

UML : diagramme de cas d'utilisation

Achref El Mouelhi

Docteur de l'université d'Aix-Marseille
Chercheur en programmation par contrainte (IA)
Ingénieur en génie logiciel

`elmouelhi.achref@gmail.com`



**UNIFIED
MODELING
LANGUAGE**

Plan

- 1 Introduction
- 2 Diagramme de contexte
- 3 Diagramme de packages
- 4 Diagramme de cas d'utilisation
- 5 Comment construire un diagramme de cas d'utilisation ?
- 6 Exercices d'application

Analyse et expression des besoins ?

- Un texte : long en taille (nombre de lignes, pages...), lent en temps (écriture ou lecture et compréhension)
- Un graphe (diagramme) : facile à comprendre, plus rapide à réaliser

© Achref EL MOUL

UML

Analyse et expression des besoins ?

- Un texte : long en taille (nombre de lignes, pages...), lent en temps (écriture ou lecture et compréhension)
- Un graphe (diagramme) : facile à comprendre, plus rapide à réaliser

En **UML 2**, cela peut passer par plusieurs étapes

- Un diagramme de contexte
- Un diagramme de package
- Un diagramme de cas d'utilisation général
- Un diagramme de cas d'utilisation détaillé

Diagramme de contexte

- Premier diagramme à faire pendant la phase d'analyse
- Ne faisant pas partie des 14 diagrammes **UML**
- Permettant de définir
 - le **système** à modéliser
 - les différents **acteurs** qui vont interagir avec le système

UML

Système ?

- futur logiciel (ou application)
- ensemble de services ou de fonctionnalités

© Achref EL MOUËL

UML

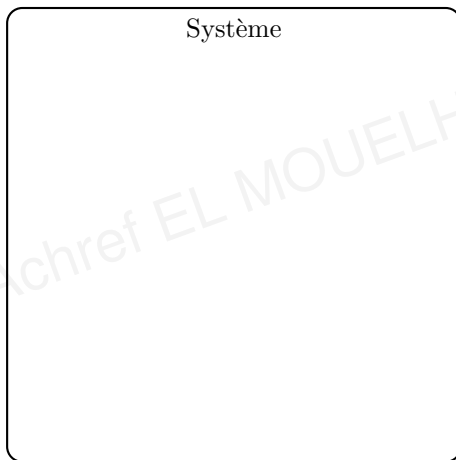
Système ?

- futur logiciel (ou application)
- ensemble de services ou de fonctionnalités

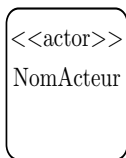
Acteur ?

- les utilisateurs de système
- ce n'est ni les développeurs ni les concepteurs ni les testeurs de système
- ce n'est forcément une personne physique

Comment représenter un système ?



Deux représentations possibles d'un acteur en UML



En utilisant un classeur



En utilisant un bonhomme

Remarque

Pas besoin d'un stéréotype `<<actor>>` dans la première représentation

Exemple : considérons un système d'achat et vente en ligne

Les différents acteurs possibles sont

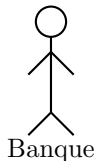
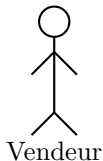
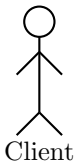
- Client
- Vendeur
- Administrateur
- Banque

UML

Exemple : considérons un système d'achat et vente en ligne

Les différents acteurs possibles sont

- Client
- Vendeur
- Administrateur
- Banque



Deux types d'acteurs en UML

- Principal : un acteur agissant directement sur le système
- Secondaire : un acteur sollicité par le système

© Achref EL MOU

Deux types d'acteurs en UML

- Principal : un acteur agissant directement sur le système
- Secondaire : un acteur sollicité par le système

Revenant à notre exemple

- Tous les acteurs sauf la banque sont des acteurs principaux
- La banque est un acteur secondaire (sollicitée par notre système pour valider le paiement)

