

UML: Exercices

Solutions des Exercices UML



Table des matières

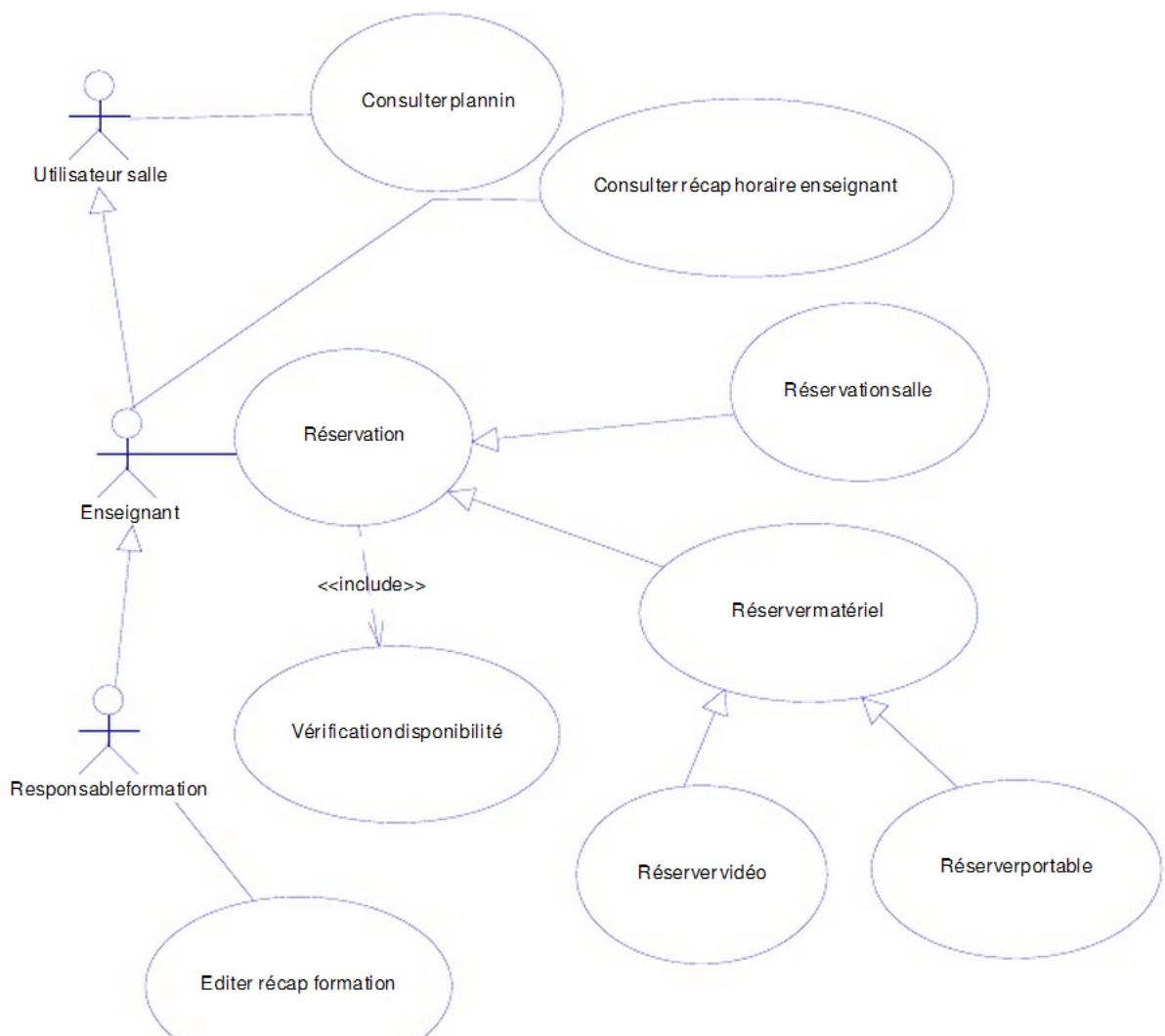
EXERCICES DE MODÉLISATION	3
DIAGRAMME DE USE CASE	3
DIAGRAMME D'ACTIVITÉ	11
DIAGRAMME DE SÉQUENCE	24
DIAGRAMME D'ETAT	28
EXERCICE INTEGRE	32

