas d'utilisation
 - diagrammes des cas d'utilisation
- Découvrir et documenter les objets
 - Diagrammes d'objets
 - Lien entre les objets
- Communication entre objets
 - Diagrammes de séquence et de collaboration (scénarios)

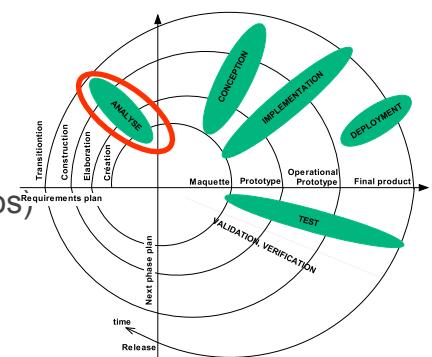


Diagramme de cas d'utilisation

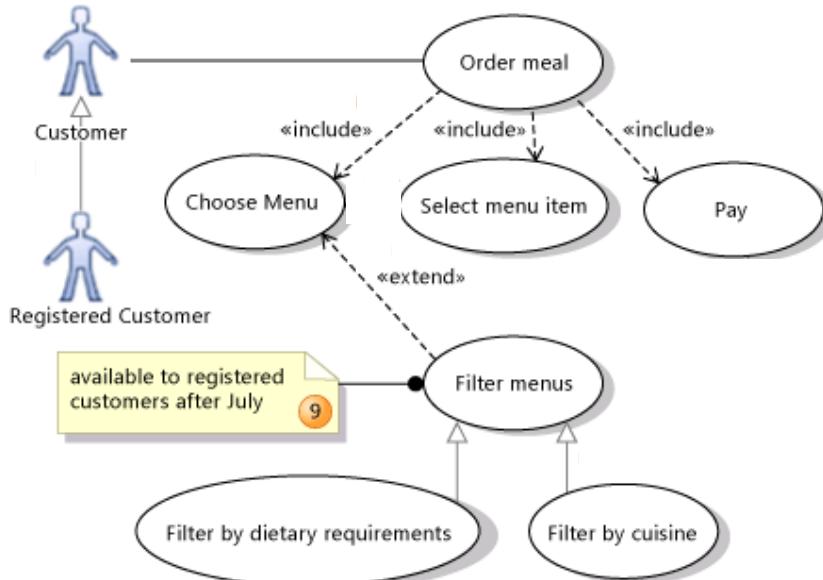


Diagramme de cas d'utilisation

• Objectifs

- Comprendre et structurer les besoins du client et les objectifs correspondants d'un système.
- Décrire des interactions du système avec son environnement
 - Expression du comportement du système (actions et réactions) selon le point de vue des utilisateurs du système
 - Détermination des besoins des utilisateurs finaux
- Les exigences d'un système sont essentiellement centrée sur les besoins de l'utilisateur.
- On se concentre ici sur les préoccupations réelles des utilisateurs
 - pas de solutions d'implémentation, pas d'inventaire fonctionnel du système.
- **Les cas d'utilisation serviront de base à la validation et à la traçabilité du système. Ils seront le fil conducteur du développement.**

Éléments de base des cas d'utilisation

• Cas d'utilisation

- Ensemble d'actions réalisées par le système
 - En relation avec un ou plusieurs acteurs.
- Ils peuvent être structurés
 - Extension, Inclusion et Généralisation
- L'ensemble des cas d'utilisation décrit les objectifs (le but) du système.

• Acteur

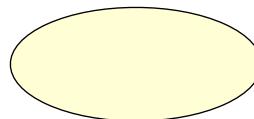
- **Entité externe** qui interagit avec le système
 - L'acteur consulte ou modifie l'état du système.
 - En réponse à l'action d'un acteur, le système fournit un service qui correspond à son besoin.
- Les acteurs peuvent être classés (généralisation / spécialisation).

Les cas d'utilisation ?

- Définit les principales façons d'utiliser le système
- Comment chaque acteur interagit-il avec le système
- Un cas d'utilisation représente une partie majeure de fonctionnalité qui est complète du début à la fin qui délivre quelque chose de valeur à l'acteur
- Une description brève décrit le pourquoi du cas d'utilisation, en fournissant une définition de haut niveau de la fonctionnalité produite

Les cas d'utilisation : Syntaxe

- **Cas d'utilisation:** est représenté par une ellipse
 - Dialogue entre l'acteur et le système
 - Représente en général un service rendu (tâche exécutée par le système) à l'acteur qui interagit avec le système
 - Décrit par un ensemble de scénarios
- **Le cas d'utilisation (use case) montre les besoins de l'utilisateur**
 - Il décrit les processus que le système réalise par rapport aux acteurs.



Les cas d'utilisation - Définition

Un cas d'utilisation (Use Case) représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier.