Représentation d'acteurs secondaires en UML

On utilise le stéréotype `<<secondary>>` sur les fonctionnalités liées aux acteurs secondaires

UML

Le diagramme de contexte de l'exemple précédent en précisant les acteurs secondaires

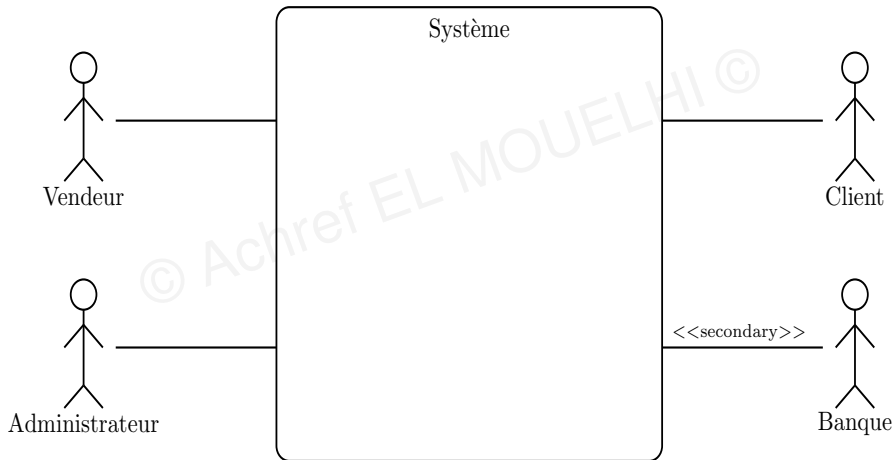
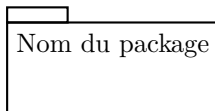


Diagramme de packages

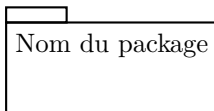
- Diagramme de structure d'**UML** (statique)
- Permettant de rassembler un ensemble d'éléments (classes, fonctionnalités, fichiers...) sémantiquement proche
- Pouvant être vu comme un ensemble de dossier (physique ou logique) + les relations de dépendance entre eux

Comment visualiser un package en UML ?

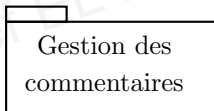


© Achref EL MOUELHI ©

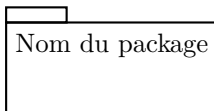
Comment visualiser un package en UML ?



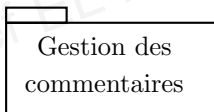
Exemple



Comment visualiser un package en UML ?

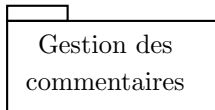
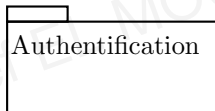
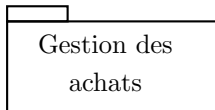


