

## Chapitre 4

# Fonctionnalité et cas d'utilisation

## Sommaire

### Identification et représentation des cas d'utilisation.

- Les cas d'utilisation dans le processus de développement
- Définition et formats
- Identification et priorité
- Diagrammes
- Relations

# Vue utilisateur

### Vue Logique

- Diagramme de classes
- Diagramme d'objets
- Diagramme de paquetage

### Vue d'implantation

- Diagramme de composants

### Vue utilisateur

- Diagramme de cas d'utilisation*

### Vue comportementale

- Diagramme d'états
- Diagramme d'activités
- Diagramme de séquence
- Diagramme de collaboration

### Vue de déploiement

- Diagramme de déploiement

# Les cas d'utilisation et le processus de développement

1. Après avoir **identifié les requis** du système, **définir les frontières** du système, puis **identifier les acteurs** et les **cas d'utilisation**.
2. Écrire tous les cas d'utilisation en **format de haut niveau** et les **catégoriser** comme primaires, secondaires ou optionnels.
3. Tracer un **diagramme de contexte** (cas d'utilisation primaires).
4. Identifier et **illustrer les relations** entre les cas.
5. Écrire les cas d'utilisation les plus **critiques, influents ou risqués** dans un format **étendu et essentiel**.
6. Écrire le minimum de cas réels.
7. **Ordonner** les cas d'utilisation en ordre d'importance.

## Cas d'utilisation

### Objectifs des cas d'utilisation:

- Capturer le **comportement désiré** du système
- Spécifier ce que le système fait (fonctions), mais pas comment il le fait.
- Servir d'**entente** entre les développeurs, les utilisateurs et les experts sur les **fonctions disponibles** et sur la **façon d'interagir** avec le système.
- Servir à valider l'**architecture logicielle** qui sera développée.

## Cas d'utilisation

Définition:

Un cas d'utilisation est un **document narratif** qui décrit la **séquence d'évènements** dans laquelle un acteur (un agent externe) utilise un système pour accomplir un **processus**.

Ce sont des descriptions de l'utilisation d'un système.

Les cas d'utilisation sont une forme de requis ou de spécifications fonctionnelles ou comportementales du système

- Les cas d'utilisation illustrent et impliquent les requis.

## Cas d'utilisation

Les cas d'utilisation peuvent être spécifiés avec différents **niveaux de détail**:

- Format de **haut niveau**:
  - Abrégé, peu détaillé
  - Décrit le scénario principal seulement
- Format **informel**:
  - Court
  - Brève description des scénarios alternatifs
- Format **étendu**:
  - Très détaillé
  - Description complète du scénario principal et des scénarios alternatifs

# Cas d'utilisation de haut niveau

Un exemple de cas d'utilisation pour le guichet automatique:

Cas d'utilisation:	<b>CU1.0 - Effectuer un retrait</b>
Acteur(s):	Client
Type:	Primaire
Description:	Un client se présente au guichet automatique, s'identifie, retire un montant d'argent et quitte avec son argent. Le solde du compte du client a été ajusté.



## Cas d'utilisation étendu

- En format étendu, un cas d'utilisation est constitué de plusieurs éléments:
  - Nom du cas d'utilisation
  - Système
  - Acteurs
  - Parties prenantes
  - Préconditions
  - Garanties en cas de succès (postconditions)
  - Une description du scénario principal
  - Une description des scénarios alternatifs

# Cas d'utilisation étendu

Cas d'utilisation:	<b>CU1.0 - Effectuer un retrait</b>
Système:	Guichet automatique
Niveau	« objectif usager » ou « sous-fonction »
Acteur primaire:	Client
Parties prenantes et intérêts:	Client: Il désire repartir avec son argent. Gestionnaire de la banque: Il veut que le système enregistre la transaction.
Préconditions:	Le client a un compte bancaire actif.
Garanties en cas de succès:	Le client a reçu son argent et son compte a été débité.
Scénario principal:	<ol style="list-style-type: none"><li>1. Un client se présente au guichet automatique</li><li>2. Il s'identifie.</li><li>3. ...</li></ol>
Scénarios alternatifs:	...