Il modélise un service rendu par le système et exprime des interactions acteurs/système qui apportent une valeur ajoutée notable à l'acteur concerné.

Les cas d'utilisation - Découverte

- **Comment découvrir les cas d'utilisation lors de la lecture d'un problème :**
 - Quelles sont les tâches de chaque acteur ?
 - Quels acteurs ou cas d'utilisation vont créer, stocker, modifier, supprimer ou lire de l'information ?
 - Est-ce que certains acteurs auront besoin d'informer le système ou d'être informés par le système sur des modifications soudaines, externes ?
 - Quels cas d'utilisation vont faire vivre et maintenir le système ?
 - Est-ce que tous les comportements principaux du système sont présents dans les cas d'utilisation ?

Les acteurs ?

- **Tout (personnes ou choses) ce qui interagit avec le système**
- **Qui reçoit et/ou envoie de l'information au système**
- **Une description brève doit identifier le rôle joué par l'acteur lorsqu'il interagit avec le système**

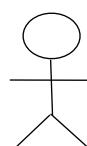
Les acteurs - Définition

Un cas d'utilisation (Use Case) représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier.

Il modélise un service rendu par le système et exprime des interactions acteurs/système qui apportent une valeur ajoutée notable à l'acteur concerné.

Les acteurs - Syntaxe

- **Acteur:** est représenté par une icône dans le diagramme
 - Représente le rôle joué par l'élément externe au système.
 - Un élément externe peut jouer plusieurs rôles : plusieurs acteurs
 - Plusieurs éléments externes peuvent jouer le même rôle : un seul acteur
- **Un acteur représente un rôle joué par une personne ou une chose qui interagit avec le système**
 - La même personne physique peut jouer le rôle de plusieurs acteurs
 - Un même rôle peut être joué par différentes personnes physiques



Les acteurs - Découverte

- **Comment découvrir les acteurs lors de la lecture d'un problème :**

- Qui a des besoins ?
- Où le système est-il utilisé dans l'organisation ?
- Qui va bénéficier de l'utilisation du système ?
- Qui va approvisionner le système avec l'information, qui va l'utiliser, et la supprimer ?
- Qui va maintenir et faire vivre le système ?
- Est-ce que le système utilise une ressource externe ?
- Est-ce qu'une personne joue plusieurs rôles différents ?
- Est-ce que plusieurs personnes jouent le même rôle ?

Exercice A - Guichet Médiathèque

- **Buts et besoins (point de vue)**

- Il s'agira d'un système de guichet automatique où l'utilisateur pourra s'inscrire et se désinscrire à la médiathèque. Lorsqu'il sera inscrit, il pourra choisir dans un menu informatisé le document qu'il désire emprunter et prendre le document. Pour le retour de documents, l'utilisateur dépose simplement le document dans le guichet.
- *Pour l'instant il est possible d'emprunter les documents suivants: des livres, des CD-Audio, des cassette audio, des jeux sur CD*

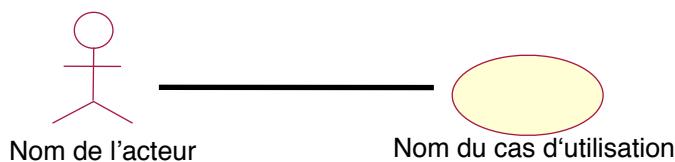
- **Exercices**

- Trouver les acteurs de la médiathèque et décrire leurs interactions possibles avec le système
- Identifier les cas d'utilisation de la médiathèque et les associer aux acteurs concernés

Relation entre acteurs et cas d'utilisation

- La participation de l'acteur au cas d'utilisation est signalé par un trait entre l'acteur et le cas d'utilisation

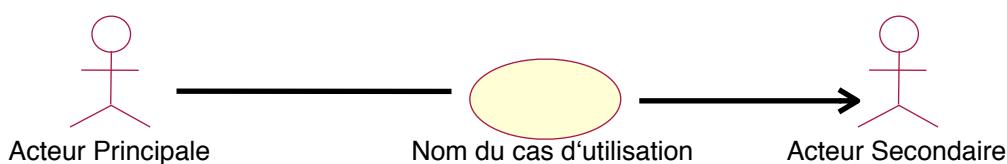
- Représente la communication entre l'acteur et le cas d'utilisation



Relation entre acteurs et cas d'utilisation

- Depuis UML 2.5, on fait une distinction entre un acteur primaire et un acteur secondaire.