Exemple

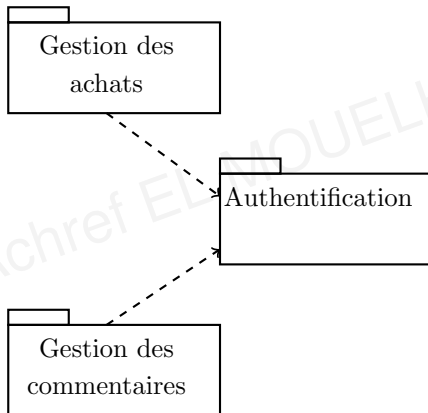


En **UML**, un package = un répertoire (un dossier) sous **Windows**

Si on décompose notre exemple en packages



On peut définir des relations de dépendances entre les packages

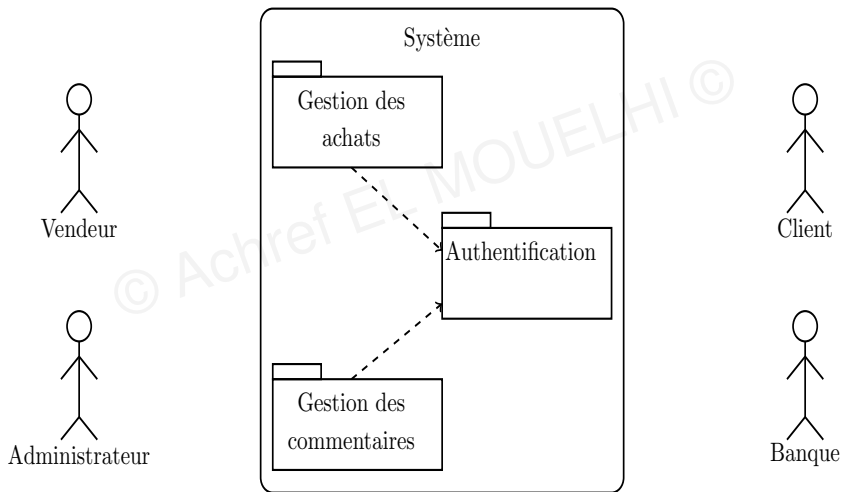


Pour obtenir le diagramme de packages, il faut

- remplacer les packages dans la partie Système du diagramme de contexte
- connecter les acteurs et les packages selon les droits

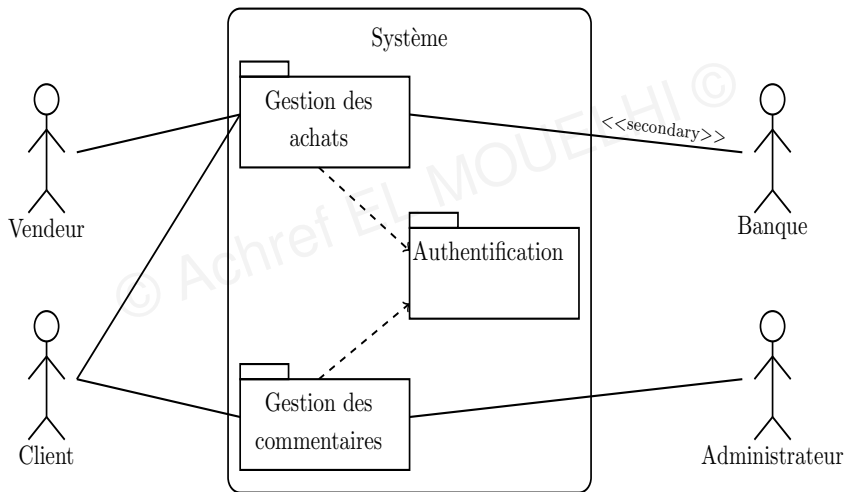
UML

Étape 1 : remplacer les packages dans la partie Système du diagramme de contexte



UML

Étape 2 : connecter les acteurs et les packages aux selon les droits



Le diagramme cas d'utilisation

- un diagramme dynamique d'**UML**
- parmi les diagrammes les plus importants
- permettant de détailler le diagramme de package en remplaçant les packages par une liste de fonctionnalités (appelées cas d'utilisation)

© Achref EL

Le diagramme cas d'utilisation

- un diagramme dynamique d'**UML**
- parmi les diagrammes les plus importants
- permettant de détailler le diagramme de package en remplaçant les packages par une liste de fonctionnalités (appelées cas d'utilisation)

Dans certains cas, on peut créer

- un diagramme de cas d'utilisation général (**Jacobson** conseille de ne pas dépasser 25 cas d'utilisation par diagramme)
- un diagramme de cas d'utilisation détaillé pour (chaque) package

Mots-clés associés

- acteur
- cas d'utilisation
- dépendance
- héritage

UML

Cas d'utilisation

fonctionnalité informatique fournie par le système

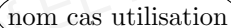
© Achref EL MOUELHI ©

UML

Cas d'utilisation

fonctionnalité informatique fournie par le système

Comment visualiser un cas d'utilisation en UML ?