## Cas d'utilisation étendu – Scénario en colonnes

1. Un client se présente au guichet
2. Le client s'identifie
3. Présentation des options
4. Le client choisit de faire un retrait.
5. Sollicitation du compte à débiter
6. Le client choisit le compte
7. Sollicitation du montant du retrait.
8. Le client entre le montant à retirer.
9. Validation de la disponibilité des fonds.
10. Débite le compte
11. Remet l'argent
12. Le client prend l'argent et quitte.

## Cas d'utilisation étendu – Déroulement alternatif

Ligne 2 :	Identification invalide. Indiquer une erreur.
Ligne 8 :	Le montant entré n'est pas un multiple de 20\$. Indiquer une erreur.
Ligne 9 :	Insuffisance de fonds. Indiquer une erreur.

# Cas d'utilisation : quelques définitions

**Cas d'utilisation (usecase):** Décrit un ensemble de séquences dans lequel chacune des séquences représente l'interaction des *acteurs* avec le système lui-même. Représente les fonctions du système visibles par les acteurs.

**Acteur:** Ensemble de rôles interprétés par les utilisateurs du système lorsque ceux-ci interagissent avec le système à travers les cas d'utilisation. Il s'agit d'une entité externe au système qui interagit avec celui-ci.

**Parties prenantes et intérêts:** Liste des personnes qui sont impliqués dans le cas d'utilisation, ce qu'elles souhaitent et de quelle façon elles sont affectées par la réalisation du cas.

**Préconditions:** Conditions qui doivent être vraies avant le début du scénario

**Garanties en cas de succès (postconditions):** Conditions qui doivent être vraies lorsque le cas d'utilisation se termine avec succès.

## Identification des cas d'utilisation

- Méthode basée sur les **acteurs**:
  1. Identifier les acteurs liés au système
  2. Identifier, les processus initiés par chacun des acteurs.
- Méthode basée sur les **évènements**
  1. Identifier les évènements externes auxquels un système doit répondre.
  2. Relier ces évènements aux acteurs et aux cas d'utilisation.

## Cas d'utilisation : erreur commune

Erreur:

- Représenter des étapes individuelles, des opérations de base ou des transactions comme des cas d'utilisation.

Un cas d'utilisation est:

- une **description relativement longue**
- **allant du début à la fin d'un processus**
- comportant **plusieurs étapes ou transactions**.

➤ Une simple étape dans un processus N'est PAS un cas d'utilisation.

## Cas d'utilisation : tests de pertinence

Trois tests pour décider de la pertinence d'un cas d'utilisation:

- Le test du patron
- Le test du processus d'affaire élémentaire
- Le test de la dimension



### Cas d'utilisation : tests de pertinence

**Le test du patron:** mon patron sera-t-il satisfait de moi si je passe ma journée à effectuer ce cas d'utilisation ?

**Le test du processus d'affaire élémentaire:** un processus d'affaire élémentaire est une tâche accomplie par une personne à un endroit et un moment donnés, en réponse à un événement d'affaire, qui ajoute une quantité mesurable de valeur et laisse les données dans un état consistant.

**Le test de la dimension :** un cas d'utilisation est extrêmement rarement constitué d'une seule action ou étape. Un cas d'utilisation comprend typiquement plusieurs étapes, et dans un format étendu nécessite souvent de 3 à 10 pages de texte afin de le décrire.

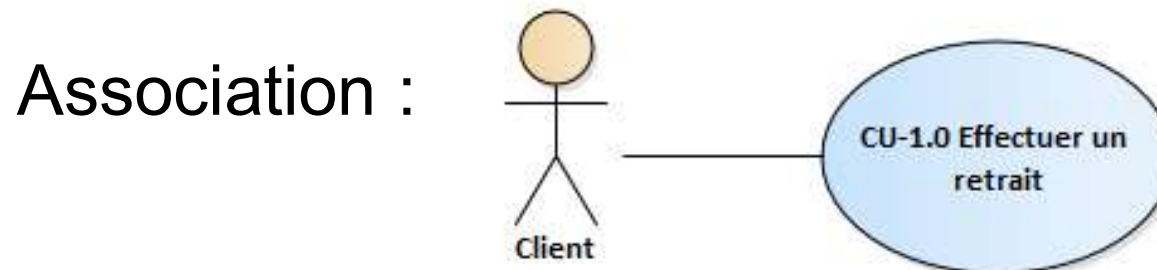
### Quelques règles d'écriture

- Écrire dans un style qui se concentre sur l'essentiel de **la fonctionnalité et des objectifs** et néglige les aspects superflus, comme les aspects d'interface-usager
- Écrire dans un style dépouillé
- Écrire des cas d'utilisation de type « boîte noire »
- Adopter le point-de-vue de l'acteur et de ses objectifs