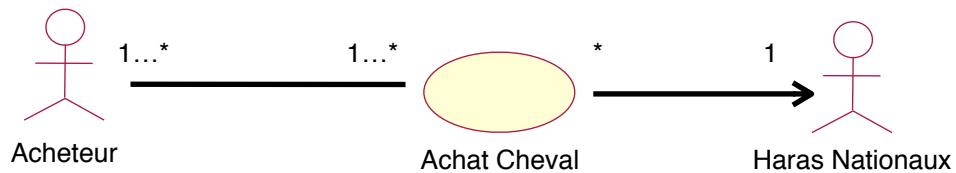
- Un acteur secondaire est représenté comme un acteur primaire. A la différence de l'association entre un acteur primaire et un cas d'utilisation, l'association entre un acteur secondaire et un cas d'utilisation possède obligatoirement un sens qui va du cas d'utilisation vers l'acteur.
 - **Acteurs principaux** : Ceux qui vont réaliser le cas d'utilisation (la relation avec le cas d'utilisation est illustrée par le trait liant le cas d'utilisation et l'acteur dans un diagramme de cas d'utilisation)
 - **Acteurs secondaires** : Ceux qui ne font que recevoir des informations à l'issue de la réalisation du cas d'utilisation



Relation entre acteurs et cas d'utilisation

- Depuis UML 2.5, il est également possible de raffiner la relation entre acteur et cas d'utilisation grâce à des cardinalités

- Les cardinalités figurent niveau de chaque extrémité de l'association.
 - La cardinalité située à l'extrémité du cas d'utilisation indique avec combien d'instances du cas d'utilisation, chaque instance de l'acteur situé à l'aire extrémité est limitée.
 - La cardinalité située à l'extrémité de l'acteur indique avec combien d'instances de l'acteur, chaque instance du cas d'utilisation situé à l'autre extrémité est liée

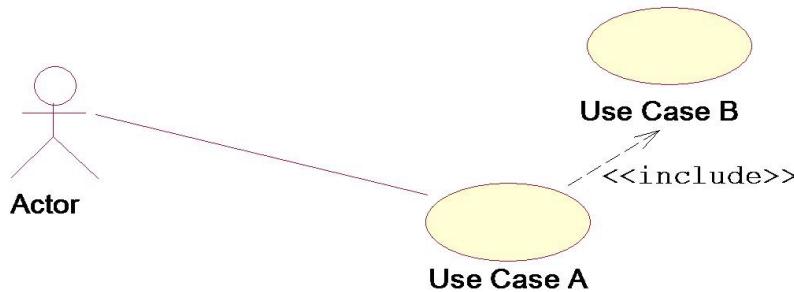


Relation entre cas d'utilisation

- UML offre la possibilité de « structurer » les cas d'utilisation
 - Exprimer des « relations » entre cas d'utilisation
- Il y a essentiellement deux raisons pour structurer les cas d'utilisation
 - On (le client) tient à exprimer une dépendance entre cas d'utilisation
 - Ca fait partie du comportement désiré du système
 - Cela permet de simplifier le diagramme
 - En factorisant certains cas d'utilisation on obtient un diagramme plus simple qui exprime la même chose

Inclusion (Include)

La relation <<include>> existe quand un comportement ou un fragment de comportement est similaire dans plusieurs cas d'utilisations



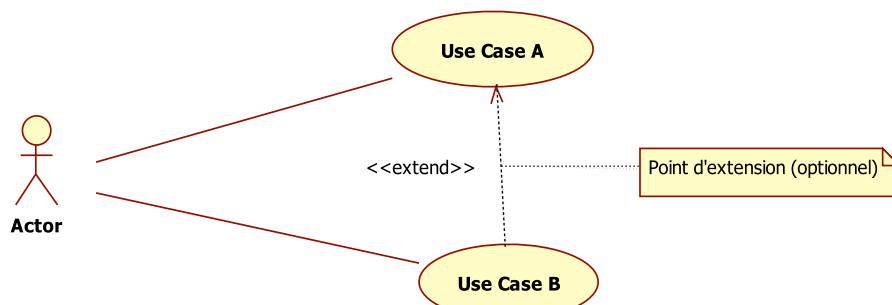
Le Use Case A inclut le Use Case B

Attention : Ne pas tomber dans le travers consistant à faire du fonctionnel

Je découpe mes comportements en petits comportements et j'utilise la relation d'inclusion comme moyen d'exprimer un décomposition fonctionnelle

Extension (Extend)

- L'extension permet d'ajouter un comportement (optionnel) à un cas d'utilisation.
 - Ces extensions correspondent à des activées optionnelles
 - Ils peuvent être activé par des « **points d'extensions** » (ou points de variance)
 - Les point d'extensions peuvent être indiqués sur la relation d'extension (flèche) qui relie les cas d'utilisation.