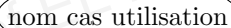
nom cas utilisation

UML

Cas d'utilisation

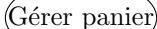
fonctionnalité informatique fournie par le système

Comment visualiser un cas d'utilisation en UML ?



nom cas utilisation

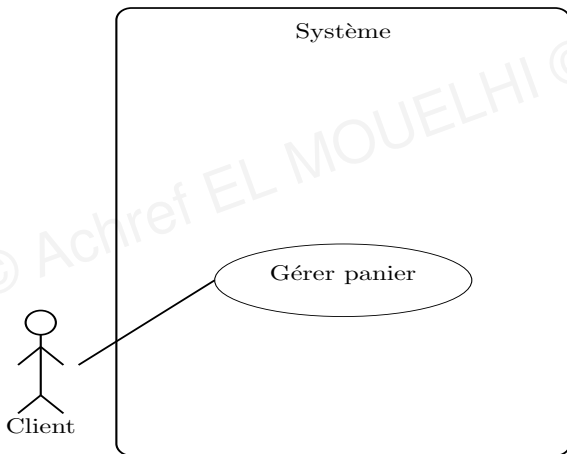
Exemple



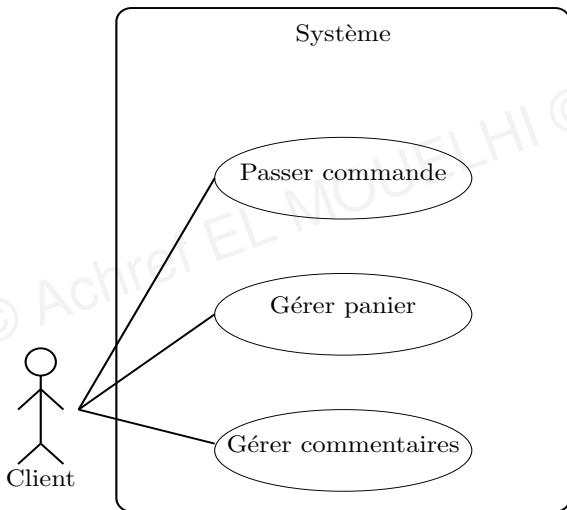
Gérer panier

Diagramme de cas d'utilisation

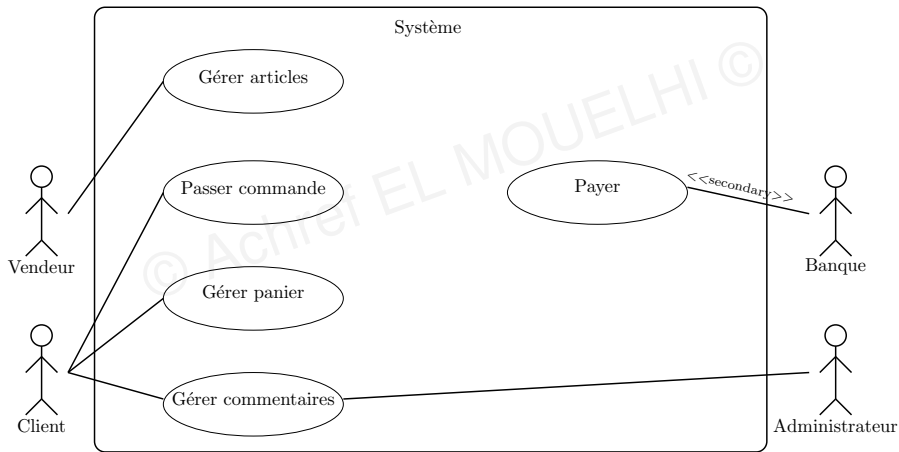
= contexte + acteurs + cas d'utilisation + associations



UML

Un acteur peut être en relation avec plusieurs cas d'utilisation

Ajoutons les cas d'utilisation de tous les acteurs de notre exemple précédent



Relations entre cas d'utilisation

- dépendance
 - inclusion
 - extension
- héritage

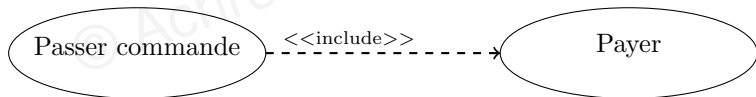
La relation `include`

Si le cas **B** inclut le cas **A**, alors **A** est une partie obligatoire de **B**.