# Les cas d'utilisation et le processus de développement

1. Après avoir listé les fonctions du système, définir les frontières du système, puis identifier les acteurs et les cas d'utilisation.
2. Écrire tous les cas d'utilisation en format de haut niveau et les catégoriser comme primaires, secondaires ou optionnels.
3. *Tracer un diagramme de contexte (cas d'utilisation primaires).*
4. Identifier et illustrer les relations entre les cas.
5. Écrire les cas d'utilisation les plus critiques, influents ou risqués dans un format étendu et essentiel.
6. Écrire le minimum de cas réels.
7. Ordonner les cas d'utilisation en ordre d'importance.

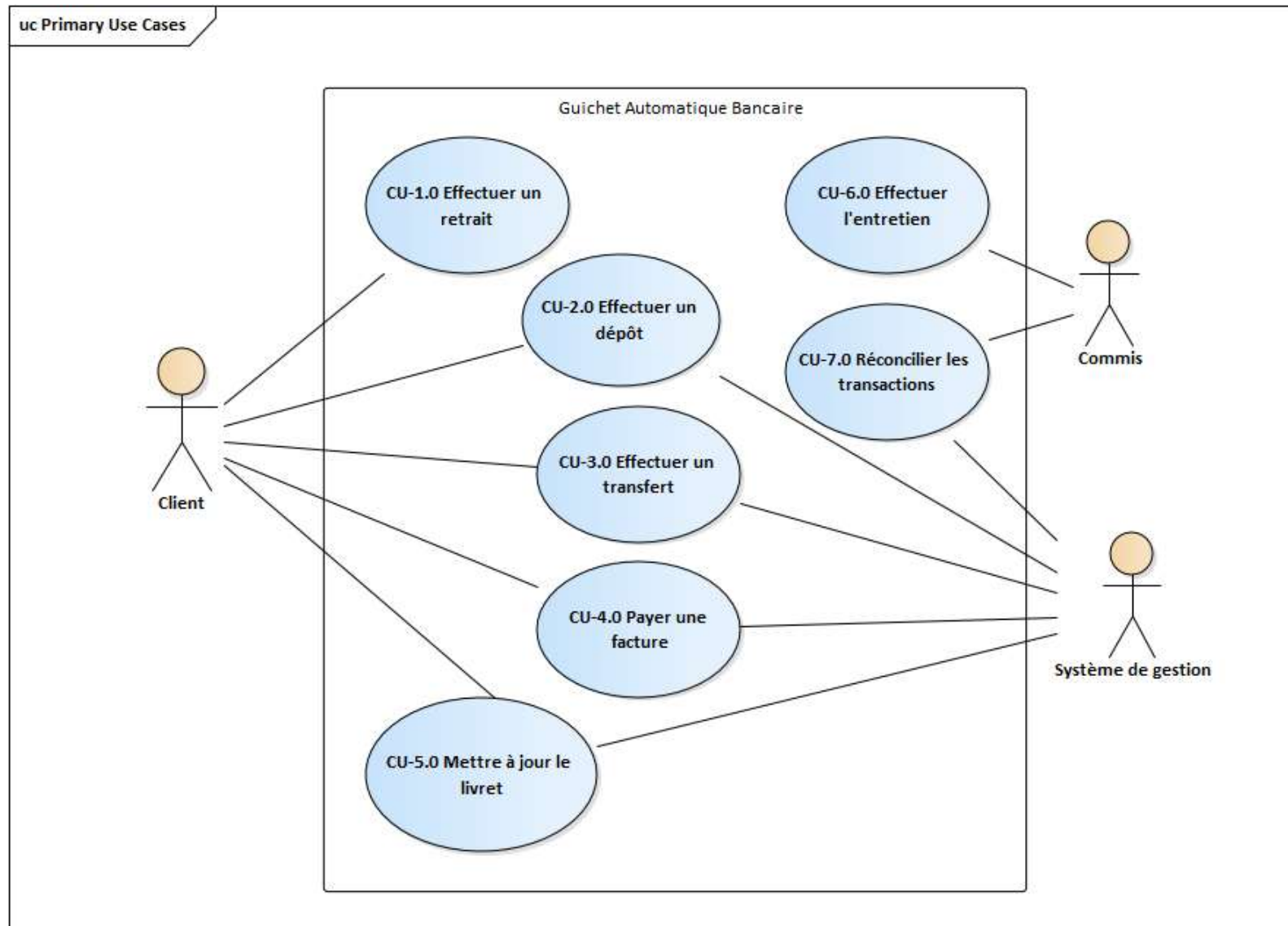
## Cas d'utilisation: représentation graphique