UML SOLUTIONS

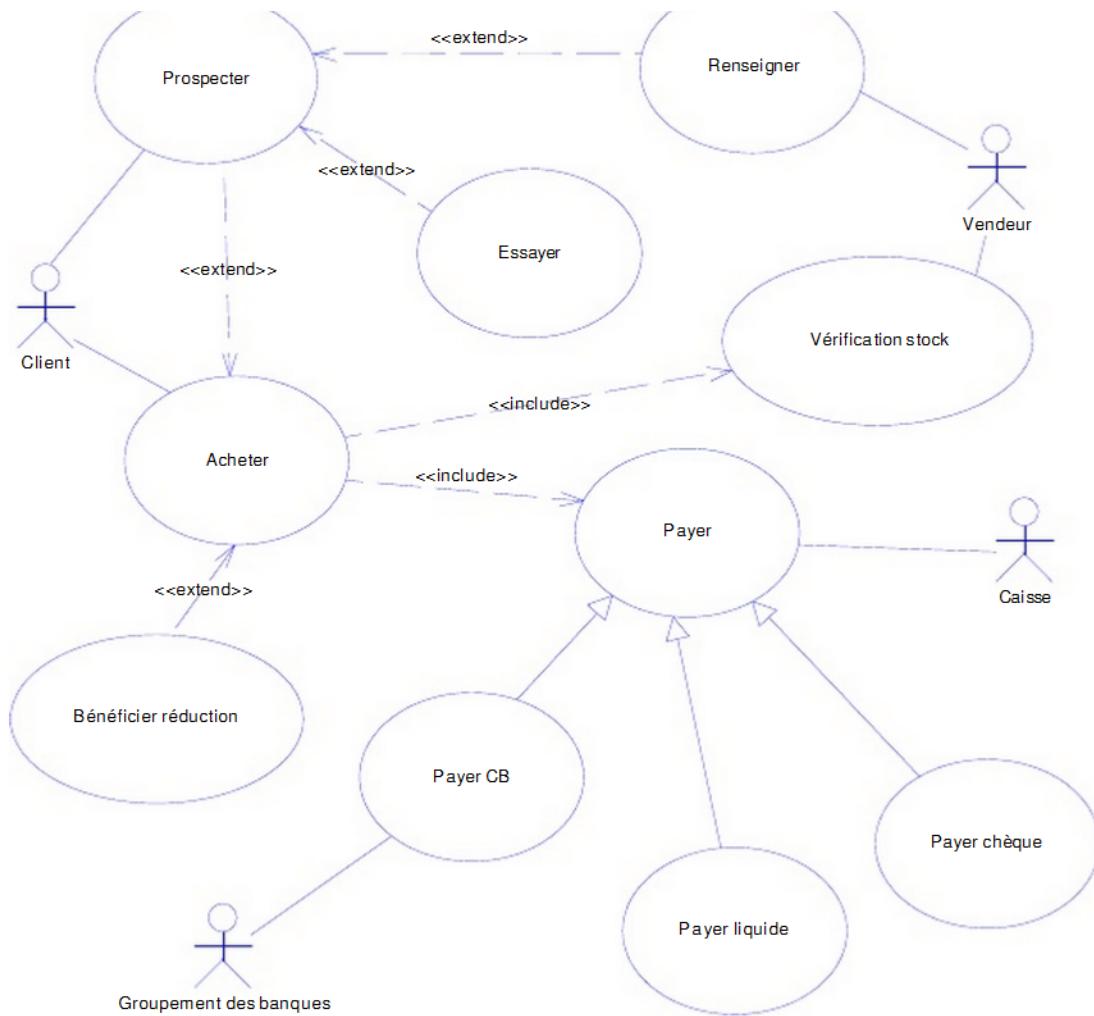
Exercices de Modélisation

Diagramme de Use Case

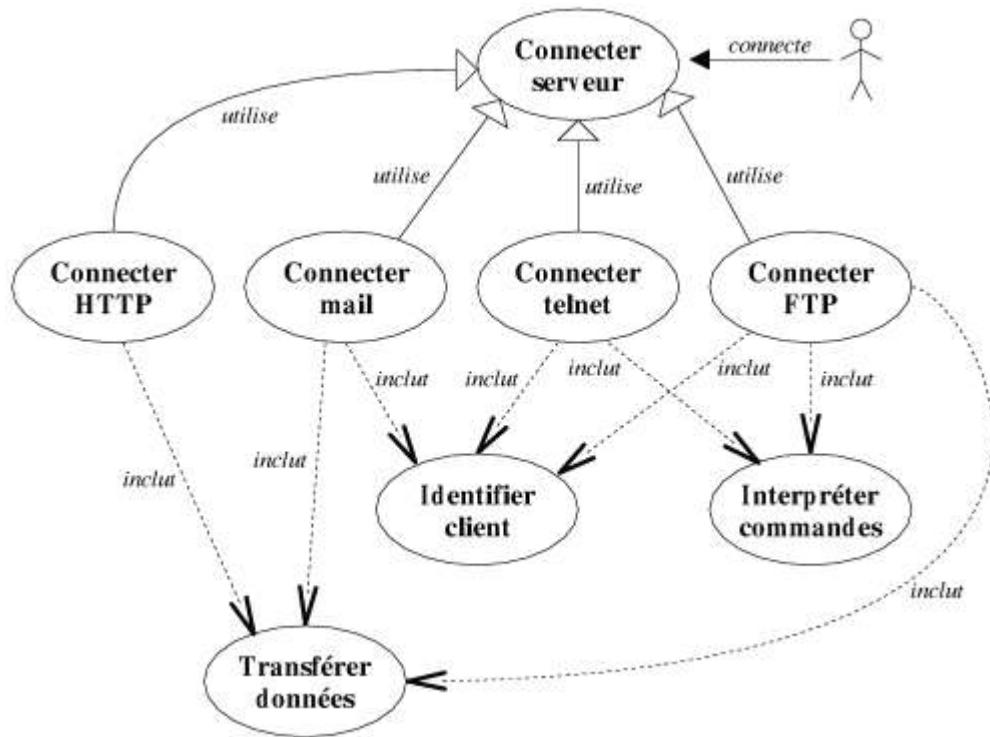
1) Gestion de salles de cours



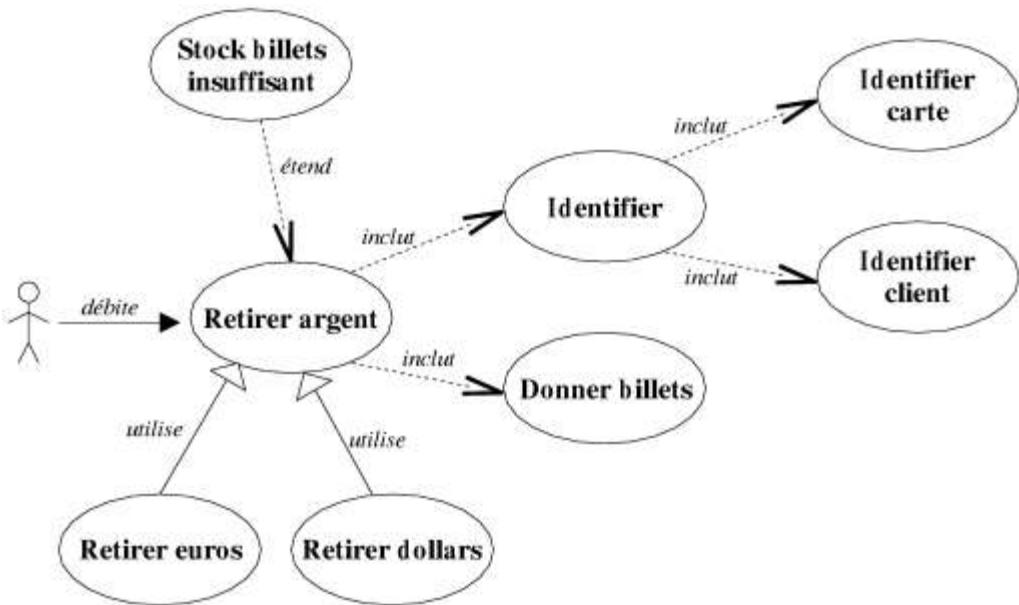
2) Le processus de vente



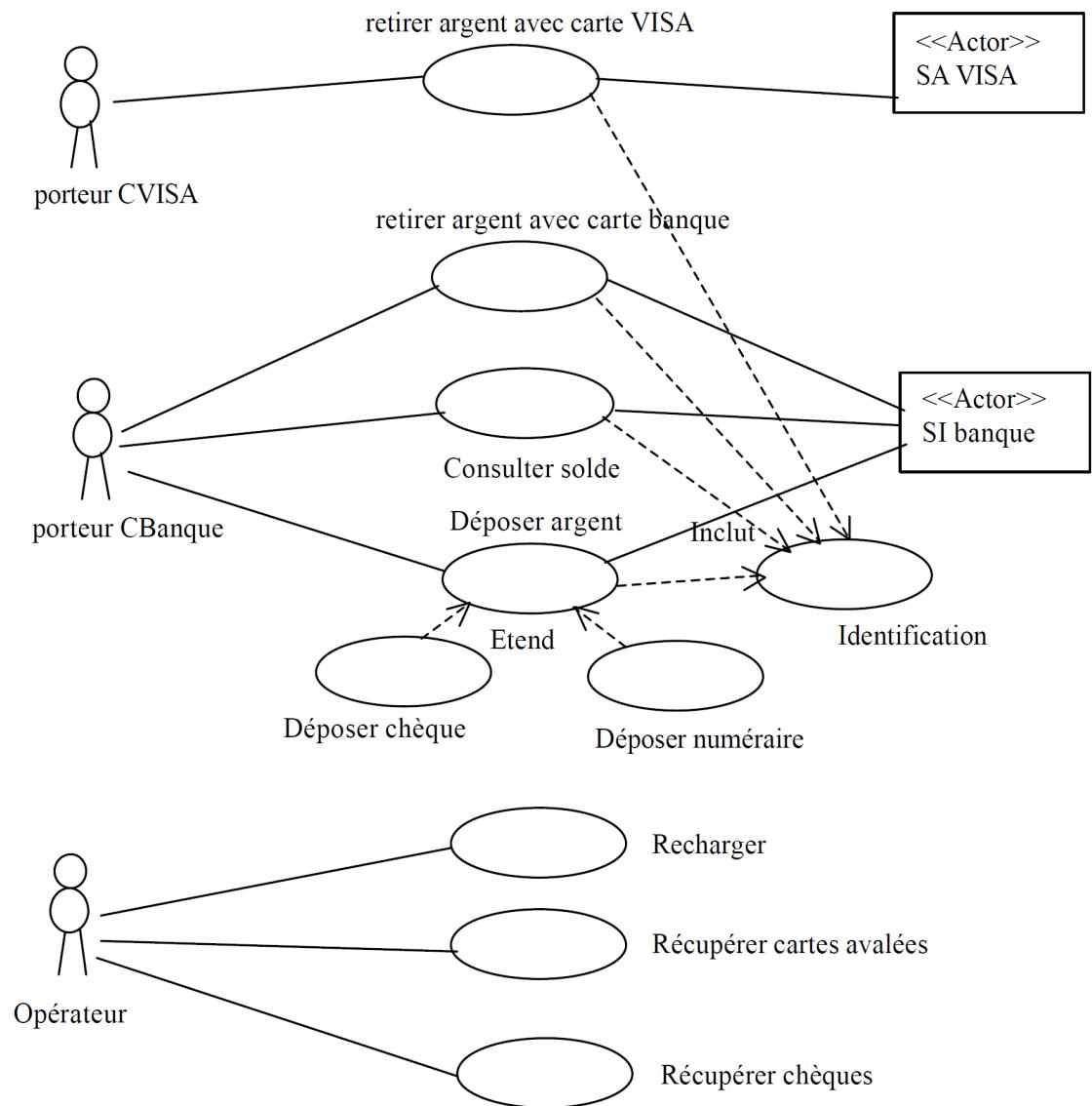
3) Connexion à un serveur



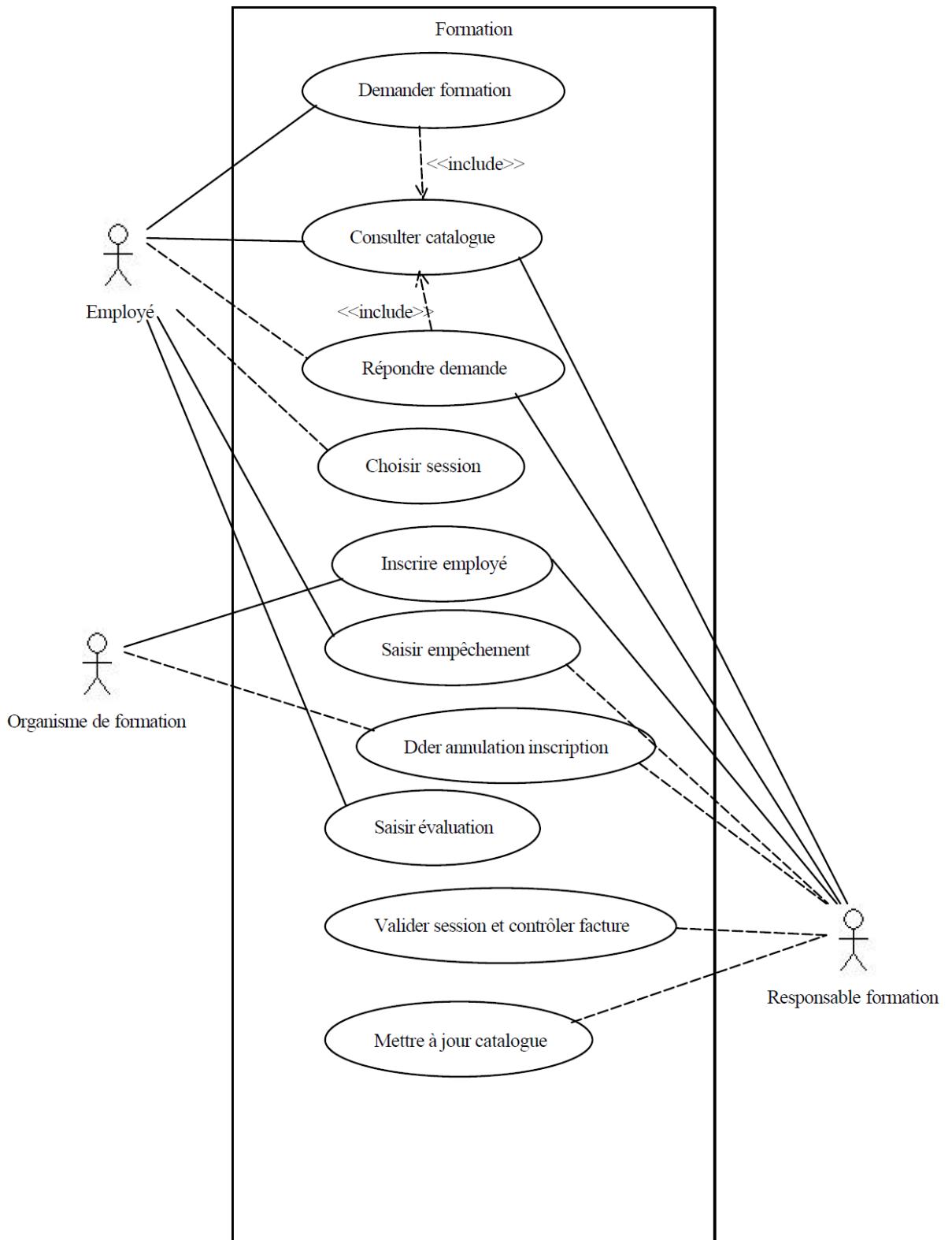
4) Le distributeur de billets



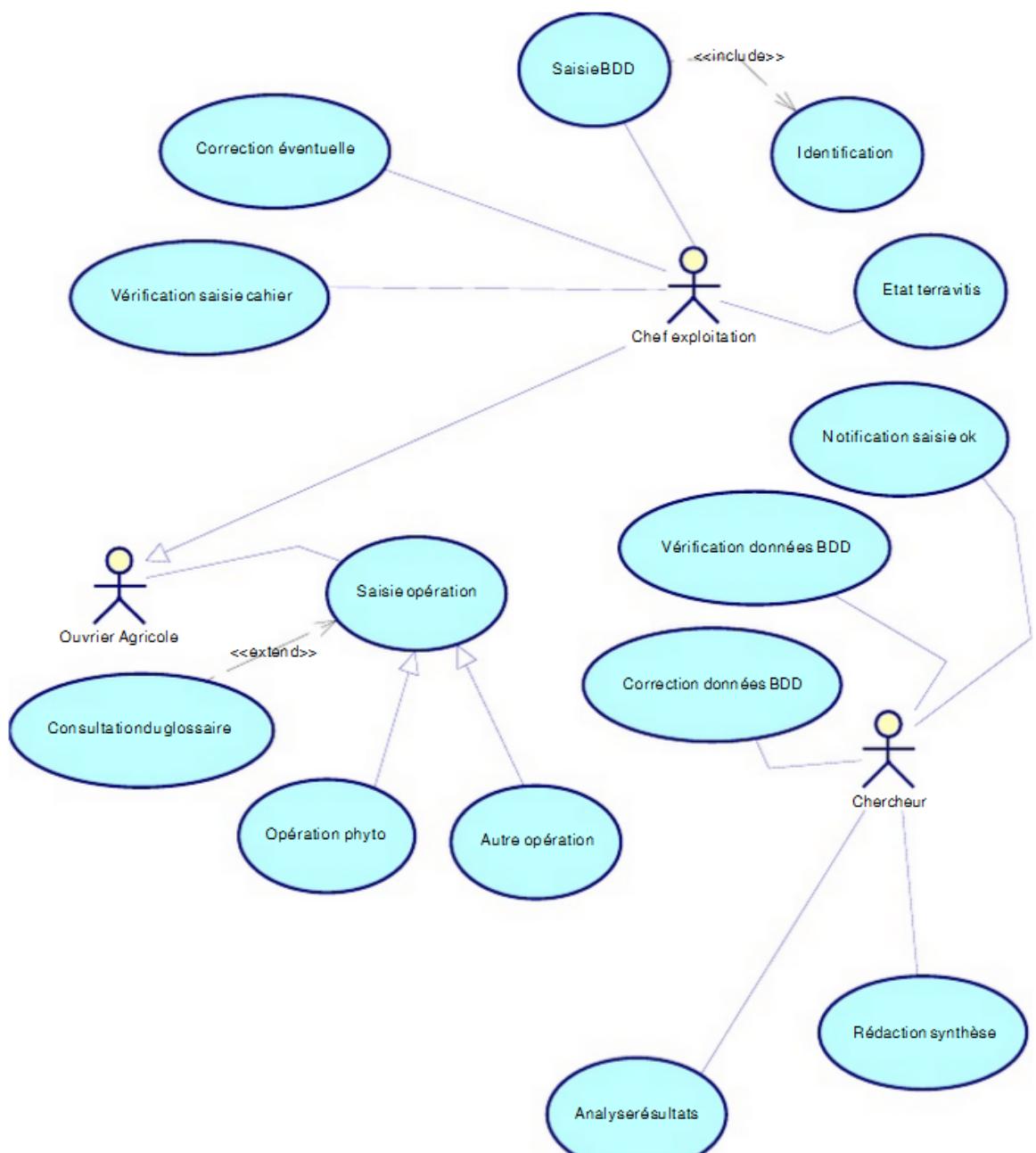
5) L'ATM : distributeur de billets avancé



6) L'organisation de formation



7) Projet de recherches en viticulture



8) Les emprunts dans une bibliothèque universitaire

Marche à suivre :

[1] Prendre connaissance de l'expression initiale des besoins.

[2] Identifier les acteurs (Rôles).

[3] Identifier les uses cases.

Trouver les Use Cases à l'initiative des différents rôles (y compris le Si à concevoir lui-même).

Recommandation : Nommer les Use Cases d'après le but de l'acteur plutôt que du point de vue du système, par exemple : "Emprunter un livre" plutôt que "Prêter livre"

[4] Établir le diagramme des use cases

- a. il s'agit d'une première mouture de la vue d'ensemble des UC. Ce schéma va certainement être amené à évoluer par la suite.
- b. Éviter d'identifier les sous-UC ainsi que les "includes" et les "extends" à priori ! Cela ne devient clair que lors de la description textuelle des Use Cases. En plus, cela surchargerait la vue d'ensemble des Use Cases et la rendrait moins lisible

[5] Faire la description textuelle des Use Cases.

[6] "*Emprunter livre sans réservation*"

- a. On commence à raisonner sur le cas le plus simple possible ("happy scénario")
- b. tout se passe comme prévu + pour l'instant emprunt d'un seul livre à la fois

[7] Post-condition : distinguer

- a. post-condition minimale : vraie dans tous les cas (à remplir à posteriori : seulement lorsque tous les UC sont spécifiés)
- b. post-condition spécifique : vraie dans le cas traité (l'UC nominal ou une des exceptions/alternatives ↗ à répéter pour chaque exception/alternative)

[8] Flux exceptionnelles/alternatifs :

On devrait maintenant considérer les cas alternatifs/exceptions. Un des premiers qui vient à l'esprit est l'emprunt de plusieurs livres en une fois. Comme ce cas est extrêmement fréquent et qu'il n'est en fait qu'une généralisation du cas de base, on décide de revoir le cas de base.

[9] On pourrait continuer arbitrairement loin dans les exceptions (par ex. Annulation possible à tout moments, etc.) et les détails (par ex. différents modes de paiement) mais il ne faut pas perdre de vue que les Use Cases :

- a. servent à donner une vue d'ensemble de la spécification mais ne doivent pas, à ce niveau-ci, décrire les détails, en particulier, les détails de l'interface (Par ex. comment se fait le paiement).

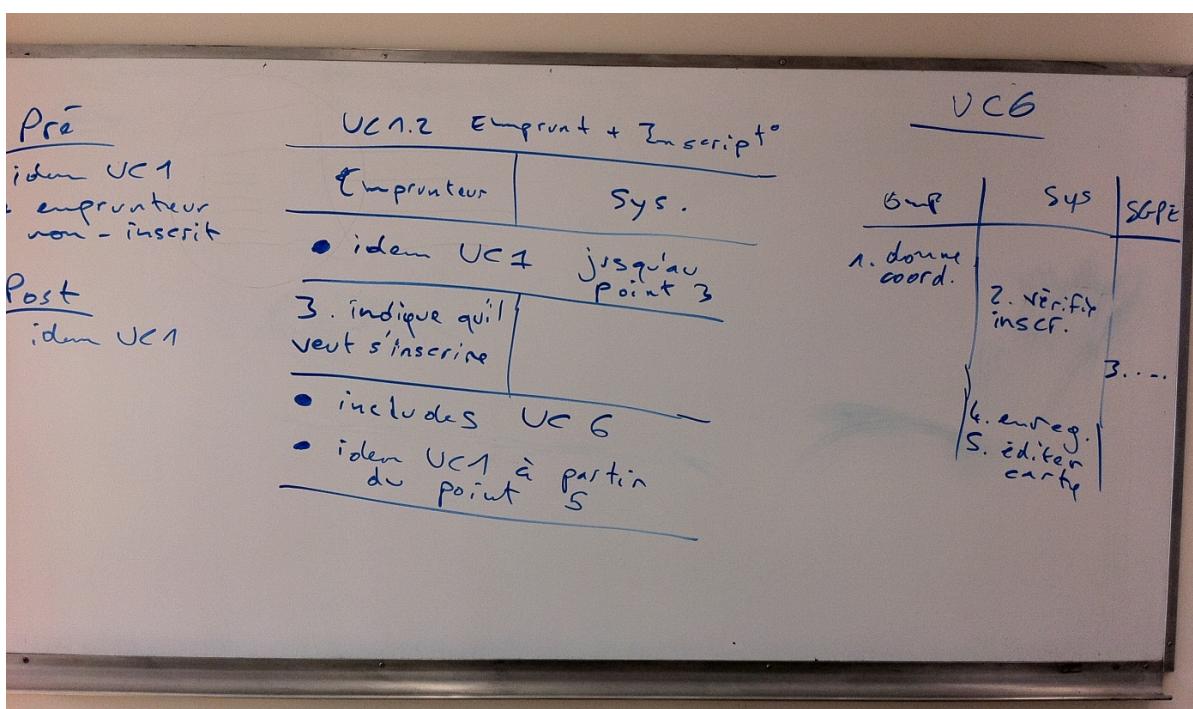
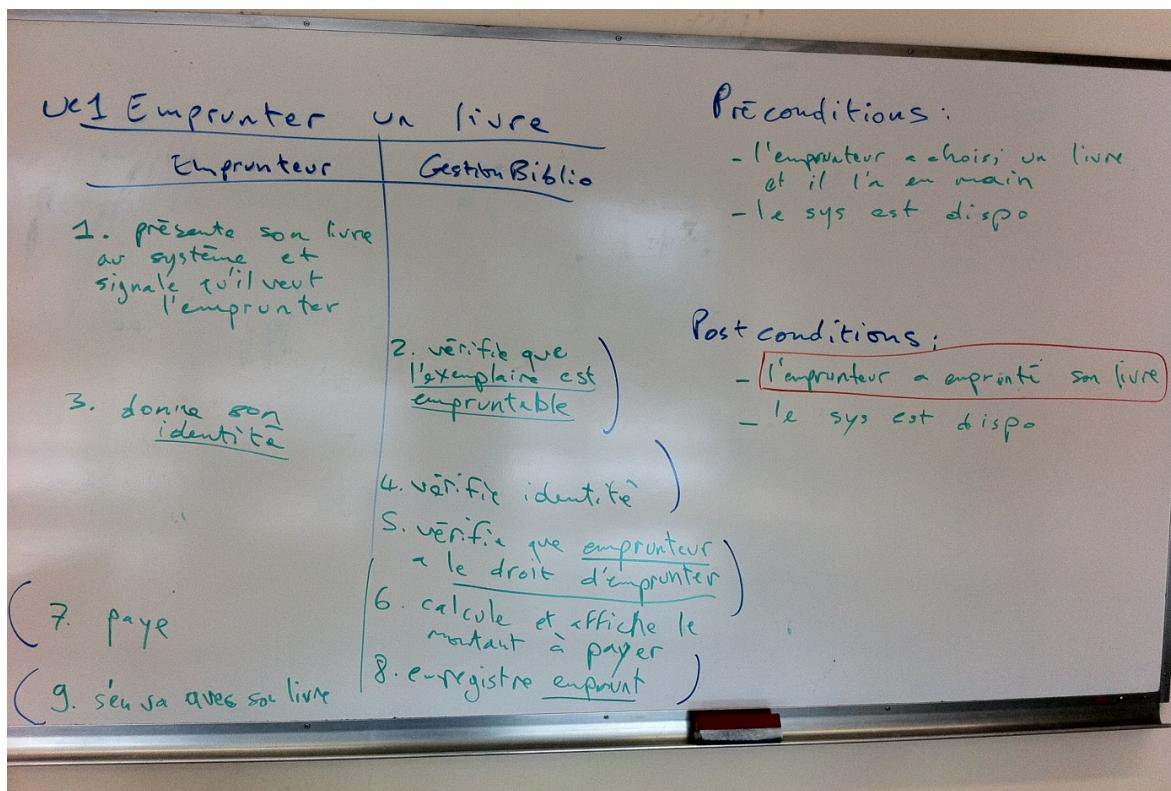
Entrer dans des UC trop détaillés compromettrait la simplicité et la lisibilité des UC qui sont leurs deux qualités principales

- b. sont plus adaptés pour donner des exemples que des généralisations du comportement : problématique de la combinatoire des exceptions.
- c. Si plus de détails sont nécessaires à ce stade (c.-à-d. Pour des développements critiques), il est recommandé de recourir à des notations plus précises (ie, autres diagrammes)

=>le tout est une question de compromis : niveau de détail vs. Lisibilité et coût de spécification

[10] Post-condition minimale : (qui doit être vérifié par tous les UC relatifs à ce cas, elles ne peuvent être déduite qu'à la fin de la spécification des exceptions)

[11] Tous ces UC peuvent être placé dans un premier package : dans le Use Case Diagramme du package, on fait de chaque exception un Use Case séparé. Dans le Use Case principal, il n'est pas recommandé de montrer ces exceptions où alors seulement quelques unes particulièrement importantes (en termes de fréquence d'utilisation)