© Achref EL MOUADJID

La relation `include`

Si le cas **B** inclut le cas **A**, alors **A** est une partie obligatoire de **B**.



La relation `extend`

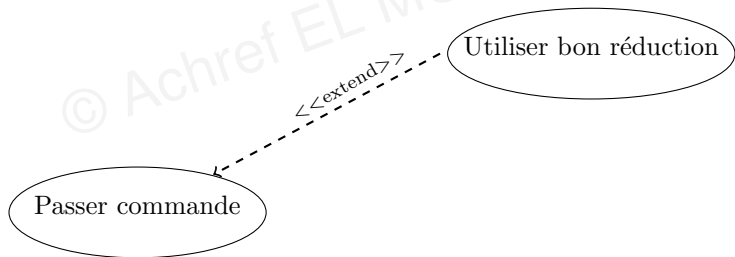
Si le cas **B** étend le cas **A**, alors **B** est une partie optionnelle de **A**.

© Achref EL MOUELHI

UML

La relation `extend`

Si le cas **B** étend le cas **A**, alors **B** est une partie optionnelle de **A**.



UML

La relation d'héritage

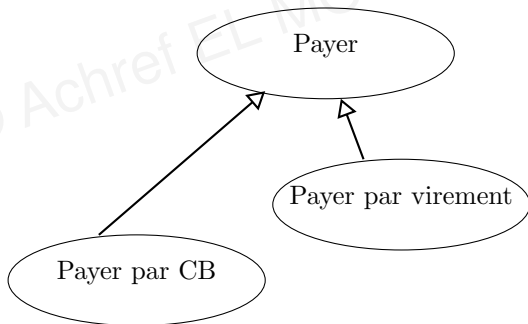
Un cas **A** est une généralisation d'un cas **B** si **B** est un cas particulier de **A**.