## Diagramme de contexte

- Le **diagramme de contexte** permet de visualiser les cas d'utilisation **primaires** d'un système.
- Ceci permet de voir rapidement les **principales fonctions** d'un système.
- Plus particulièrement, le diagramme de contexte définit:
  - Les **limites** du système modélisé.
  - Les principaux **acteurs** du modèle.
  - Les cas d'utilisation **primaires**.

## Diagramme de contexte : le guichet automatique



# Les cas d'utilisation et le processus de développement

1. Après avoir listé les fonctions du système, définir les frontières du système, puis identifier les acteurs et les cas d'utilisation.
2. Écrire tous les cas d'utilisation en format de haut niveau et les catégoriser comme primaires, secondaires ou optionnels.
3. Tracer un diagramme de contexte (cas d'utilisation primaires).
4. *Identifier et illustrer les relations entre les cas.*
5. Écrire les cas d'utilisation les plus critiques, influents ou risqués dans un format étendu et essentiel.
6. Écrire le minimum de cas réels.
7. Ordonner les cas d'utilisation en ordre d'importance.

## Relations entre les cas d'utilisation

On ajoute des relations entre les cas d'utilisation pour améliorer la compréhension et réduire les duplications

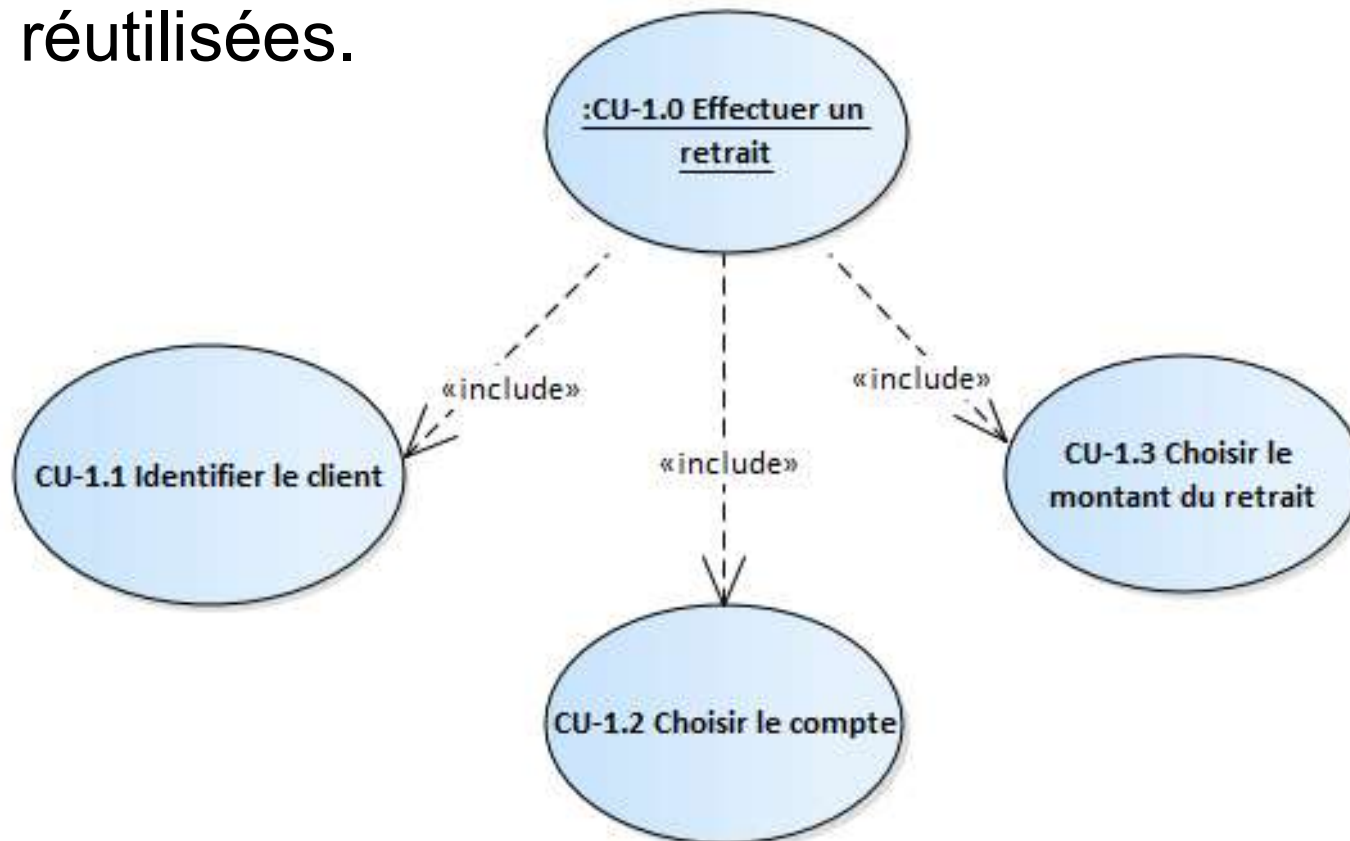
- Relations d'utilisation
  - association stéréotypée « *include* ».
- Relations d'extension
  - association stéréotypée « *extend* ».
- Relation de généralisation.