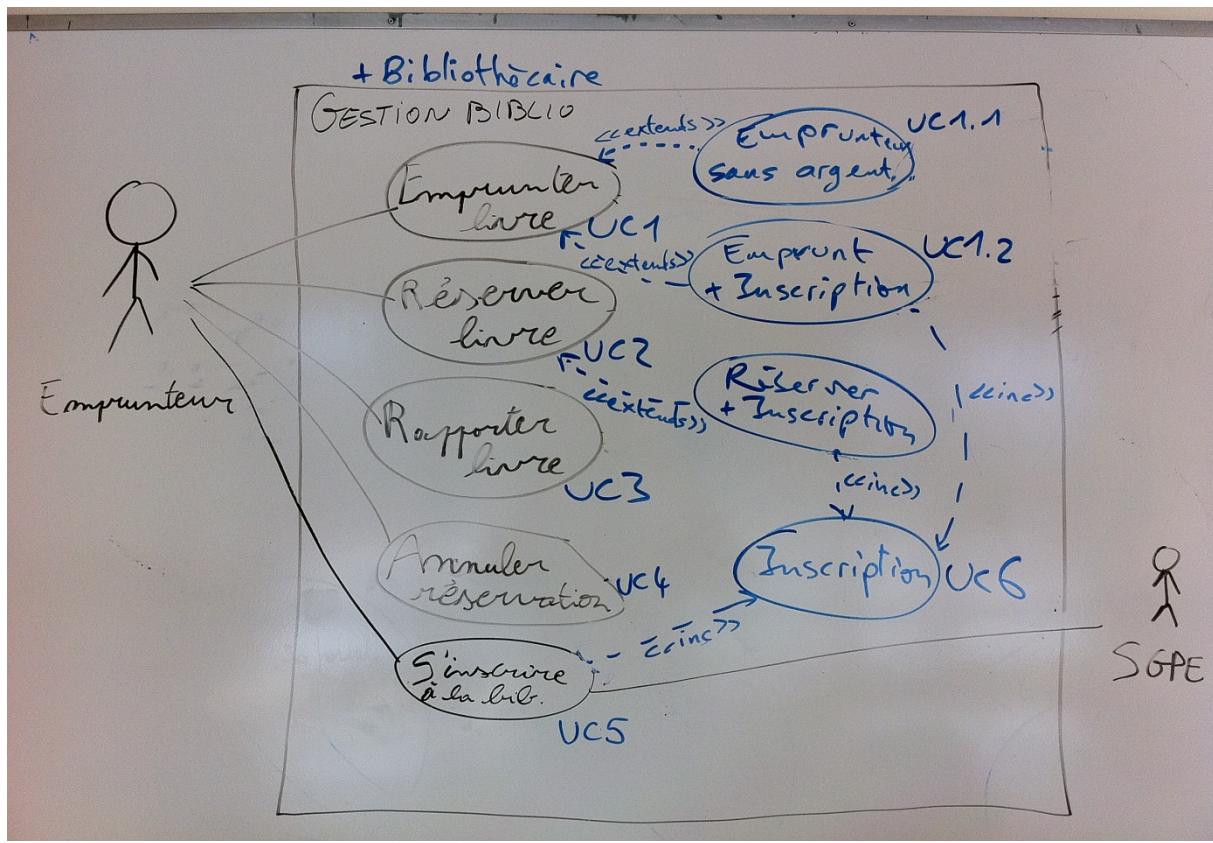
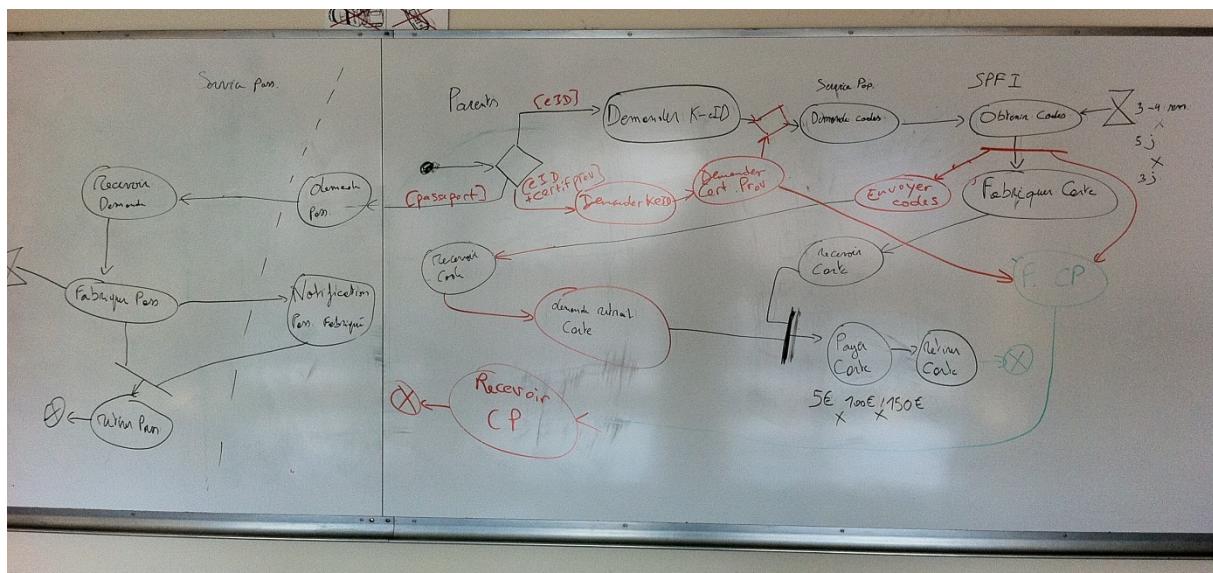
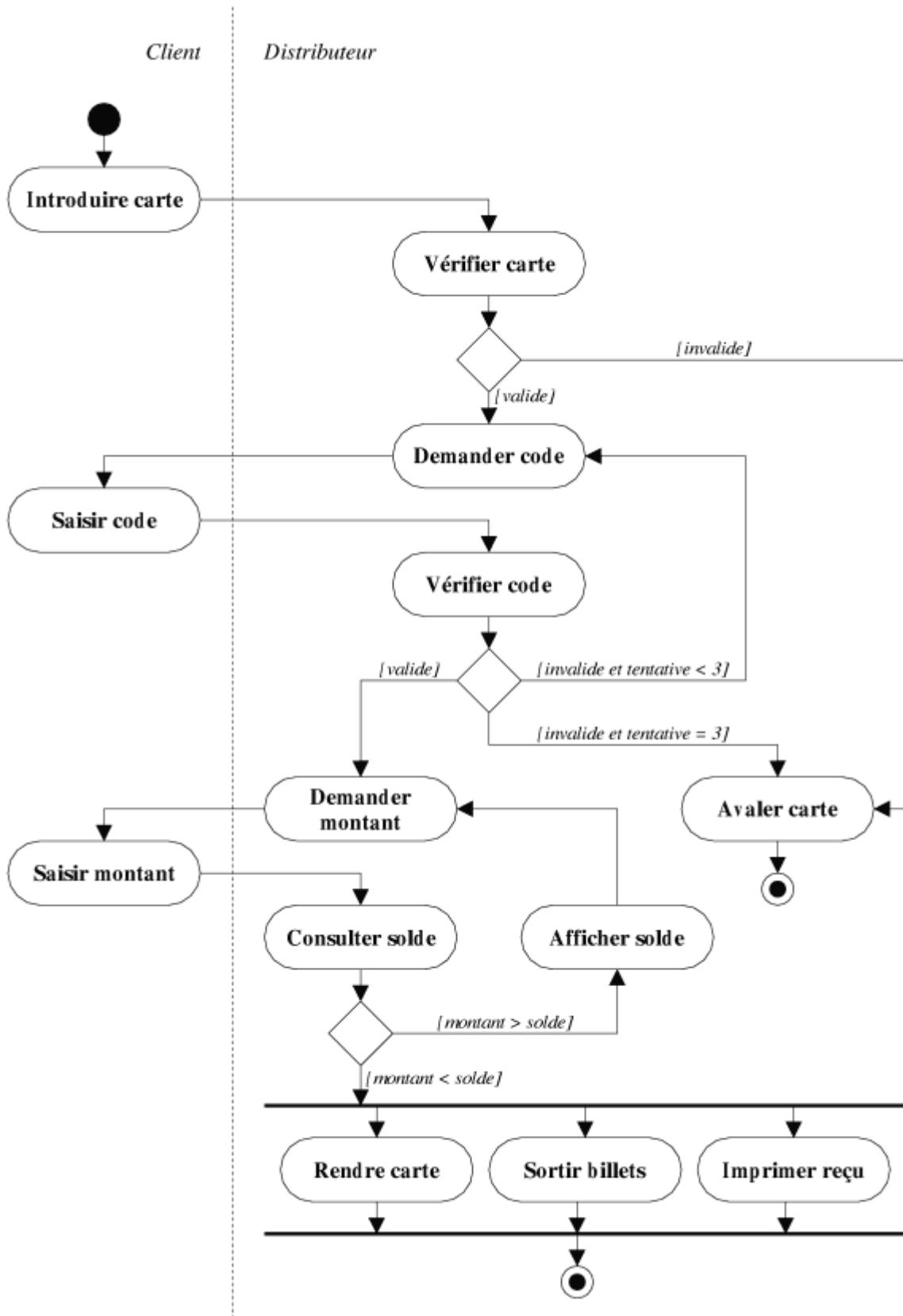


Diagramme d'Activité

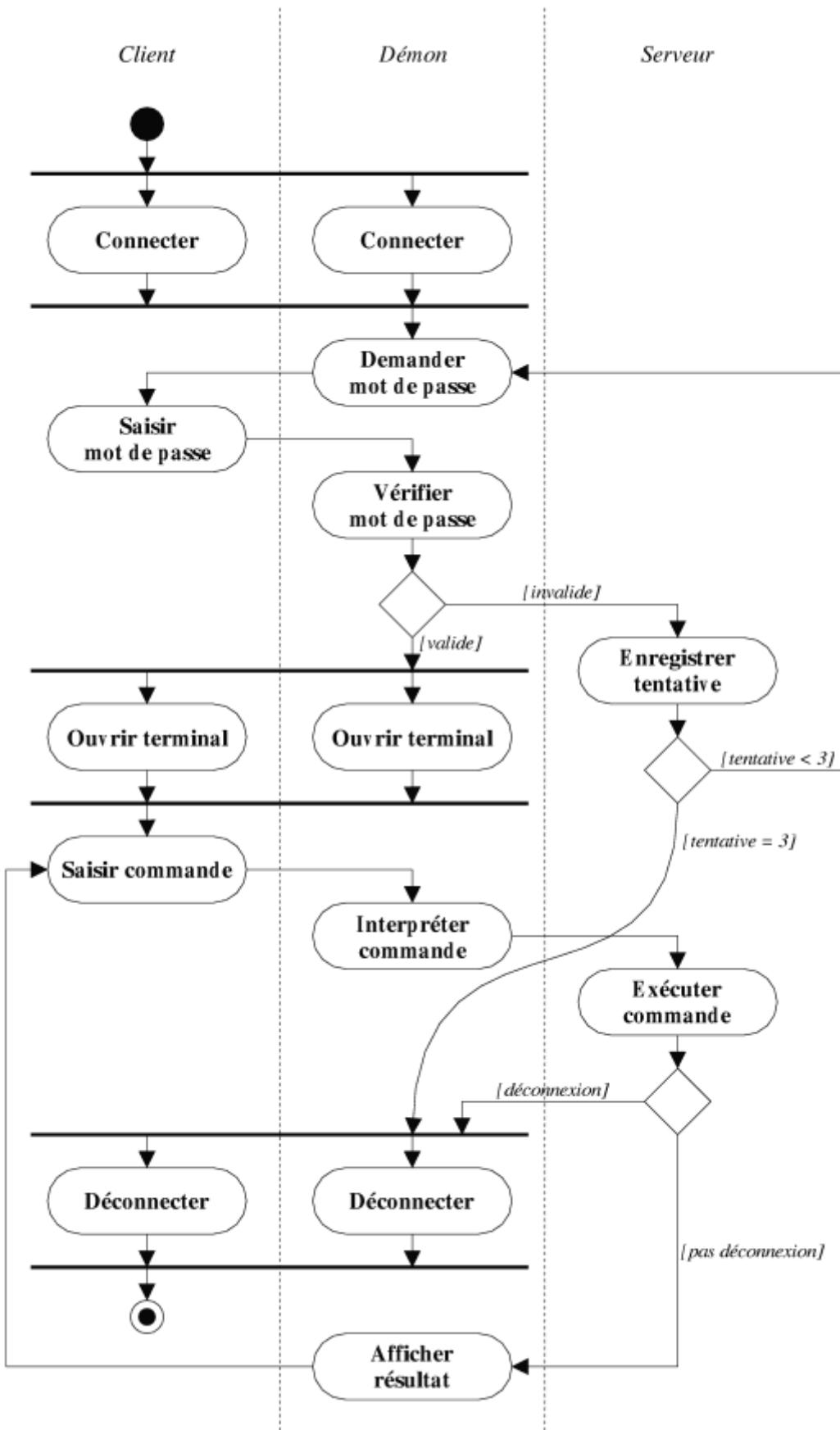
9) Identité des enfants



10) Le fonctionnement d'un distributeur de billets



11) Connexion à Telnet



12) Gestion des sinistres dans une compagnie d'assurance

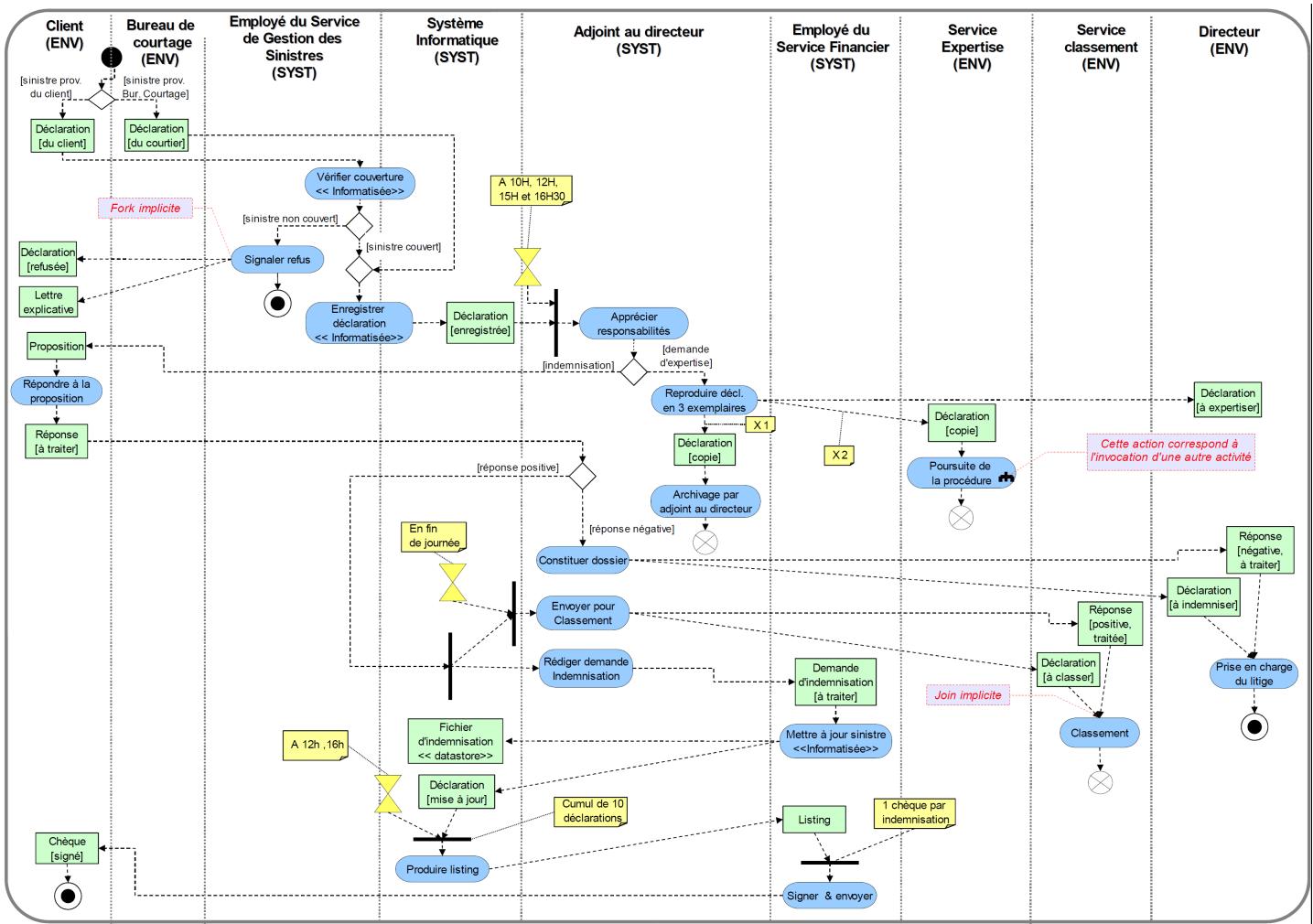
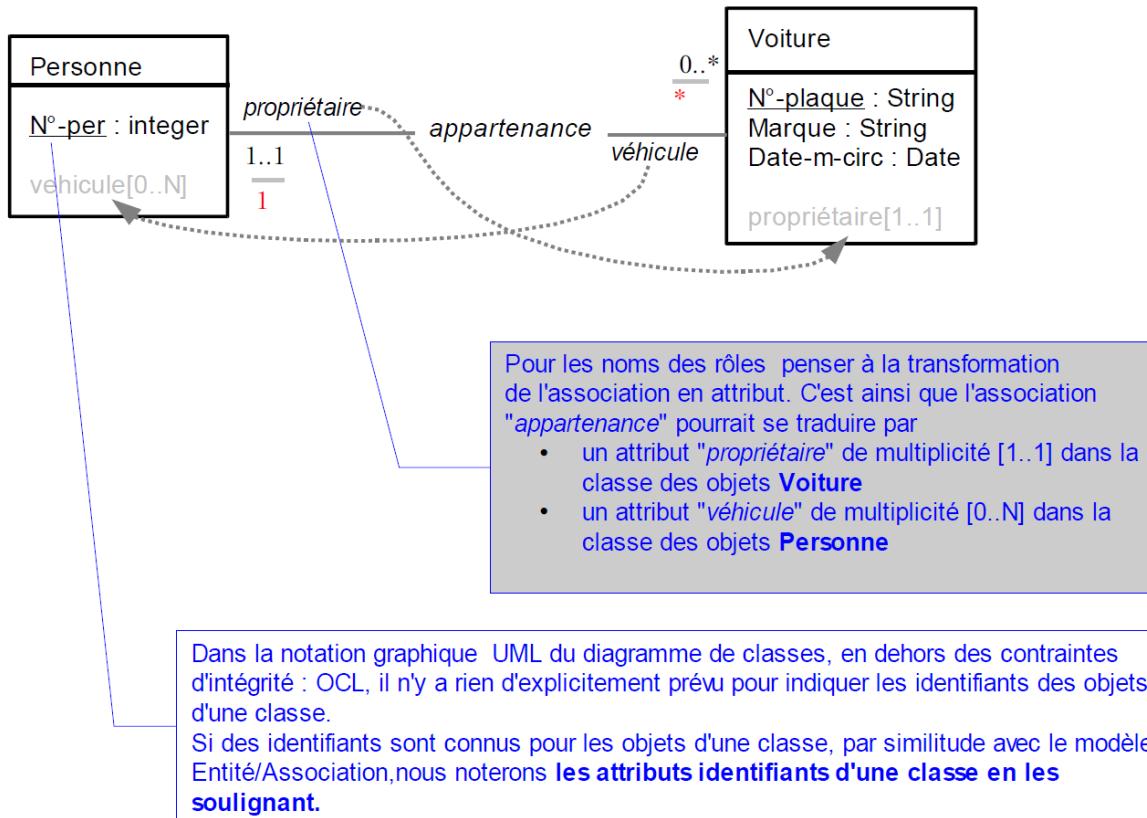
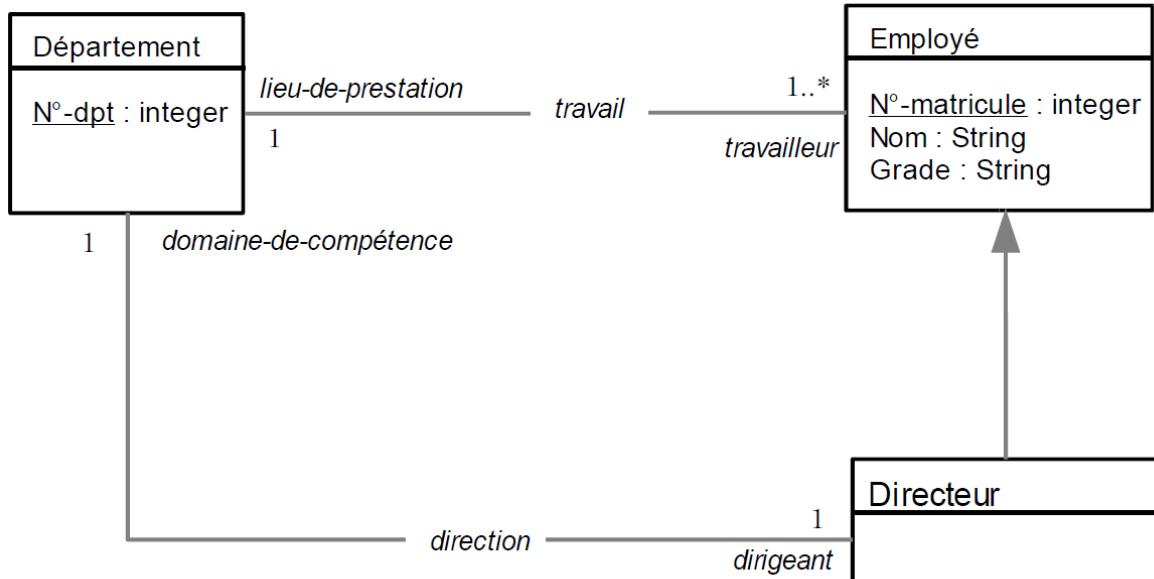


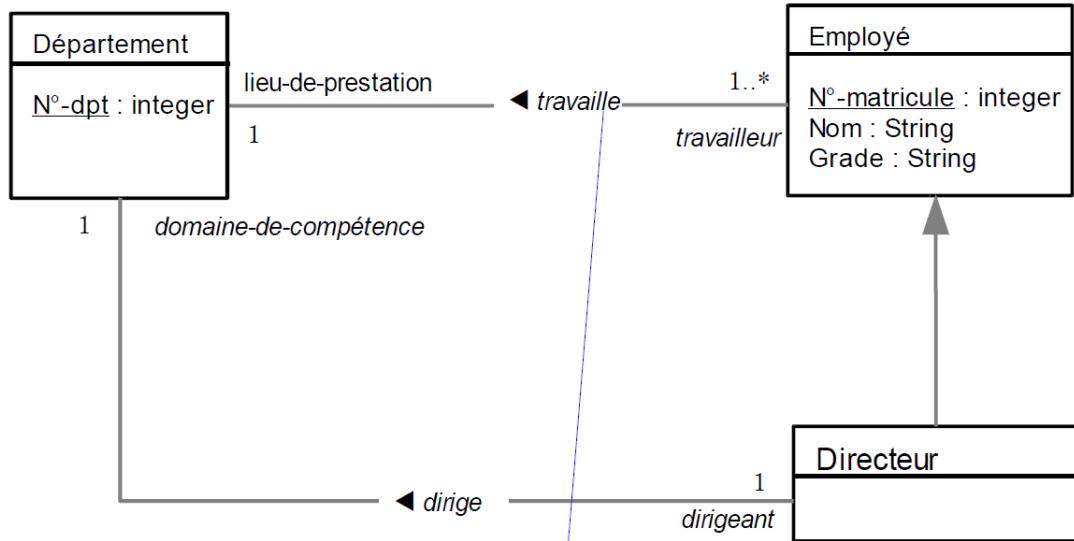
Diagramme de Classe

13) Propriétaire de voiture



14) Classification des employés





Cette notation est équivalente à la précédente, la seule différence est que le nom de l'association ne prend son sens qu'en le lisant dans le sens indiqué par la flèche.