© Achref EL MOUELHI

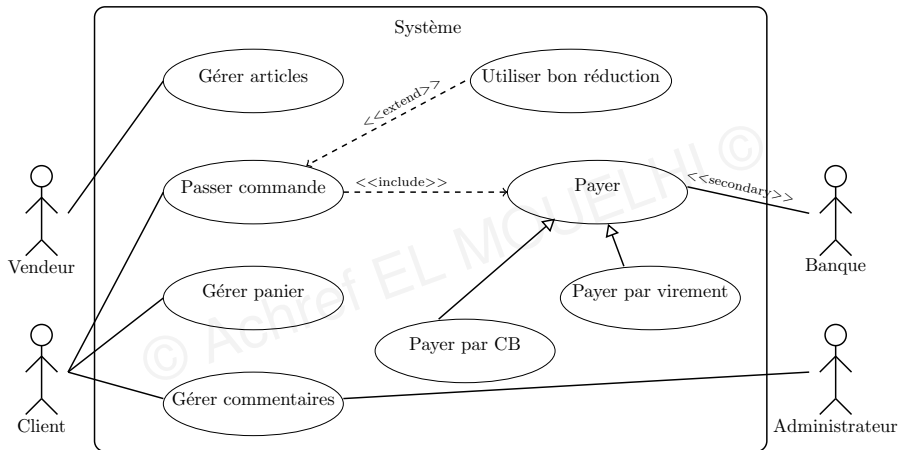
UML

La relation d'héritage

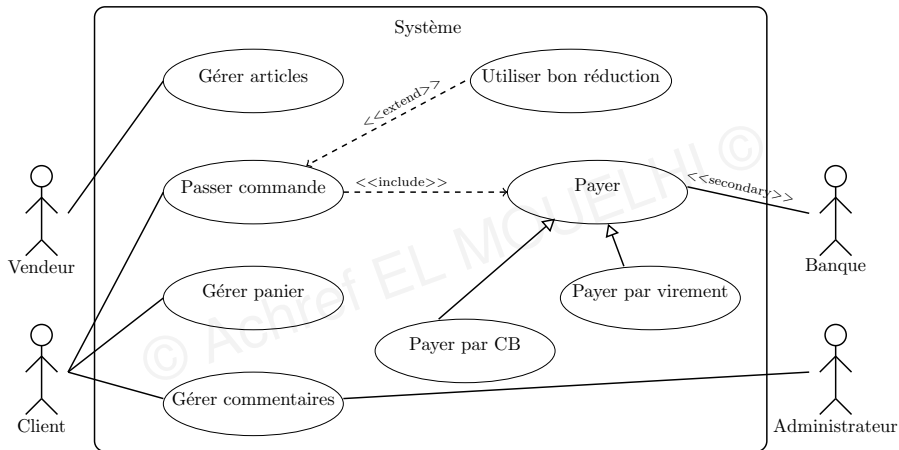
Un cas **A** est une généralisation d'un cas **B** si **B** est un cas particulier de **A**.



Le diagramme de cas d'utilisation de notre exemple



Le diagramme de cas d'utilisation de notre exemple



Pas d'ordre chronologique dans un diagramme de cas d'utilisation.

Relations entre acteurs : héritage

- Un acteur **A** hérite d'un acteur **B** si l'acteur **A** peut faire tout ce que l'acteur **B** a le droit de faire
- Tous les cas d'utilisation accessibles à **B** le sont aussi à **A** (la réciproque est fausse)

© Achref EL

UML

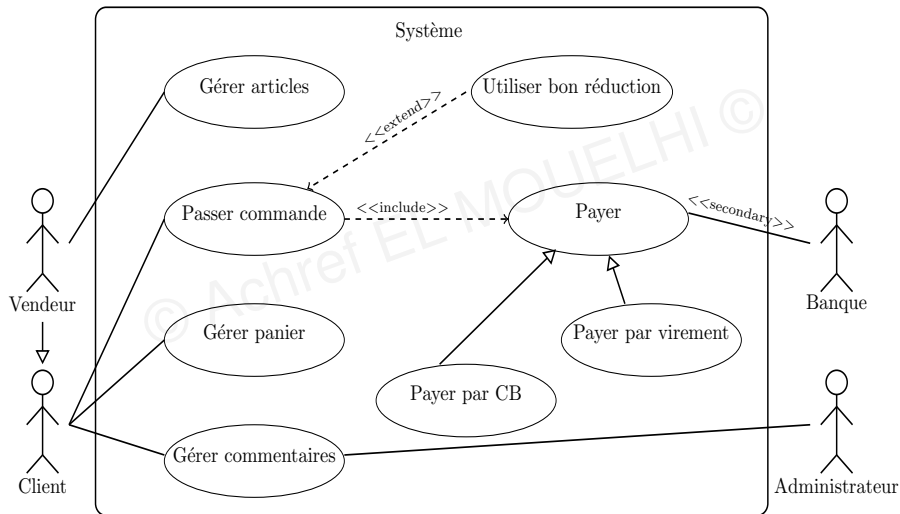
Relations entre acteurs : héritage

- Un acteur **A** hérite d'un acteur **B** si l'acteur **A** peut faire tout ce que l'acteur **B** a le droit de faire
- Tous les cas d'utilisation accessibles à **B** le sont aussi à **A** (la réciproque est fausse)

Revenant à l'exemple précédent

- Si on suppose que le vendeur peut aussi passer de commandes et commenter les articles
- Alors le vendeur peut hériter de client