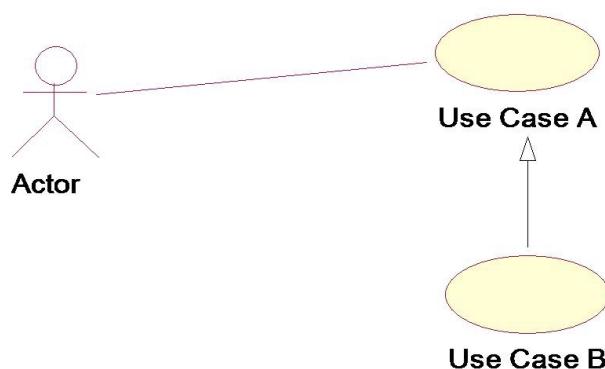
Le Use Case B étend le Use Case A

Exercice B - Guichet Médiathèque

- En se basant sur l'exercice A, on précise que
 - Lors de chaque utilisation de la médiathèque l'utilisateur doit se loguer.
 - lors de l'utilisation (emprunt) ou de l'inscription, l'utilisateur peut, sur demande, consulter une aide en ligne.
- Exercice
 - On demande de compléter le diagramme de cas d'utilisation de la médiathèque pour tenir compte de ces ajouts

Généralisation (Spécification)

La généralisation me permet de factoriser un comportement et d'en dériver des variantes

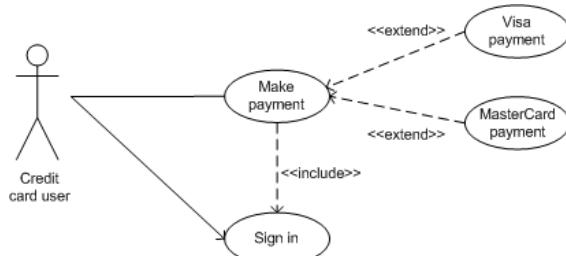


Le Use Case B est un cas particulier du Use Case A

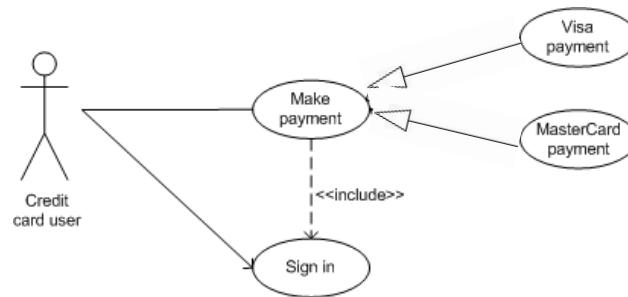
Inclusion vs Généralisation

Il peut-être parfois difficile de décider s'il faut utiliser la généralisation ou l'extension.

Extend...



... ou Généralisation ?



Exercice C - Guichet Médiathèque

- Exercice**

- En se basant sur l'exercice B, proposez un diagramme de cas d'utilisation pour la médiathèque qui utilise la généralisation

Structurer les cas d'utilisation

- UML définit trois types de relations entre cas d'utilisation
 - L'inclusion (<<include>>)
 - Cette relation sert à exprimer qu'un cas d'utilisation est aussi une partie d'un autre cas. Autrement dit qu'un fragment de comportement est similaire dans plusieurs (au moins 2) cas d'utilisation.
 - La généralisation (→)
 - La généralisation est utilisée lorsqu'un cas d'utilisation est semblable à un autre mais que ce dernier comporte une fonctionnalité supplémentaire. Afin d'éviter de devoir décrire deux fois la partie commune, le deuxième cas fait référence au premier (il spécialise le premier).
 - L'extension (<<extend>>)
 - Un cas d'utilisation "étend" un autre cas lorsqu'il constitue un comportement possible (mais pas obligatoire) d'un autre cas. L'extension ressemble à la spécialisation/généralisation mais comporte plus de restrictions

Structurer les cas d'utilisation

- UML définit trois types de relations entre cas d'utilisation
 - L'inclusion (<<include>>)
 - Cette relation sert à exprimer qu'un cas d'utilisation est aussi une partie d'un autre cas. Autrement dit qu'un fragment de comportement est similaire dans plusieurs (au moins 2) cas d'utilisation.
 - La généralisation (→)
 - La généralisation est utilisée lorsqu'un cas d'utilisation est semblable à un autre mais que ce dernier comporte une fonctionnalité supplémentaire. Afin d'éviter de devoir décrire deux fois la partie commune, le deuxième cas fait référence au premier (il spécialise le premier).
 - L'extension (<<extend>>)
 - Un cas d'utilisation "étend" un autre cas lorsqu'il constitue un comportement possible (mais pas obligatoire) d'un autre cas. L'extension ressemble à la spécialisation/généralisation mais comporte plus de restrictions

Diagramme de cas d'utilisation

- Resumé

- Comprendre et structurer les besoins du client (et donc les objectifs du système)



- Décrire des interactions du système avec son environnement



- Décrire de manière détaillée le système