C'est ainsi que :

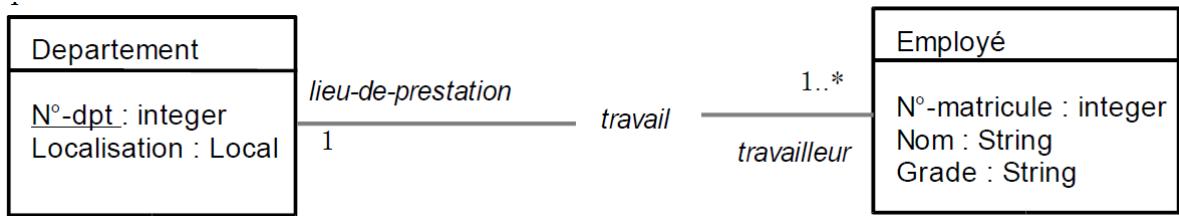
- **Employé** *travaille* dans un **Département**
- et pas un **Département** qui *travaille* dans **Employé**

Contrainte d'intégrité :

Un directeur (en tant qu'employé) a comme *domaine de compétence* le même département que celui qui est son *lieu de prestation*

$$\forall \text{dir} \in \text{Directeur}: \text{dir. domaine-competence} \Rightarrow \text{dir. lieu-de-prestation}$$

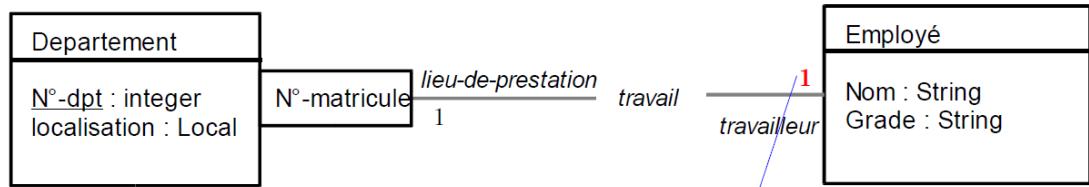
15) Organisation des départements



Contrainte d'intégrité :

L'identifiant d'un Employé est constitué de son N°-matricule et de l'identifiant du département qu'il a comme *lieu de prestation*

- Expression de la C.I dans le formalisme du modèle E/A : $ID(Employé) = N°\text{-matricule} + \text{un rôle ...}$
 - Expression de la C.I dans un formalisme mathématique :
- $$\neg \exists e, e' \in \text{Employé} : (e \neq e') \wedge (e.N°\text{-matricule} = e'.N°\text{-matricule}) \wedge (e.\text{lieu-de-prestation} = e'.\text{lieu-de-prestation})$$



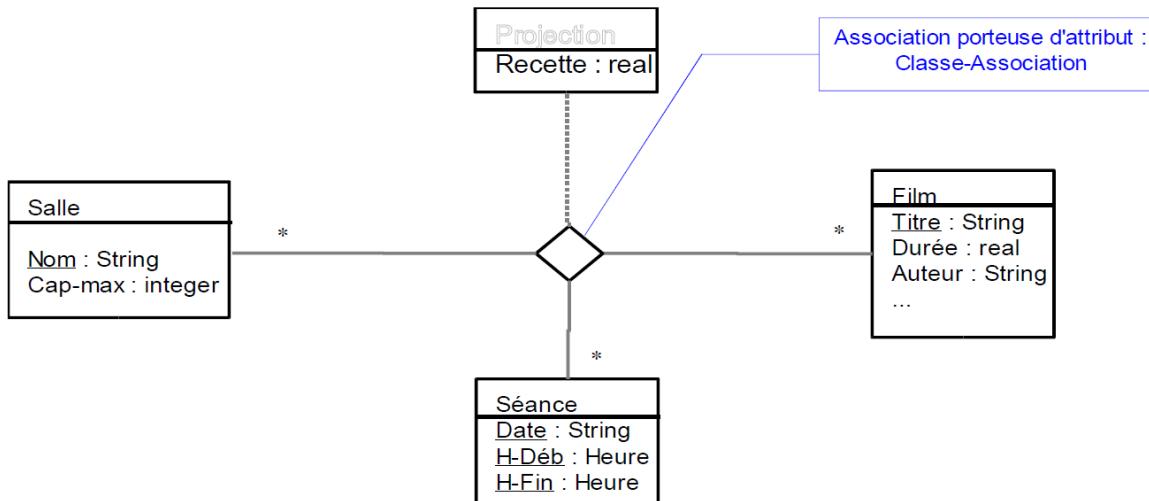
Pour l'association **travail**, un **Employé** ne peut jouer qu'une et seule fois rôle *travailleur* dans le tuple (**Département**, **N°-Matricule**).

\Rightarrow l'employé est identifié par l'identifiant du département et par son N°-matricule.

Tentative d'approche UML pour modéliser la notion de rôle identifiant dans l'Entité/Association

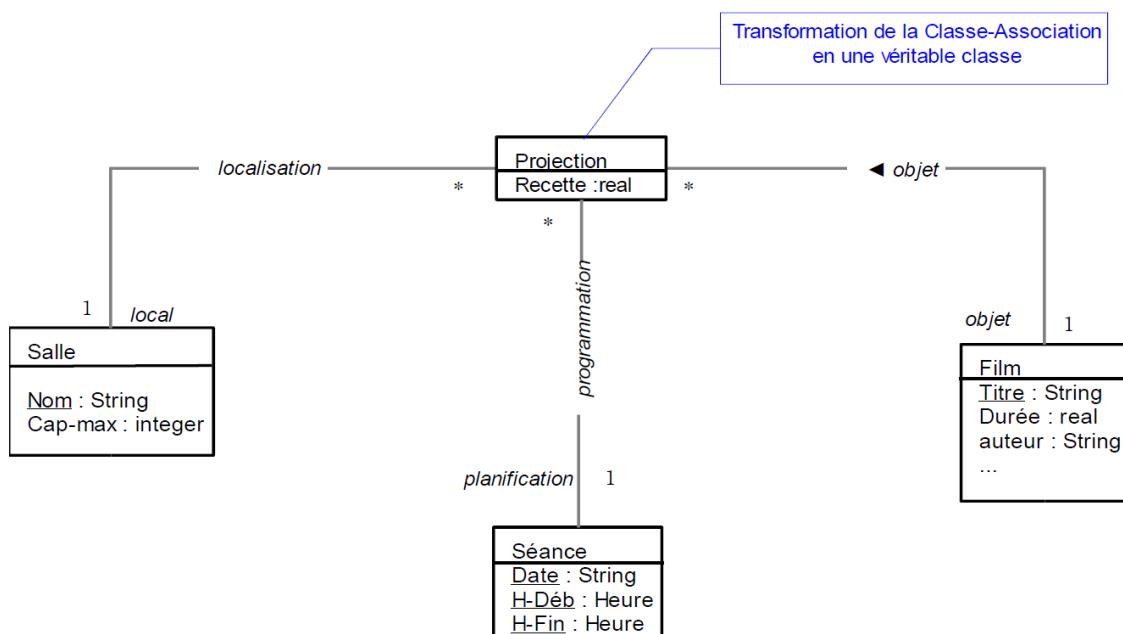
16) Le complexe de cinéma

1^{ière} solution



Problèmes de la solution n°1 :

- La principale difficulté est de nommer la classe association, en effet sans ce nom, il est pratiquement interdit de s'exprimer sur cette association.
- Le schéma ainsi obtenu ne facilite pas la communication avec les partenaires. Les concepts utilisés (les rôles) n'étant pas nommés il est très difficile de s'exprimer sur ces concepts.
- Les multiplicités des rôles peuvent s'interpréter comme suit :
 - un film peut être associé à 0 ou plusieurs doublets (Salle, Séance)
 - une Séance peut être associé 0 ou plusieurs doublets (Salle, Film)
 - une salle peut être associée à 0 ou plusieurs doublets (Séance, Film)
- 2^{ième} solution



Un objet de la classe Projection correspond donc à la projection d'un film donné dans une salle donnée et à une séance donnée.

Contraintes d'intégrité :

- Parce que l'association ternaire Projection est la traduction UML d'une relation ternaire dans le modèle Entités/Associations et que dans ce modèle E/A toute relation est implicitement identifiée par les rôles qu'elle joue, nous identifierons "naturellement" l'association ternaire UML par ses trois rôles.
 - Expression de cette C.I. en français: L'identifiant de la classe Projection est constitué de la réunion de l'identifiant du film qu'elle a pour *objet*, de l'identifiant de la Salle qu'elle a pour *local* et de l'identifiant de la séance qu'elle a comme *planification*
 - Expression de la C.I. dans un formalisme mathématique :

$$\neg \exists p, p' \in \text{Projection} : (p \neq p') \wedge (p.\text{objet} = p'.\text{objet}) \wedge (p.\text{local} = p'.\text{local}) \wedge (p.\text{planification} = p'.\text{planification})$$

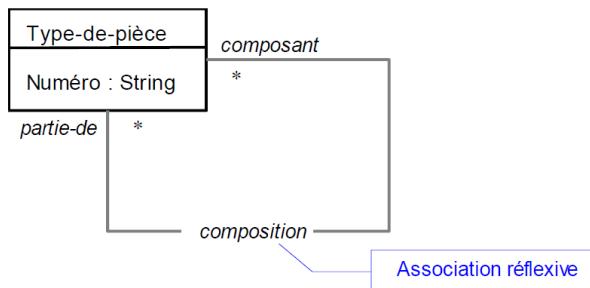
Attention:

Avec cette forme d'identifiant il n'est cependant pas interdit d'avoir plusieurs projections dans une même salle programmées à une même séance, pourvu que les films projetés diffèrent. Pour éviter ce dysfonctionnement, il faut admettre que l'identifiant d'une projection n'est pas l'identifiant "implicite" mais un identifiant "réduit". L'identifiant réduit de la classe projection est constitué de la réunion de l'identifiant de la Salle qui son *local* et de l'identifiant de la Séance qui est sa planification.

- Les plages de temps définies pour les séances d'une même journée ne peuvent pas se chevaucher.
- La durée d'un film ne peut pas excéder la durée de la séance pendant laquelle il est projeté.
 - Expression de la C.I. dans un formalisme mathématique :

$$\forall p \in \text{Projection} : p.\text{objet.durée} \leq (p.\text{planification.H-Fin} - p.\text{planification.H-Déb})$$

17) Les types de pièces

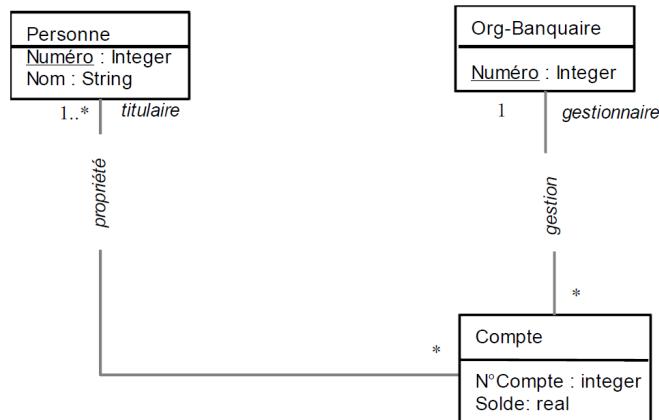


Contrainte d'intégrité :

- Un type de pièce ne peut pas être directement ou indirectement *partie-de* de lui-même.

Remarque : Il est heureux que les multiplicités des deux rôles soient * car toute tentative de remplacer une de ces deux multiplicités par 1..* conduit inéluctablement à concevoir une nomenclature infinie. En effet, avec ces multiplicités il ne pourrait pas exister de type de pièce sans composant ou de type de pièce qui ne soit partie-d' autre(s) type(s) de pièce.

18) Les comptes bancaires

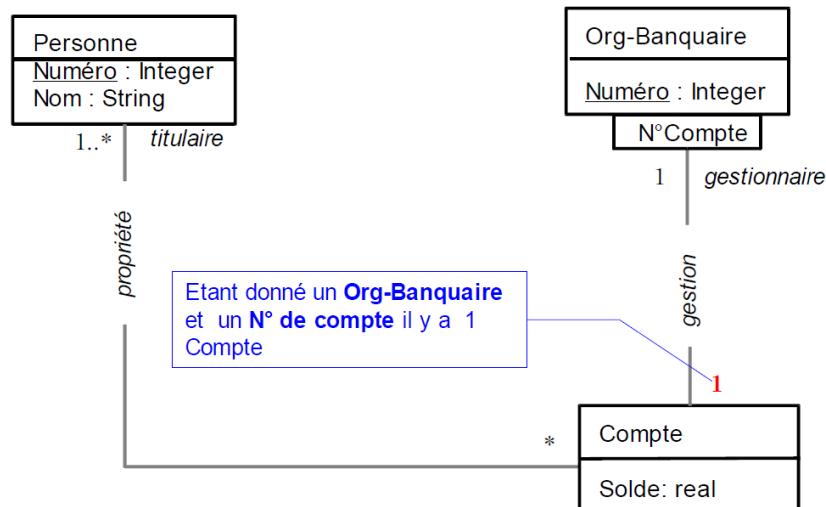


Contrainte d'intégrité :

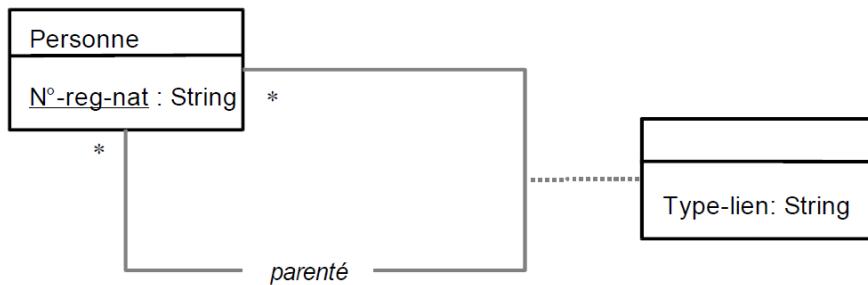
- L'identifiant d'un objet de la classe Compte résulte de la réunion du Numéro de ce compte et de l'identifiant de l'Org-Banquaire qu'il a comme *gestionnaire*.
 - Expression de cette C.I dans le formalisme E/A :

$$\text{ID(Compte)} = \text{Compte.Numéro} + \text{le rôle ...}$$
 - Expression de cette C.I sous forme mathématique :

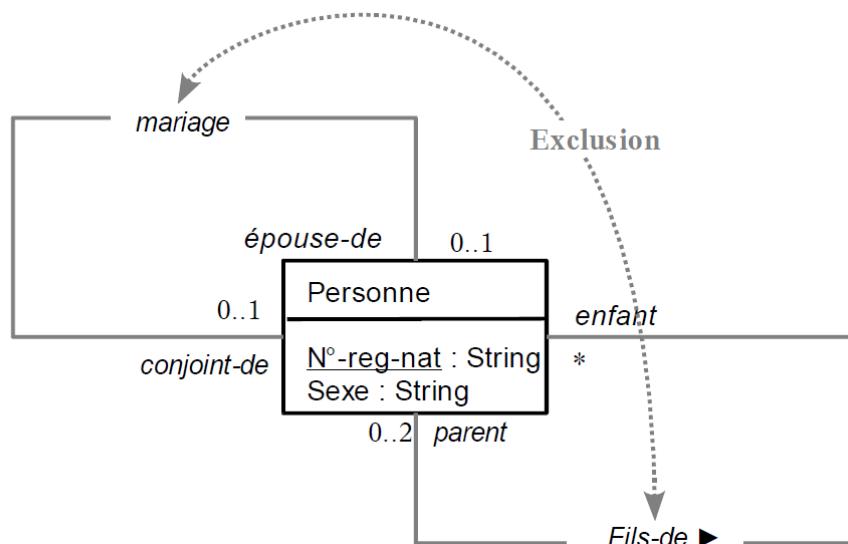
$$\neg \exists c, c' \in \text{Compte} : (c \neq c') \wedge (c.\text{Numéro} = c'.\text{Numéro}) \wedge (c.\text{gestionnaire} = c'.\text{gestionnaire})$$