## Relation d'utilisation (« include »)

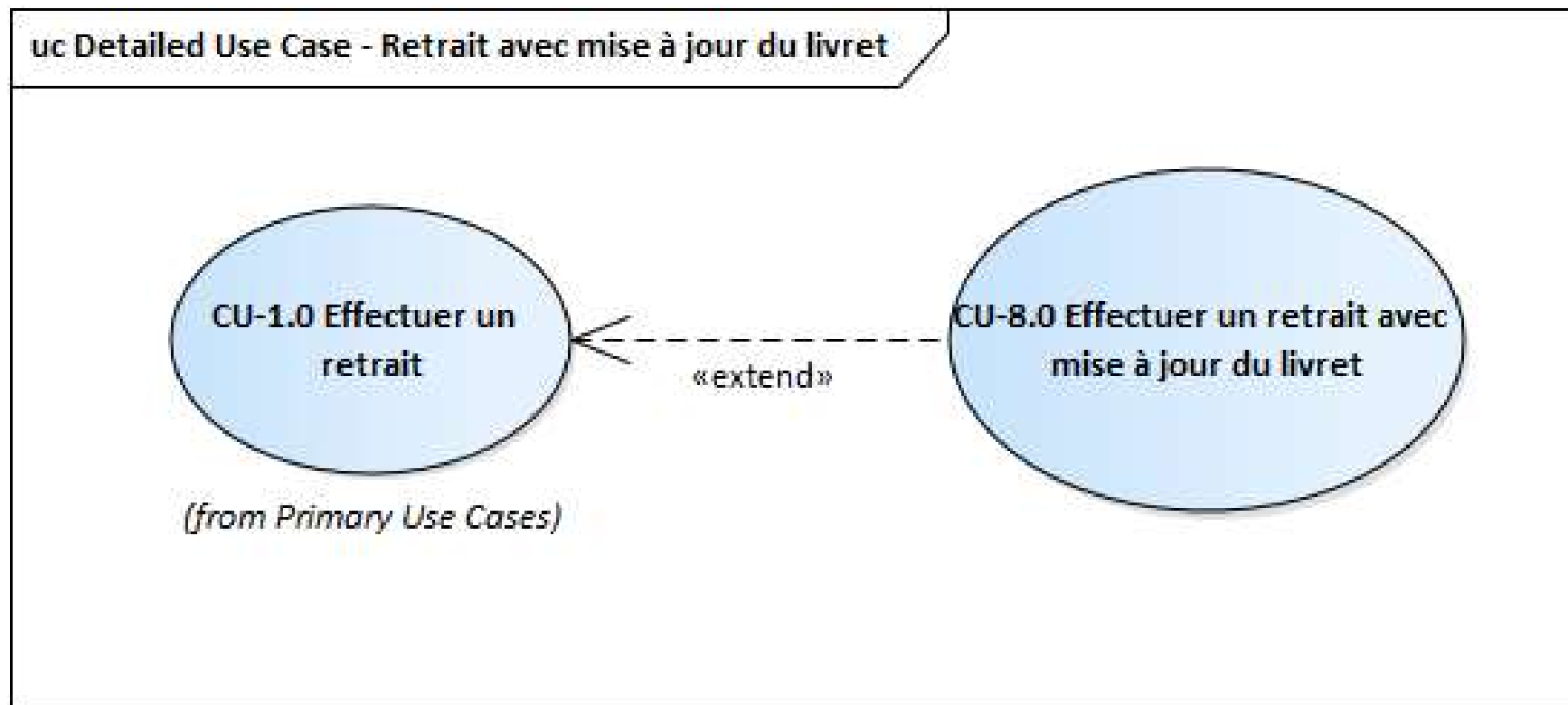
La relation la plus commune et la plus utile.