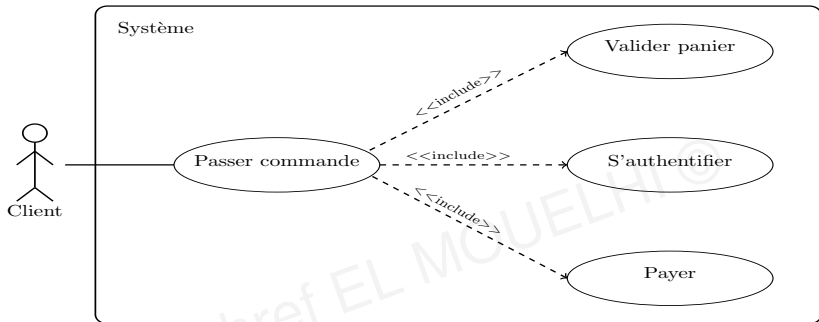
Nouveau diagramme de cas d'utilisation



Comment construire un diagramme de cas d'utilisation ?

- Identifier les acteurs et les classifier
- Trouver les cas d'utilisation (et les classifier en package)
- Établir les relations entre les cas d'utilisation et les acteurs
- Identifier les relations de dépendance entre les cas d'utilisation
- Vérifier s'il est possible de simplifier le diagramme avec la relation d'héritage
- Identifier les cas d'utilisation les plus complexes pour les détailler avec les diagrammes de séquence et d'activité

UML



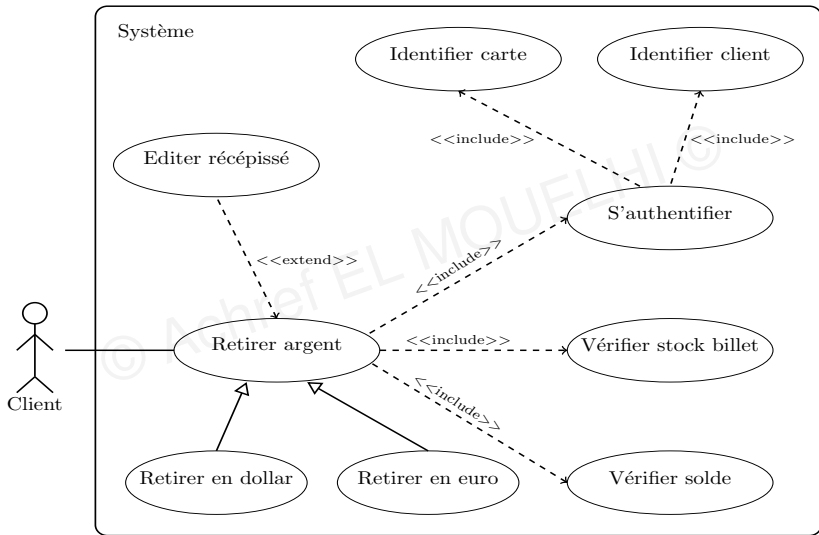
Exercice 1

- Le système permet-il au client de payer ses commandes ?
- Le système permet-il au client de suivre ses commandes ?
- Le client est-il obligé de valider son panier avant de payer ?
- Le client est-il obligé de valider son panier pour passer sa commande ?

Exercice 2

- Déterminer le diagramme de cas d'utilisation d'un distributeur de billets.
- On considère le scénario où le client désire retirer de l'argent en euro ou en dollar, bien sûr, cela nécessite une authentification.
- Il faut aussi traiter le cas où le stock de billet et/ou le solde sont insuffisants.
- Le client peut, s'il le désire, imprimer un récépissé.

Une solution possible



Exercice 3

- Déterminer le diagramme de cas d'utilisation d'une caisse enregistreuse.
- Le caissier enregistre les articles. Ensuite, pour payer, il faut signaler la fin de vente.
- Une fois la fin de vente est signalée, il est possible d'utiliser un bon de réduction.
- Trois modes de paiement sont autorisés : liquide, chèque et carte bancaire. Ce dernier nécessite de contacter le centre d'autorisations bancaires.
- Pour chaque vente, et après paiement, il faut transmettre les infos au gestionnaire de stock.

Une solution possible