19) Relations familiales



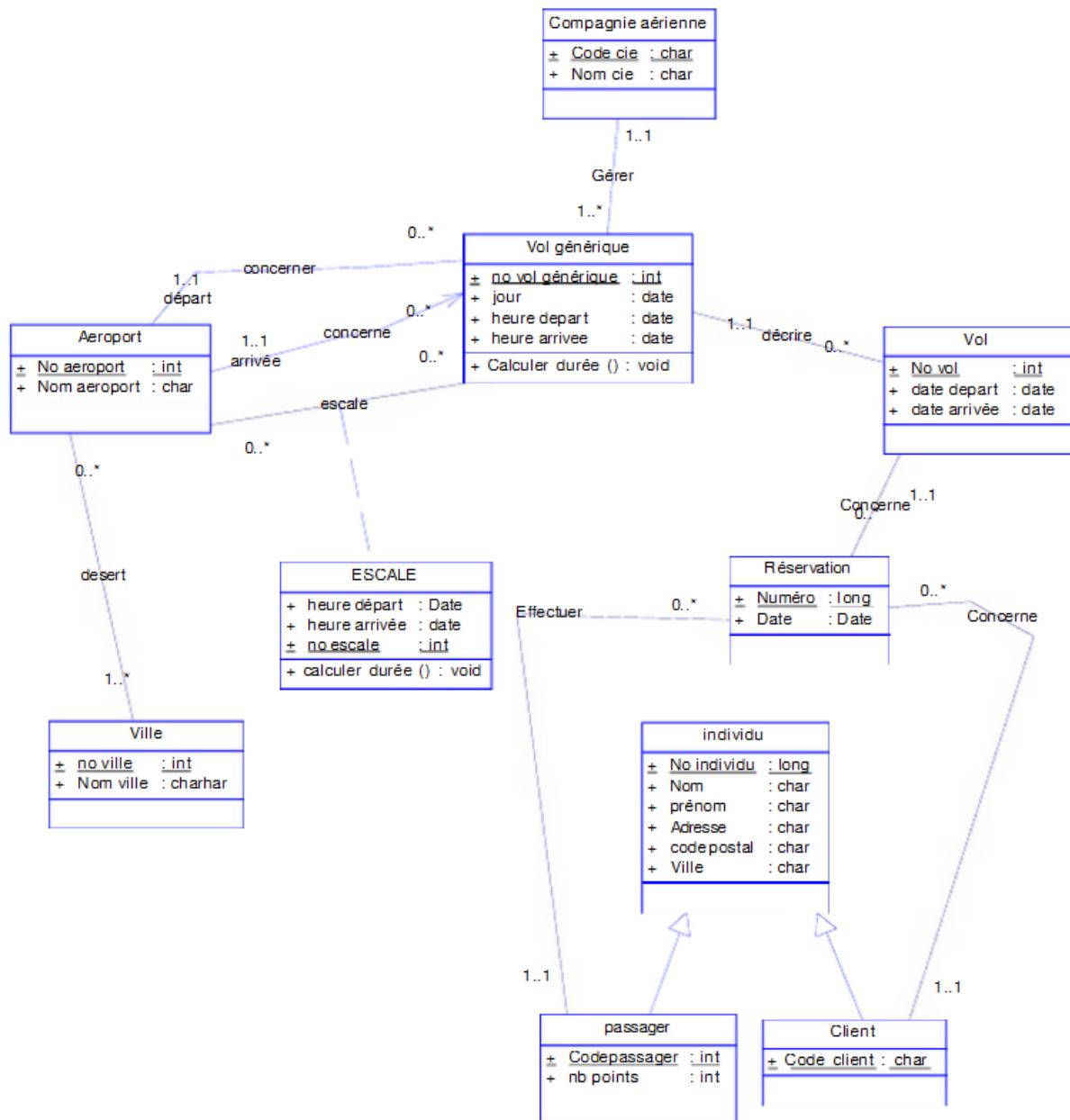
Cette solution n'est guère expressive (elle n'est pas un bon outil de communication). Pour refléter toutes les contraintes liées aux "interdits sociaux" elle devra être accompagnée de beaucoup de contraintes d'intégrité qui vont lier la valeur de l'attribut Type-lien (Mariage, Filiation, ...) aux multiplicités des rôles.



Contraintes d'intégrité :

- Il ne peut pas exister d'association de *mariage* entre deux personnes du même sexe.
- S'il existe deux personnes qui jouent le rôle *parent* pour une même autre personne, alors ces personnes là doivent être de sexe différent.
- S'il existe une association de *filiation* entre deux personnes, il ne peut exister d'association de *mariage* entre ces deux mêmes personnes

20) Réservation de vols



21) Le magasin d'électroménagers

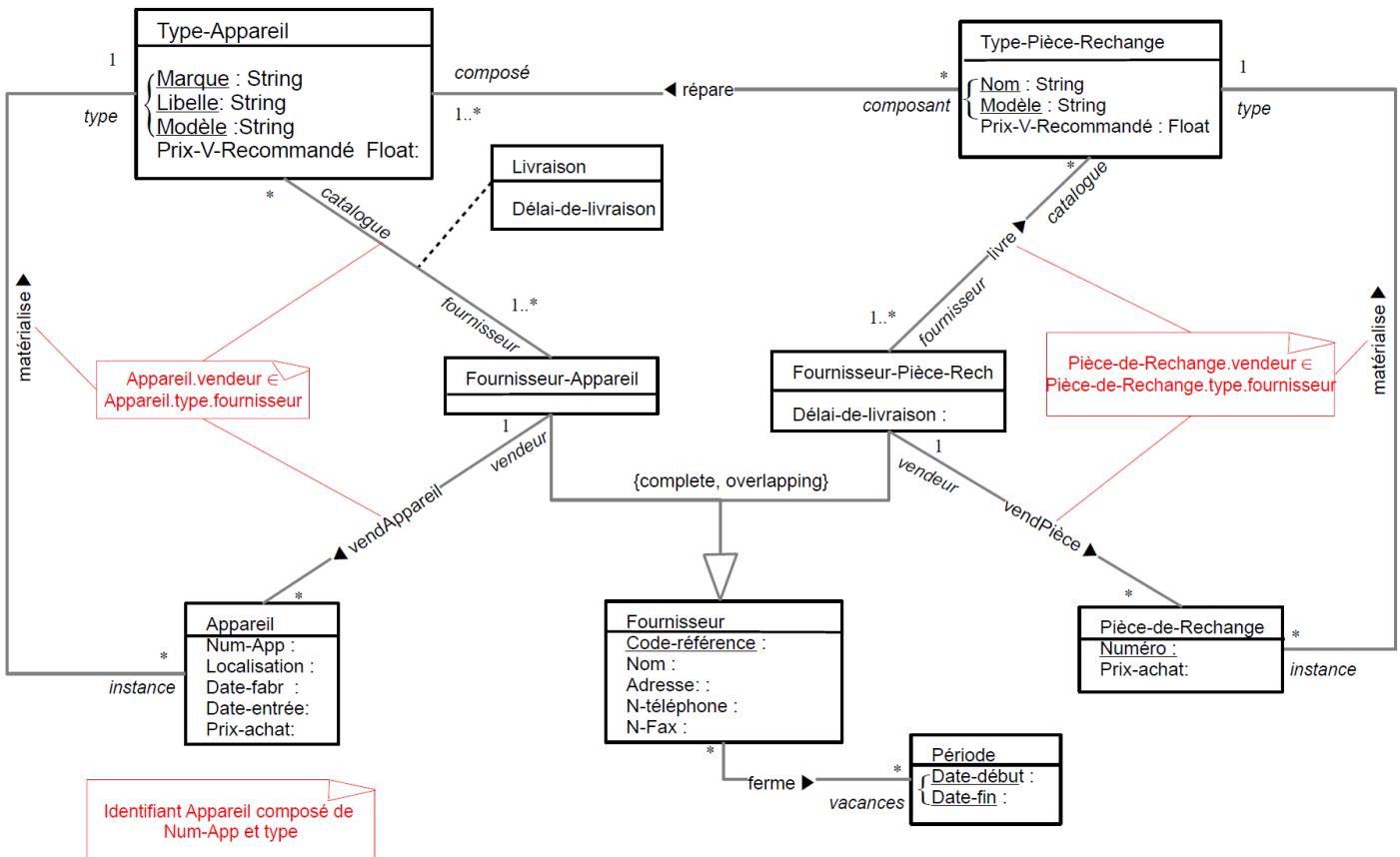
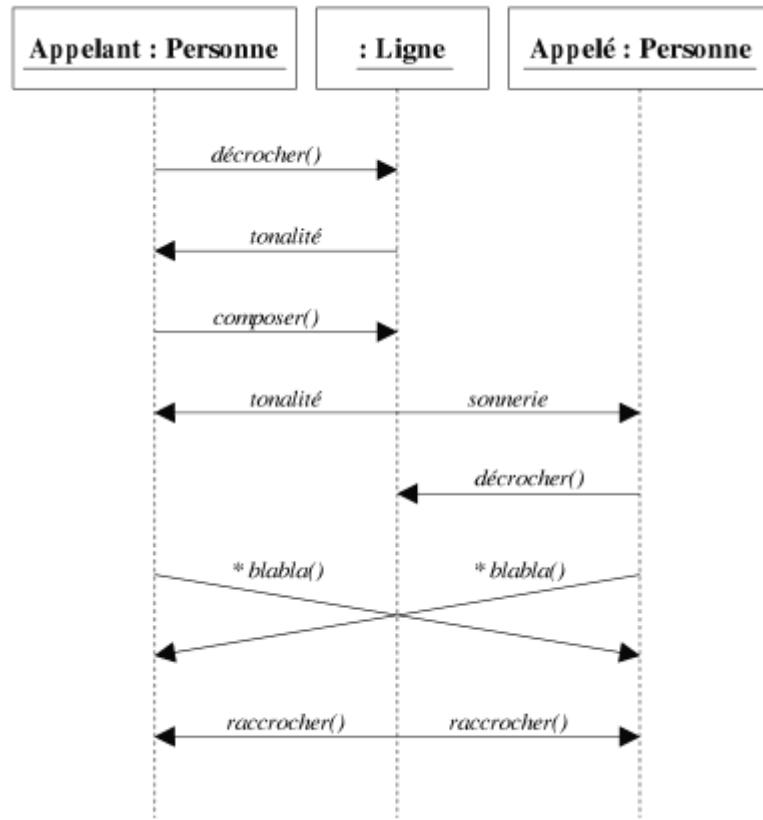


Diagramme de Séquence

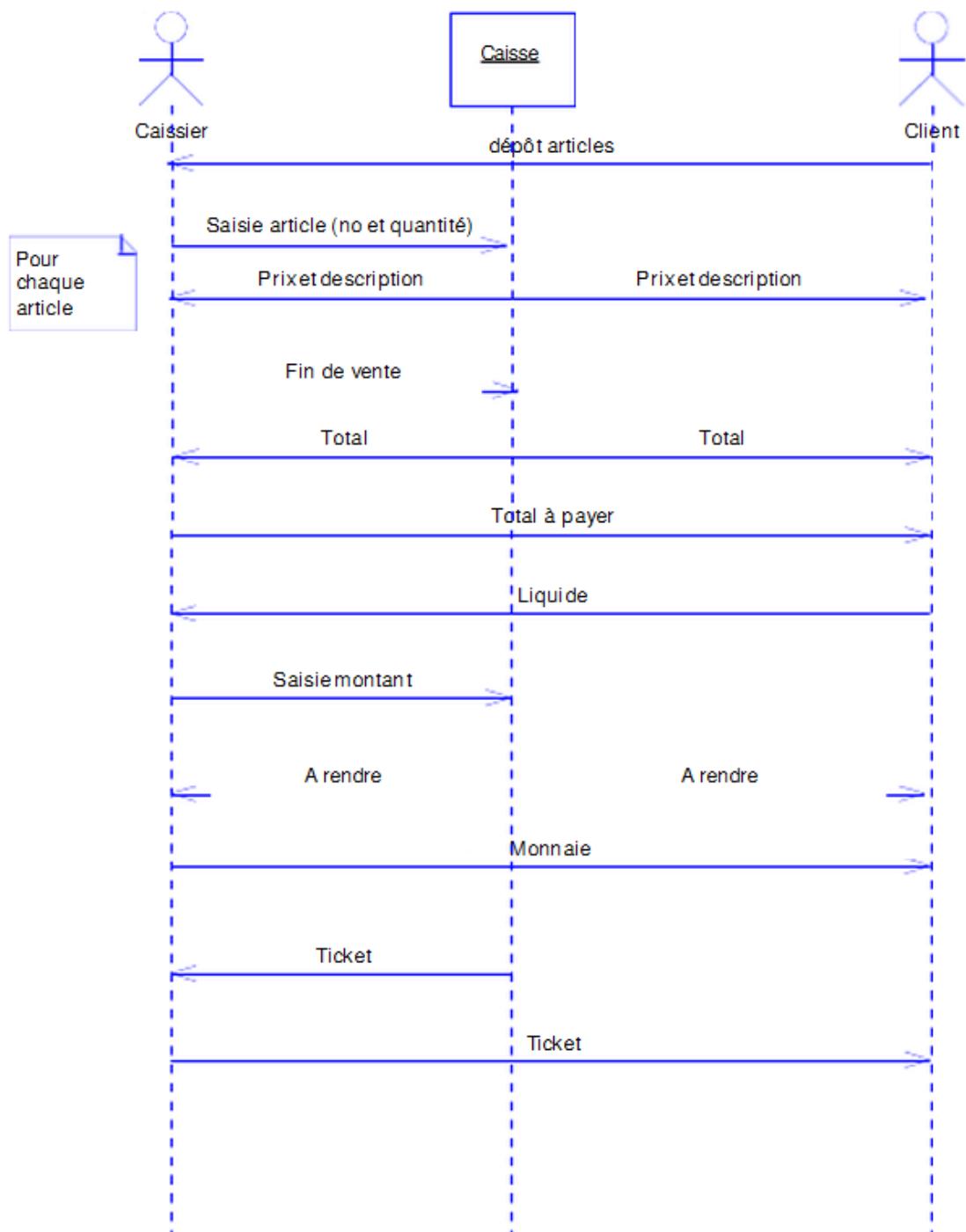
22) La conversation téléphonique



23) L'ascenseur

Corrigé à créer...

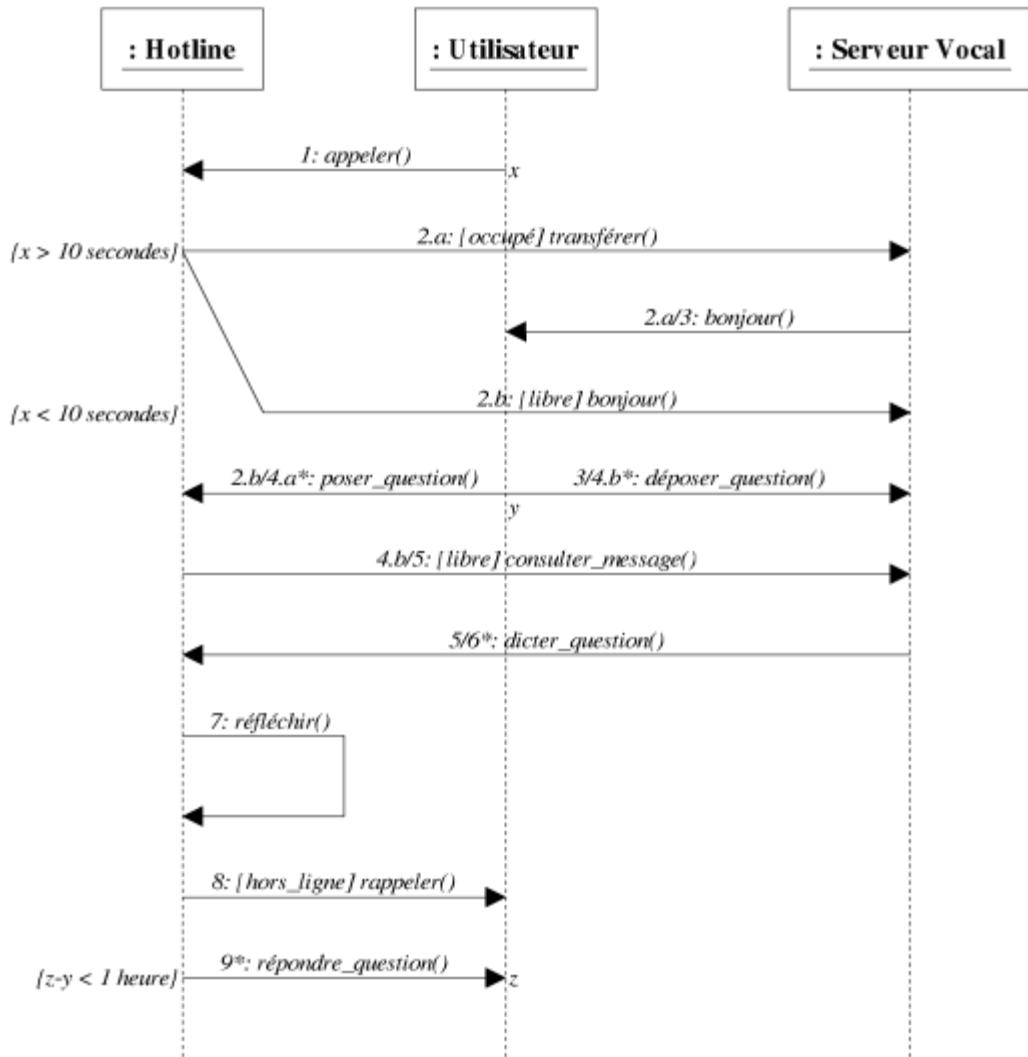
24) La caisse enregistreuse



25) Le retrait d'argent

Corrigé à créer...