- La relation *include* permet de modéliser des ‘sous-fonctions’ ou des séquences d’étapes qui peuvent être réutilisées.



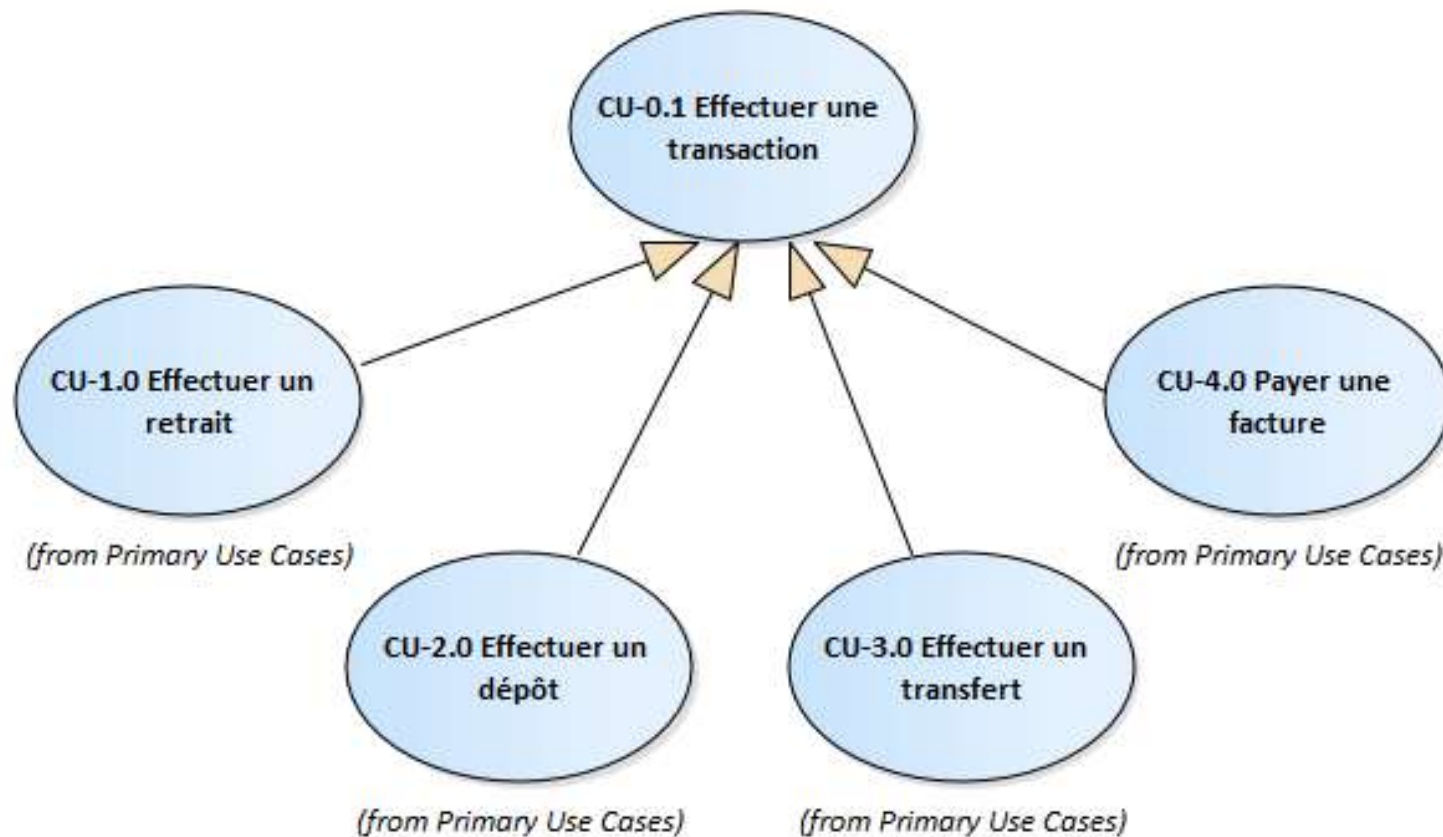
### Relation d'extension (« extend »)