26) Assistance téléphonique



27) Projet de recherches en viticulture (diagramme de séquence)

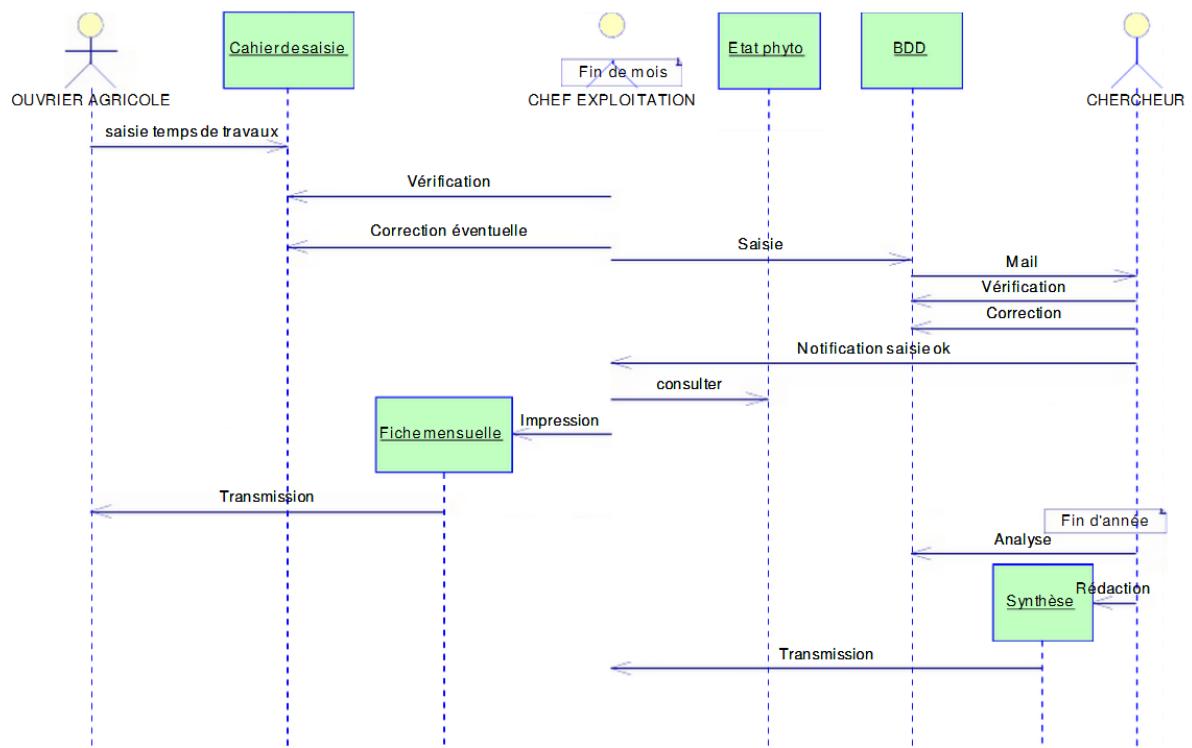
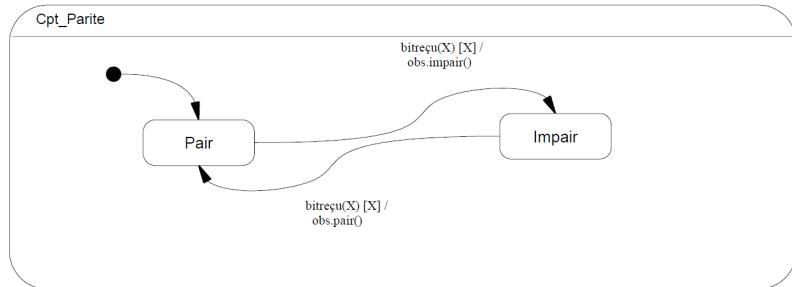


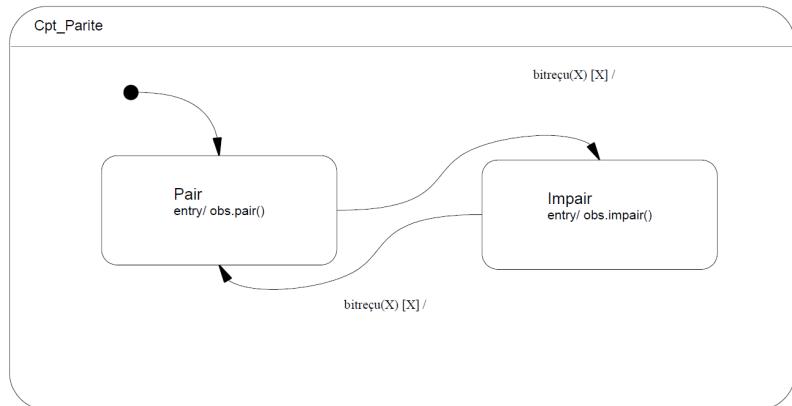
Diagramme d'Etat

28) La parité

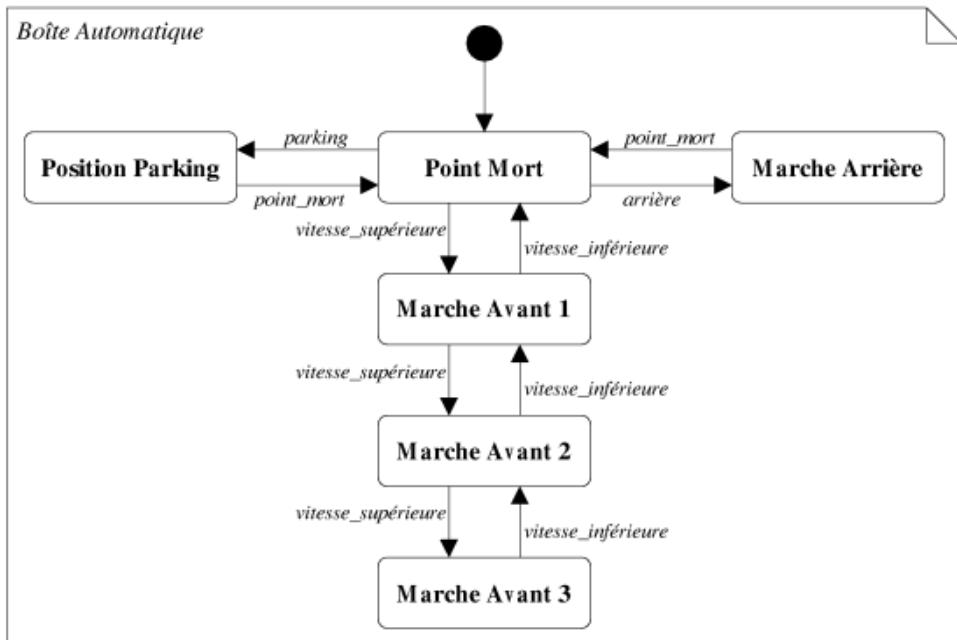
State Diagram : modèle de comportement (avec actions sur les transitions)



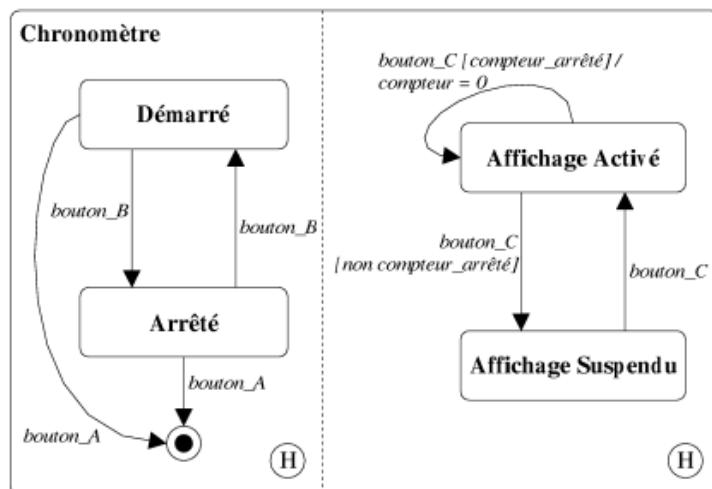
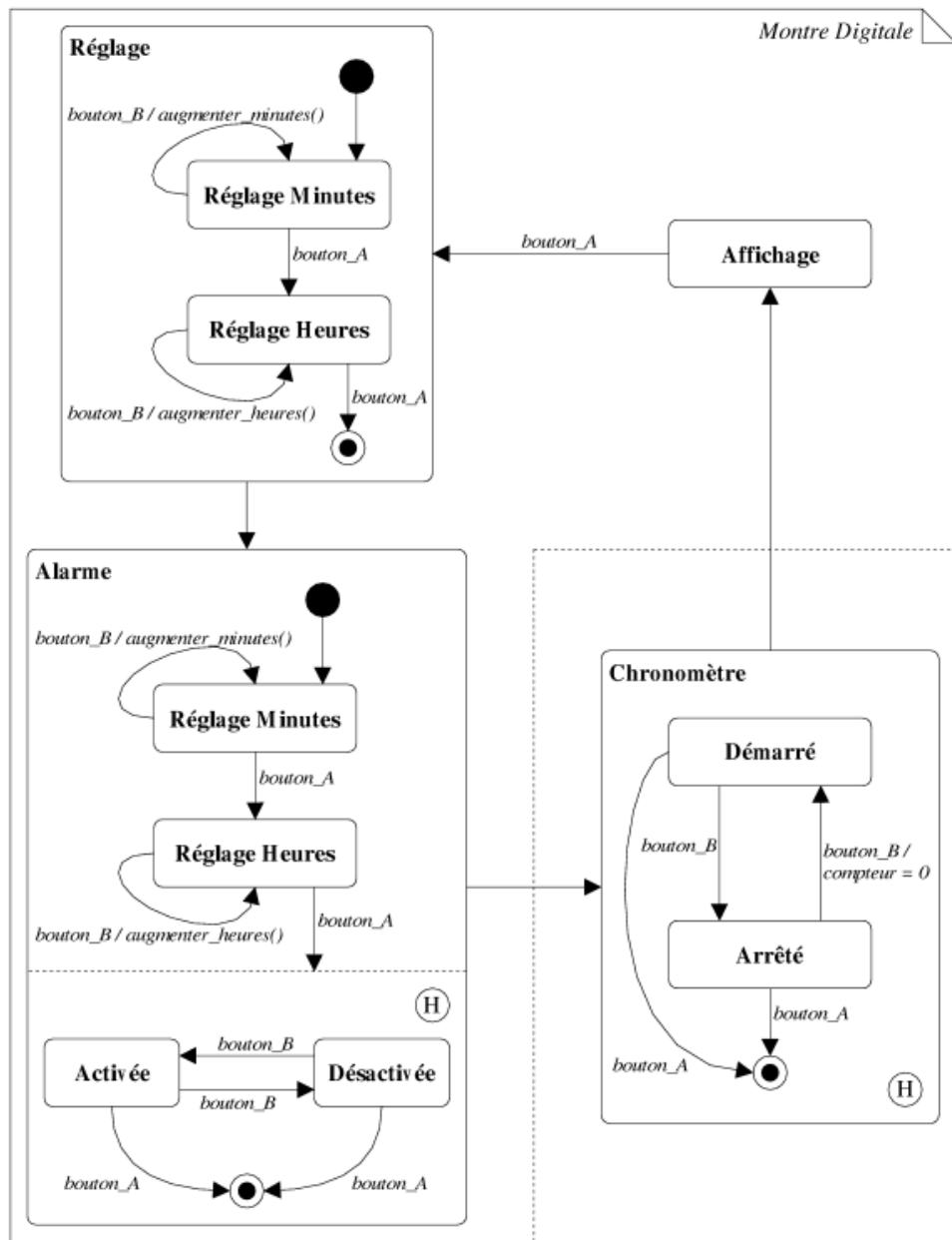
State Diagram : modèle de comportement (sans action sur les transitions, mais avec usage des "Activity Entry")



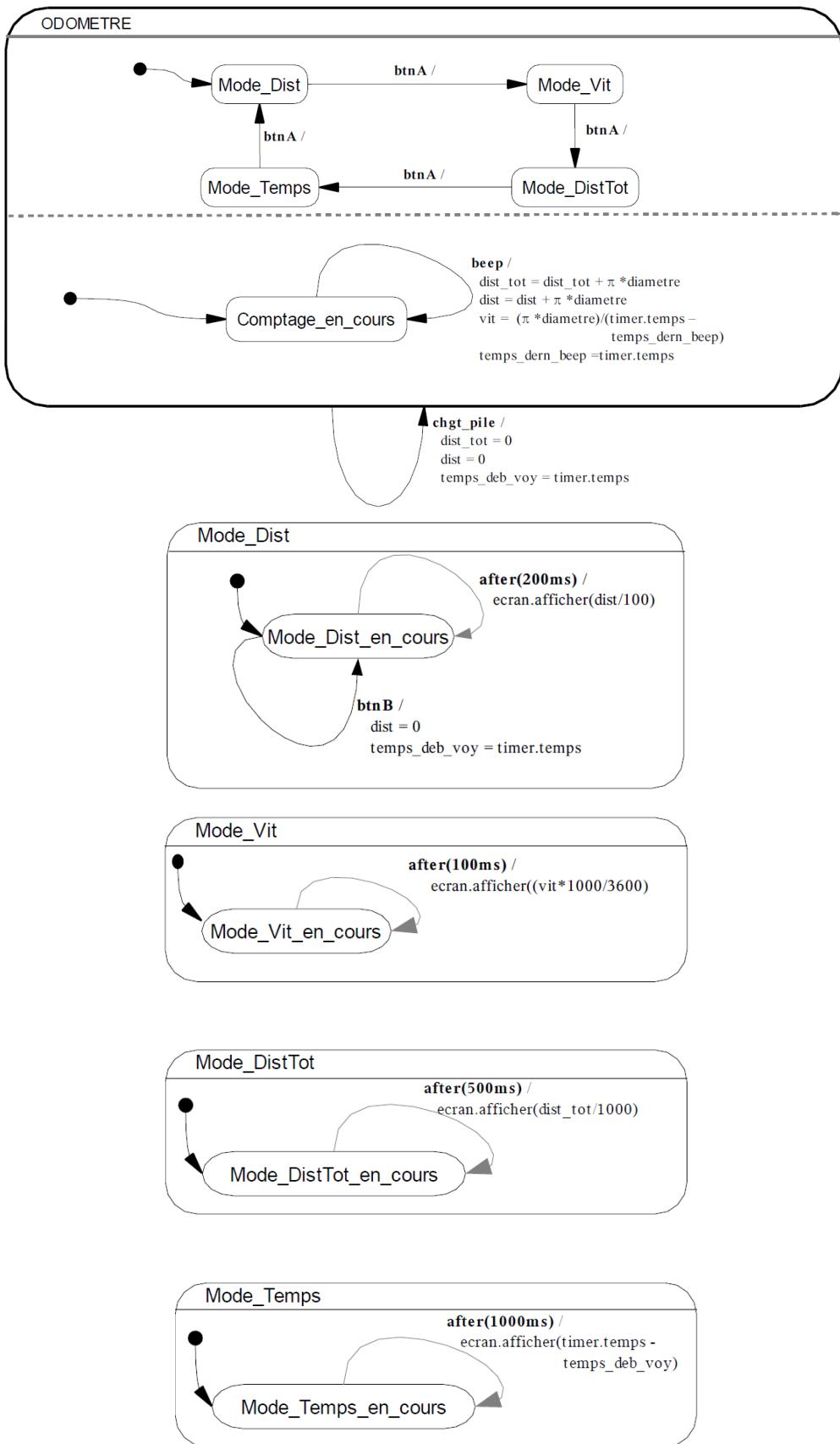
29) La boîte de vitesse automatique



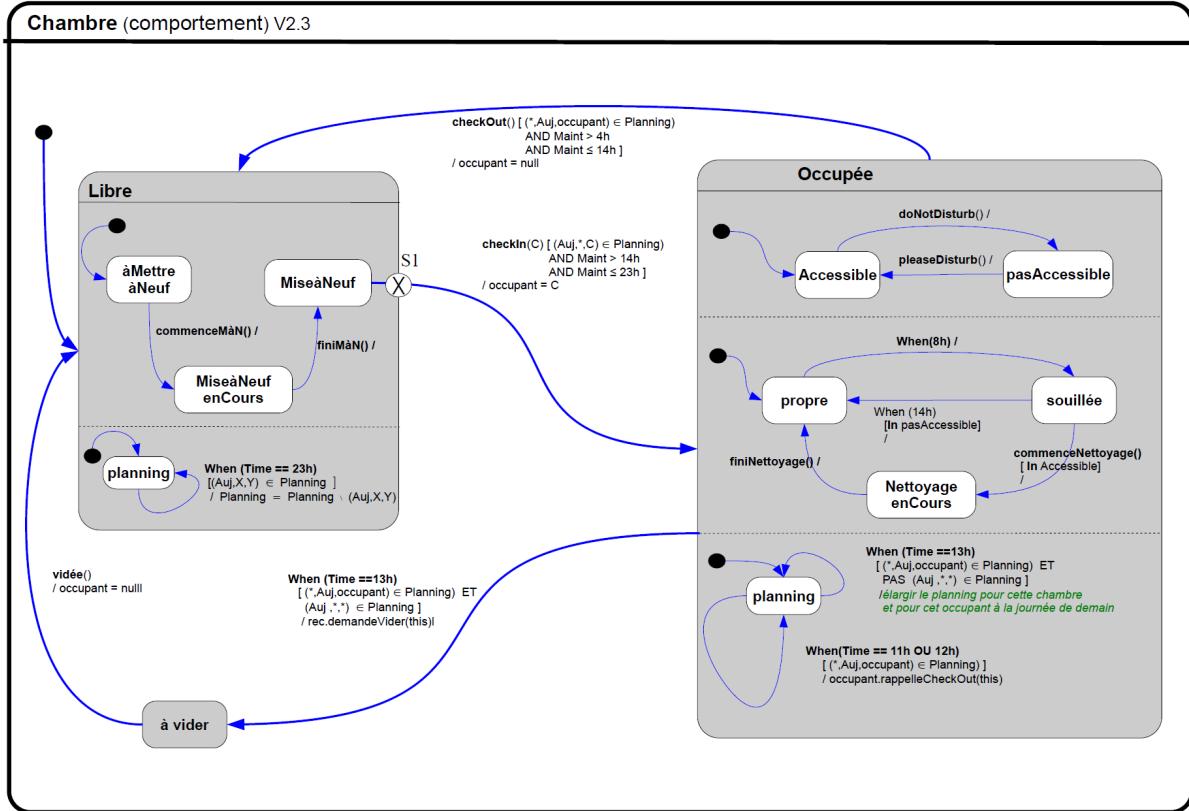
30) La montre digitale



31) L'odomètre



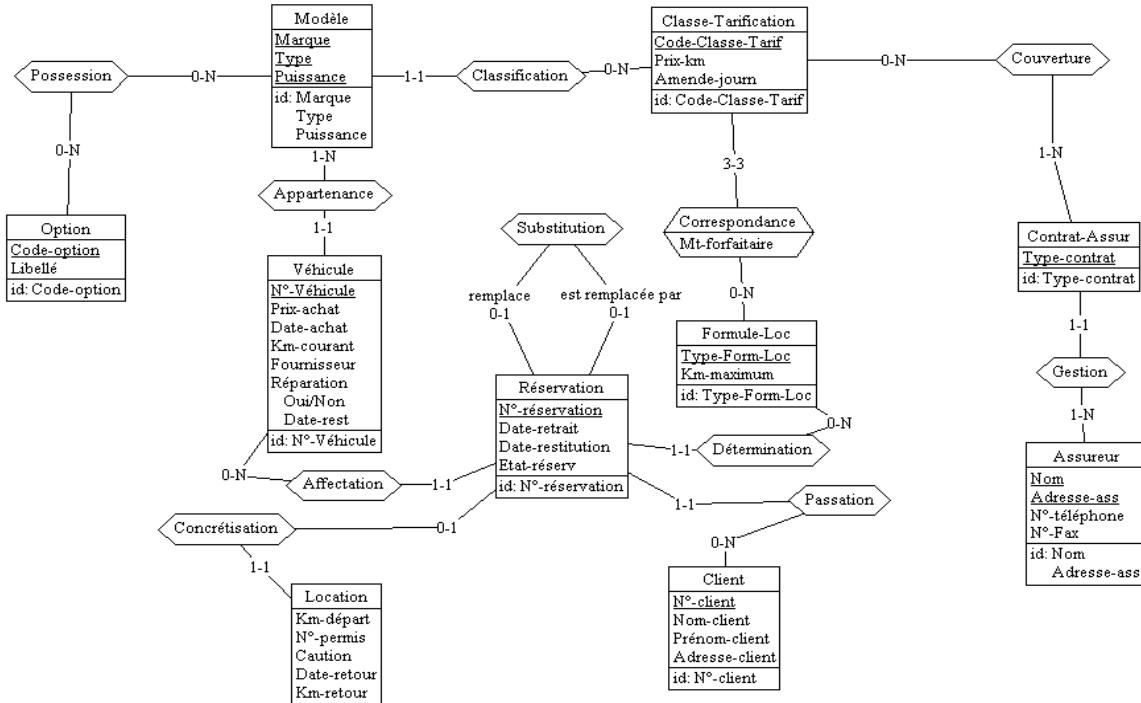
32) Gestion des chambres d'un hôtel



Exercice Intégré

33) Système d'information d'une société de location de voitures

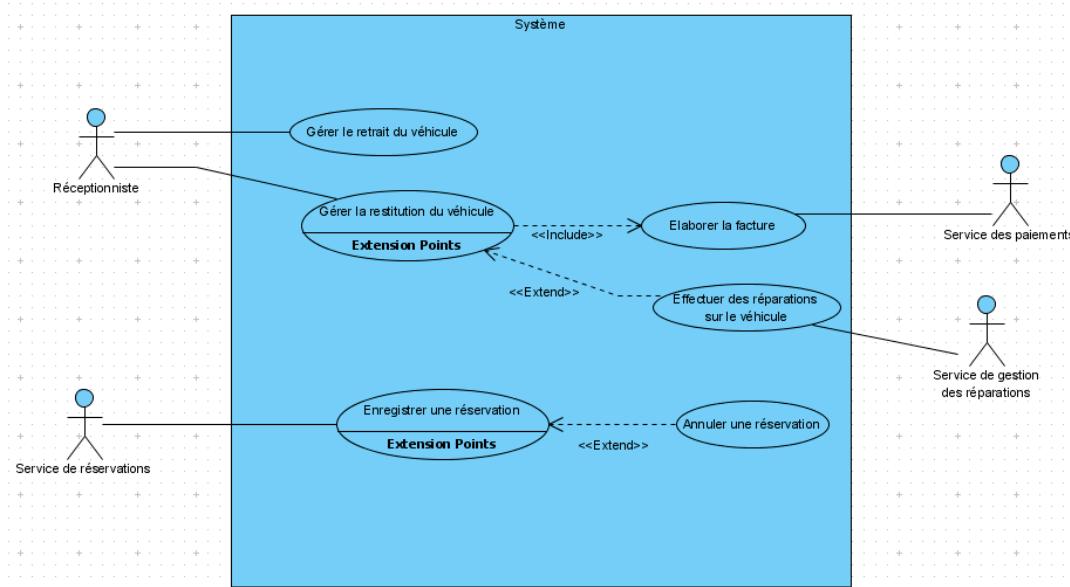
Section 1: Le schéma Entité Association



Section 2 : Documentation du schéma Entité Association

1. Use Case diagram

1.1. Description semi formelle



1.2. Description informelle

1.2.1. Use case : « Enregistrer une réservation »

Brève description : Un utilisateur (le service de réservation) souhaite savoir si un véhicule est disponible et procéder à la réservation.

- *Pré-condition(s) globale(s)* :
 - le système est opérationnel
- *Post-condition(s) globale(s)* :
 - le système est opérationnel

Cas normal

- *Pré-condition(s)* :
 - le client est enregistré dans le système
 - des véhicules sont disponibles
- *Post-condition(s)* :
 - la réservation est enregistrée

SERVICE DE RESERVATION	SYSTEME
1. L'utilisateur manifeste son intention de trouver un véhicule.	2. Le système demande à l'utilisateur de fournir les informations nécessaires pour chercher un véhicule ¹ .
3. L'utilisateur fournit les informations demandées.	4. Le système vérifie les informations. 5. Le système fournit la liste des véhicules disponibles et leurs caractéristiques.
6. L'utilisateur manifeste son intention d'enregistrer une réservation.	7. Le système demande le numéro du client et le numéro du véhicule choisi.
8. L'utilisateur entre les informations demandées.	9. Le système vérifie les informations fournies ² . 10. Le système attribue un numéro par compostage à la réservation.
11. L'utilisateur demande l'impression d'un bon de réservation ³ .	12. Le système effectue la demande.

¹ Les informations nécessaires sont par exemple : les options souhaitées, le modèle du véhicule, la formule de tarification,...

² Le système vérifie que la réservation concerne bien un seul client, un seul mode de location et un seul véhicule

³ Les informations sur le bon de réservation sont : les dates, le véhicule ainsi que le prix indicatif

Cas alternatif : Véhicule non disponible mais réservation enregistrée

- *Pré-condition(s) :*
 - le client est enregistré dans le système
 - des véhicules ne sont pas disponibles
- *Post-condition(s) :*
 - la réservation est enregistrée

SERVICE DE RESERVATION	SYSTEME
(idem point 1 – 3 du UC Normal)	(idem point 2-4 du UC Normal)

7. L'utilisateur fournit les informations demandées.

5. Le système ne fournit pas de liste des véhicules disponibles.
 6. Le système demande à l'utilisateur d'introduire de nouvelles informations nécessaires pour trouver un véhicule disponible¹.

(retour au point 4 du UC Normal)

¹ Les opérations 5,6 et 7 ne pourront être effectuées que deux fois par le système. S'il dépasse ce nombre, l'opération sera automatiquement annulée

(Opération cyclique : après l'opération 7, le système passe soit à l'opération 4 du UC Normal ou à l'opération 5 mais dans ce cas, le système ne pourra y revenir que deux fois)

Cas alternatif : Véhicule non disponible et annulation de l'opération

- *Pré-condition(s) :*
 - le client est enregistré dans le système
 - des véhicules ne sont pas disponibles
- *Post-condition(s) :*
 - la réservation n'est pas enregistrée

SERVICE DE RESERVATION	SYSTEME
(idem point 1 – 3 du UC Normal)	(idem point 2-4 du UC Normal)

7. L'utilisateur n'entre pas de nouvelles informations

5. Le système ne fournit pas de liste des véhicules disponibles.
 6. Le système demande à l'utilisateur d'introduire de nouvelles informations nécessaires pour trouver un véhicule disponible.
 8. L'opération est annulée

Cas alternatif : Véhicule disponible mais client non enregistré

- *Pré-condition(s) :*
 - le client n'est pas enregistré dans le système
 - des véhicules sont disponibles
- *Post-condition(s) :* la réservation est enregistrée

SERVICE DE RESERVATION	SYSTEME
(idem point 1 – 3 – 6 du UC Normal)	(idem point 2 - 4 – 5 du UC Normal) 7. Le système demande les informations concernant l'enregistrement du client ¹ .
8. L'utilisateur fournit les informations demandées.	9. Le système vérifie les informations et attribue au client un numéro par compostage. 10. Le système demande le numéro du véhicule choisi.
11. L'utilisateur fournit l'information demandée.	

(retour au point 9 du UC Normal)

¹ Les informations concernant l'enregistrement du client sont : son nom, son prénom, son adresse.

Cas alternatif : Véhicule non disponible et client non enregistré

- *Pré-condition(s) :*
 - le client n'est pas enregistré dans le système
 - des véhicules ne sont pas disponibles
- *Post-condition(s) :* la réservation est enregistrée

SERVICE DE RESERVATION	SYSTEME
(idem point 1 – 3 du UC Normal)	(idem point 2-4 du UC Normal)
7. L'utilisateur fournit les informations demandées.	5. Le système ne fournit pas de liste des véhicules disponibles. 6. Le système demande à l'utilisateur d'introduire de nouvelles informations nécessaires pour trouver un véhicule disponible ¹ .
10. L'utilisateur manifeste son intention d'enregistrer une réservation.	8. Le système vérifie les informations. 9. Le système fournit la liste des véhicules disponibles et leurs caractéristiques.

(retour au point 7 du UC Alternatif : Véhicule disponible mais client non enregistré)

¹ Les opérations 5,6 et 7 ne pourront être effectuées que deux fois par le système. S'il dépasse ce nombre, l'opération sera automatiquement annulée

(Opération cyclique : après l'opération 7, le système passe soit à l'opération 4 du UC Normal ou à l'opération 5 mais dans ce cas, le système ne pourra y revenir que deux fois)

1.2.2. Use case : « Annuler une réservation »

Brève description : Un client a, à tout moment, la possibilité d'annuler sa réservation.

Cas normal

- *Pré-condition(s) :*
 - le système est opérationnel
 - la réservation a été enregistrée
- *Post-condition(s) :*

- le système est opérationnel
- la réservation est annulée

SERVICE DE RESERVATION	SYSTEME
1. L'utilisateur manifeste son intention d'annuler la réservation.	2. Le système lui demande de fournir le numéro de la réservation ainsi que le numéro du client.
3. L'utilisateur fournit l'information demandée.	4. Le système vérifie la validité des données et annule la réservation. 5. Le système émet, s'il y a lieu ¹ , un message à un autre utilisateur (le service des paiements) qu'une pénalisation financière devra être payée ² .

¹ Une pénalisation financière devra être payée si l'annulation de la réservation a lieu moins de deux jours avant la date de retrait prévue.

² La pénalisation financière est égale à la caution qui aurait été demandée et sera traitée par le service des paiements en fin de semaine.

1.2.3. Use case : « Gérer le retrait du véhicule »

Brève description : Un utilisateur (le réceptionniste) vérifie que le véhicule réservé est disponible et procède à l'enregistrement de la location.

- *Pré-condition(s) globale(s)* : le système est opérationnel
- *Post-condition(s) globale(s)* : le système est opérationnel

Cas normal :

- *Pré-condition(s)* : le véhicule réservé est disponible
- *Post-condition(s)* : la location est enregistrée

RECEPTIONNISTE	SYSTEME
<p>1. L'utilisateur constate de visu que le véhicule correspondant au bon de réservation est effectivement disponible.</p> <p>2. L'utilisateur manifeste son intention d'enregistrer la location.</p> <p>4. L'utilisateur lui fournit les informations demandées.</p> <p>6. L'utilisateur demande au système d'imprimer deux contrats de location.</p> <p>8. L'utilisateur fait signer un des contrats par le client qu'il archivera en fin de journée.</p> <p>9. L'utilisateur remet au client l'autre contrat ainsi que les clefs du véhicule.</p>	<p>3. Le système lui demande de fournir les informations nécessaires à l'enregistrement¹.</p> <p>5. Le système vérifie les informations et enregistre la location.</p> <p>7. Le système effectue la demande.</p>

¹ Les informations nécessaires à l'enregistrement de la location sont : le kilométrage du véhicule, le numéro de permis du conducteur et si la caution a été payée.

Cas alternatif : véhicule réservé n'est pas disponible mais la location est enregistrée

- *Pré-condition(s)* : le véhicule réservé n'est pas disponible
- *Post-condition(s)* : la location est enregistrée

RECEPTIONNISTE	SYSTEME
<p>1. L'utilisateur constate de visu que le véhicule correspondant au bon de réservation n'est pas disponible.</p> <p>2. L'utilisateur manifeste son intention d'enregistrer que la réservation a été annulée.</p> <p>4. L'utilisateur fournit les informations demandées.</p>	<p>3. Le système demande le numéro de réservation et le numéro du client</p> <p>5. Le système vérifie les informations.</p> <p>6. Le système enregistre que le client bénéficiera d'un dédommagement</p>

(retour au UC « Enregistrer une réservation ». On pourra soit avoir le UC Normal soit le UC Alternatif « véhicule non disponible mais réservation enregistrée»
 (retour au point 1 du UC Normal)

Cas alternatif : véhicule réservé n'est pas disponible et la location n'est pas enregistrée

- *Pré-condition(s)* : le véhicule réservé n'est pas disponible
- *Post-condition(s)* : la location n'est pas enregistrée

RECEPTIONNISTE	SYSTEME
(idem point 1 – 2 – 4 du UC Alternatif : véhicule réservé n'est pas disponible mais la location est enregistrée)	(idem point 3 – 5 – 6 du UC alternatif : véhicule réservé n'est pas disponible mais la location est enregistrée)
(retour au UC « Enregistrer une réservation ». On pourra soit avoir le UC Normal soit le UC Alternatif « véhicule non disponible mais réservation enregistrée»)	
7. L'utilisateur constate de visu que le véhicule correspondant au bon de réservation n'est pas disponible. 8. Annulation de l'opération. (Hypothèse)	

Cas alternatif : véhicule réservé n'est pas disponible et aucune autre réservation n'est effectuée

- *Pré-condition(s)* : le véhicule réservé n'est pas disponible
- *Post-condition(s)* : la location n'est pas enregistrée

RECEPTIONNISTE	SYSTEME
(idem point 1 – 2 – 4 du UC Alternatif : véhicule réservé n'est pas disponible mais la location est enregistrée)	(idem point 3 – 5 – 6 du UC alternatif : véhicule réservé n'est pas disponible mais la location est enregistrée)
(retour au UC « Enregistrer une réservation ». On a le UC Alternatif « véhicule non disponible et annulation de l'opération»)	

1.2.4. Use case : « Gérer la restitution du véhicule »

Brève description : Un utilisateur (le réceptionniste) gère la restitution du véhicule et clôture la location.

Cas normal

- *Pré-condition(s)* :
 - le système est opérationnel
 - le véhicule est restitué
- *Post-condition(s)* :
 - le système est opérationnel
 - la location est clôturée

RECEPTIONNISTE		SYSTEME (includes <i>Elaborer la facture</i>)
1. L'utilisateur vérifie de visu l'état du véhicule. 2. L'utilisateur manifeste son intention de clôturer la location. 4. L'utilisateur entre les informations demandées. 7. L'utilisateur demande au système d'imprimer un formulaire de clôture de location ² .		3. Le système lui demande les informations nécessaires à la clôture de la location ¹ . 5. Le système vérifie les informations. 6. Le système enregistre la clôture de la location. 8. Le système effectue la demande.

¹ Informations nécessaires à la clôture la location sont : le kilométrage du véhicule lors de la restitution et la date effective de restitution.

² Informations sur le formulaire de clôture sont : identification du véhicule, kilométrage courant et date effective de restitution.

1.2.5. Use case : « Elaborer la facture »

Brève description : Sur base du formulaire de clôture de location, l'utilisateur (le service des paiements) élabore la facture.

Cas normal

- *Pré-condition(s)* :
 - le système est opérationnel
 - le véhicule a été restitué
- *Post-condition(s)* :
 - le système est opérationnel
 - la facture est élaborée

SERVICE DES PAIEMENTS	SYSTEME
<p>1. L'utilisateur manifeste son intention d'élaborer la facture.</p> <p>3. L'utilisateur fournit l'information demandée.</p> <p>6. L'utilisateur demande au système d'imprimer la facture en deux exemplaires.</p> <p>8. L'utilisateur remet au client un exemplaire² et archivera l'autre en fin de semaine.</p>	<p>2. Le système lui demande de fournir les informations présentes sur le formulaire de clôture de réservation ainsi que le numéro du client.</p> <p>4. Le système vérifie la validité des données.</p> <p>5. Le système élabore la facture en tenant compte de différents paramètres¹.</p> <p>7. Le système s'exécute.</p>

¹ Les différents paramètres sont :

- si la caution a été payée
- le montant forfaitaire
- dédommagement éventuel
- des suppléments éventuels.

Aucune réduction n'est prévue

² Les factures non payées immédiatement sont payables dans les 15 jours

1.2.6. Use case : « Effectuer des réparation sur le véhicule »

Brève description : Lors de la restitution du véhicule, l'utilisateur (réceptionniste) vérifie l'état de la voiture et émet, s'il y a lieu, un message à un autre utilisateur (service de gestion des réparations) qui effectuera les réparations sur le véhicule.

Nous ne développerons pas davantage ce Use Case.