La relation d'extension permet d'ajouter des étapes à un cas existant sans modifier le cas original.



## Relations de généralisation

La relation de généralisation permet de hiérarchiser les cas des plus abstraits vers les plus concrets.



# Les cas d'utilisation et le processus de développement

1. Après avoir listé les fonctions du système, définir les frontières du système, puis identifier les acteurs et les cas d'utilisation.
2. Écrire tous les cas d'utilisation en format de haut niveau et les catégoriser comme primaires, secondaires ou optionnels.
3. Tracer un diagramme de contexte (cas d'utilisation primaires).
4. Identifier et illustrer les relations entre les cas.
5. Écrire les cas d'utilisation les plus critiques, influents ou risqués dans un format étendu et essentiel.
6. Écrire le minimum de cas réels.
7. *Ordonner les cas d'utilisation en ordre d'importance.*

# Trier les cas d'utilisation

Les cas d'utilisation devraient être triés en **ordre d'importance** de façon à ce que les cas les plus importants soient **traités le plus tôt** possible.

On considère importants les cas d'utilisation qui:

1. Ont un **impact important sur l'architecture du système**, par exemple parce qu'ils introduisent beaucoup de concepts ou classes, ou exigent de la persistance.
2. Contiennent beaucoup **d'information sur la conception** sans nécessairement nécessiter beaucoup d'effort.
3. Contiennent des fonctions **risquées, complexes ou urgentes**.
4. Impliquent des **efforts** importants de **recherche** ou l'utilisation de **nouvelles technologies**.
5. Représentent **l'activité commerciale principale**.
6. Impliquent directement des **revenus supplémentaires** ou des **réductions de coûts**.

## Décomposition de cas d'utilisation

- Dans un développement itératif, on se donne un **échéancier** serré pour chaque **cycle de développement**.
- *Il peut donc arriver qu'un cas d'utilisation ne soit pas complètement réalisé au cours d'un cycle.*
- *Dans ces cas, on peut en faire des **versions simplifiées** réalisables dans un cycle.*
- Les versions simplifiées sont **raffinées** de cycle en cycle jusqu'à correspondre au cas d'utilisation initial.

# Rédaction des cas d'utilisation selon les phases du développement

Les cas d'utilisation sont rédigés de façon incrémentale au fur et à mesure que le besoin se fait sentir

**Phase de planification:** les cas d'utilisation sont rédigés en format de haut-niveau. Les 10% à 20% des cas les plus importants sont écrits en format étendu.

**Phase d'élaboration:** durant cette phase, la grande majorité des requis sont identifiés et clarifiés. Chaque itération focalise sur un sous-ensemble des cas, selon leur priorité.

**Phase de construction:** durant cette phase, le système est complété. Des ajustements mineurs aux cas d'utilisation peuvent être apportés.

## Élaboration des tests basés sur les cas d'utilisation

Les cas d'utilisation fournissent le canevas sur lequel se base l'élaboration des tests du système.

La réalisation de chaque cas d'utilisation doit fournir une plus-value quantifiable à l'utilisateur.

L'obtention de cette plus-value est la mesure du succès du cas d'utilisation.