2. Diagramme d'activité

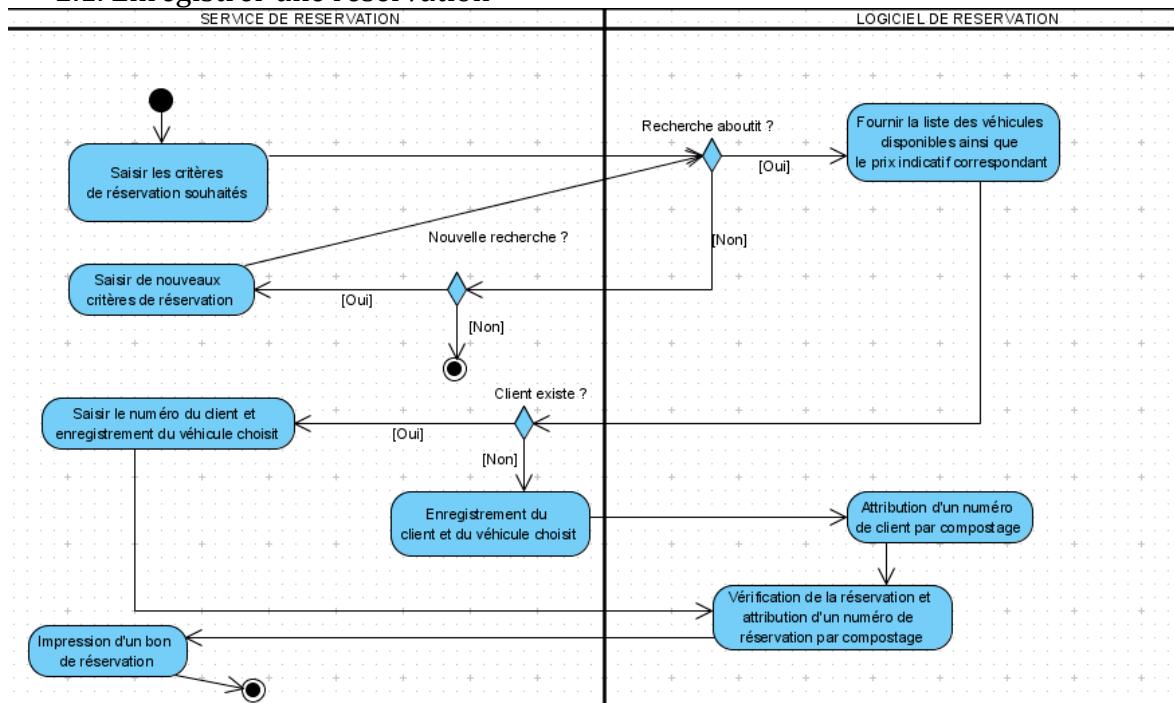
Deux diagrammes d'activité ont été réalisés pour les Use Case suivants :

- *Enregistrer une réservation*
- *Gérer le retrait du véhicule*

En effet, le diagramme d'activité permet de bien visualiser toutes les différentes alternatives qui se présentent dans des Use Case.

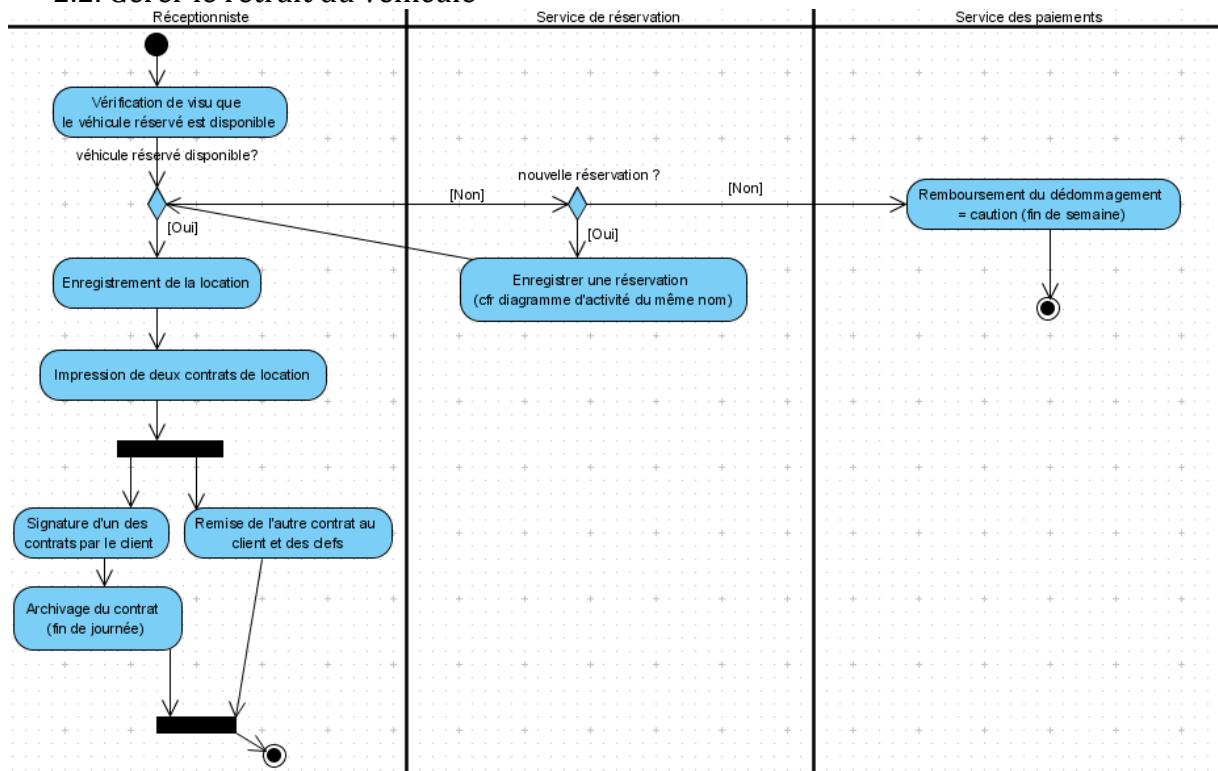
Il n'aurait pas été pertinent d'en faire pour les autres Use Case au vu du peu d'alternatives qu'ils proposent.

2.1. Enregistrer une réservation



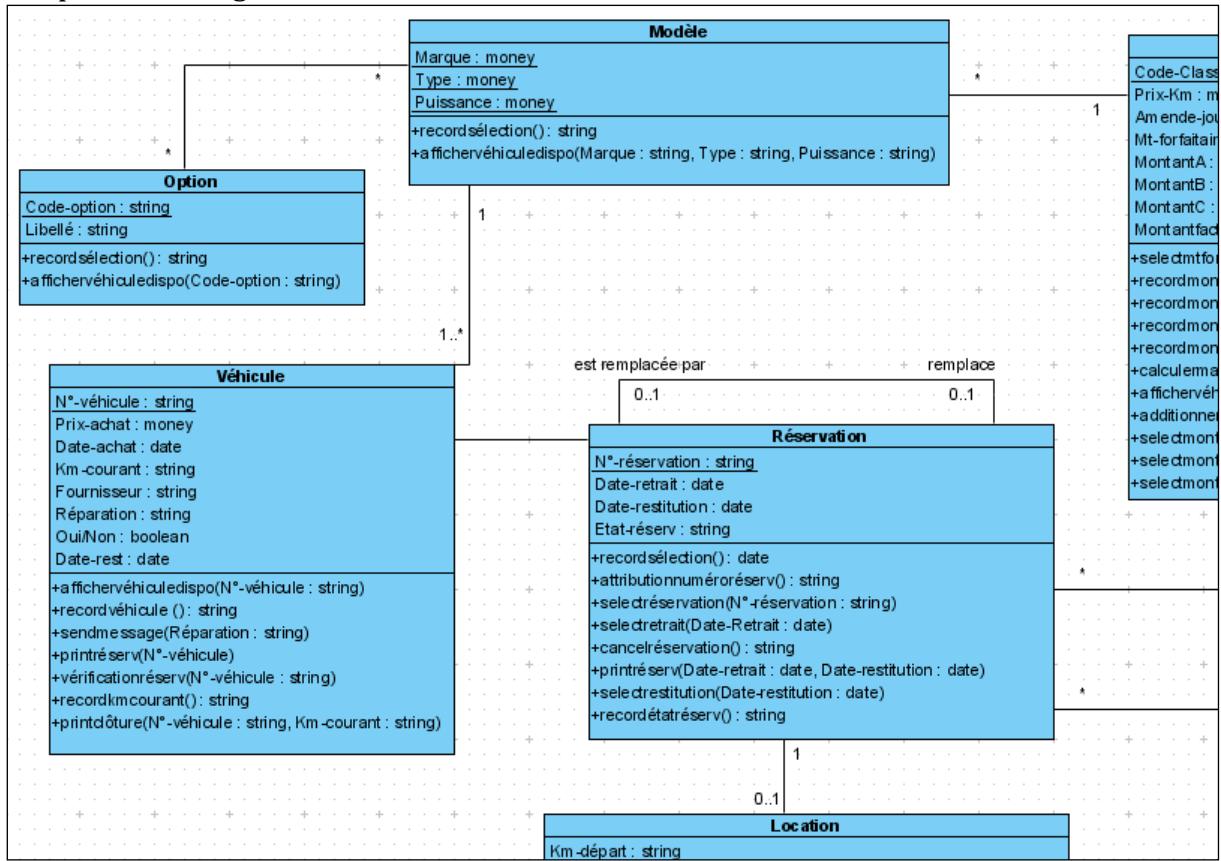
Si après deux fois, l'opération « nouvelle recherche » n'aboutit toujours pas, alors le client arrête les procédures de réservation. (Hypothèse)

2.2. Gérer le retrait du véhicule

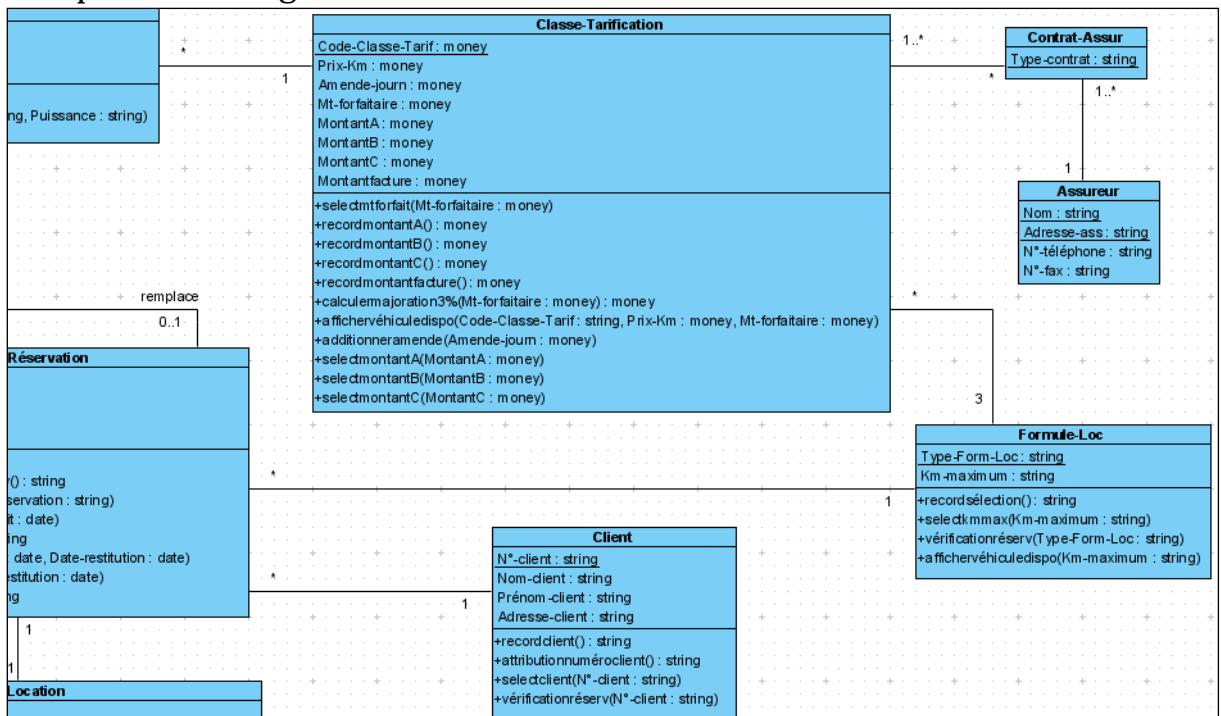


3. Diagramme de classe

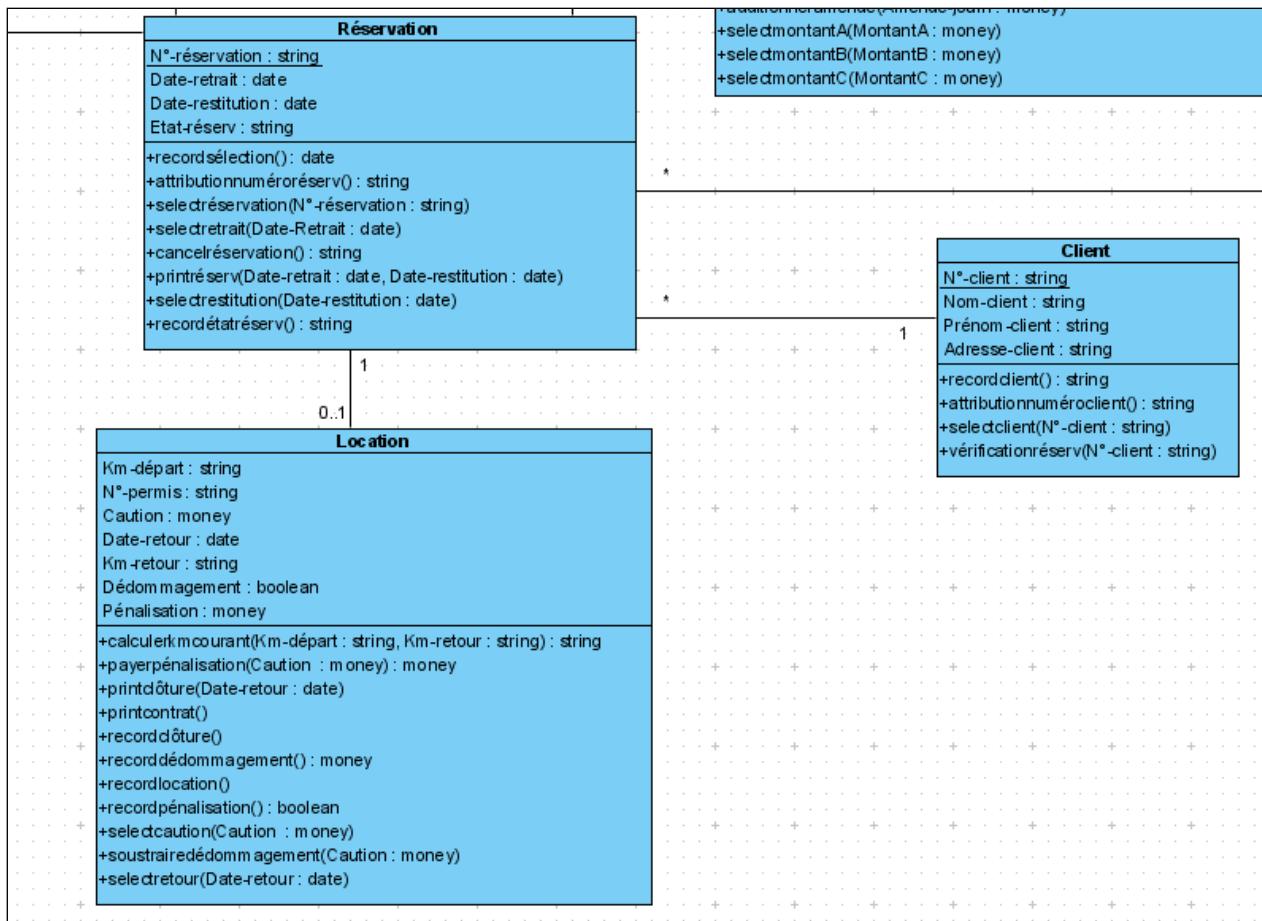
1^{ère} partie du diagramme de classe :



2^{ème} partie du diagramme de classe

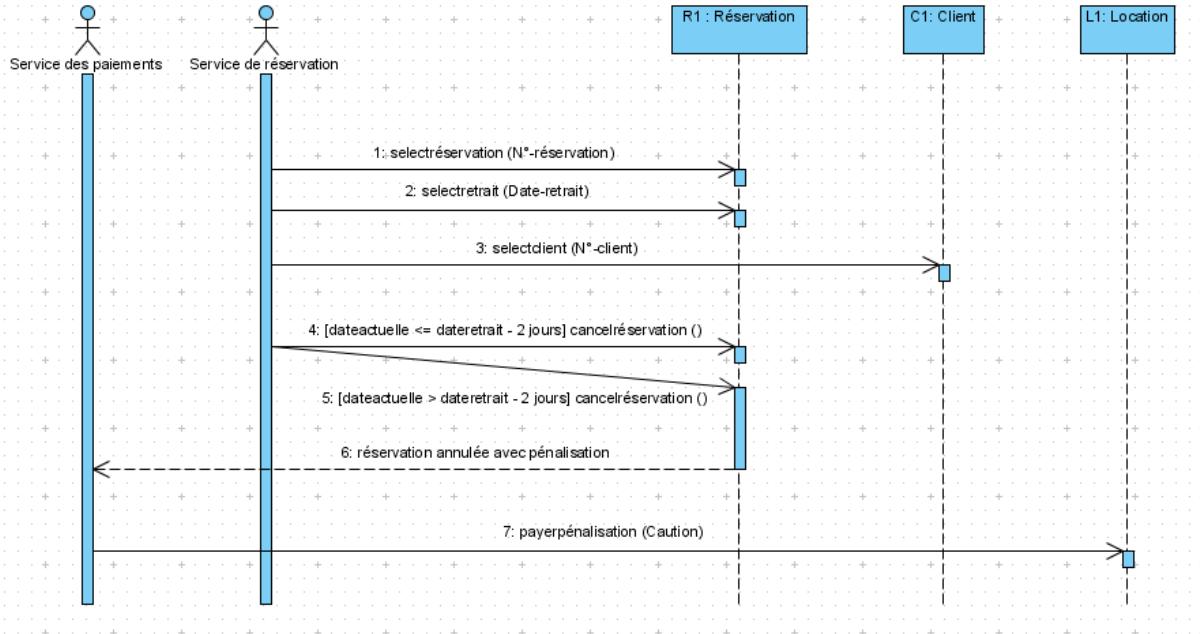


3^{ème} Partie du diagramme de classe

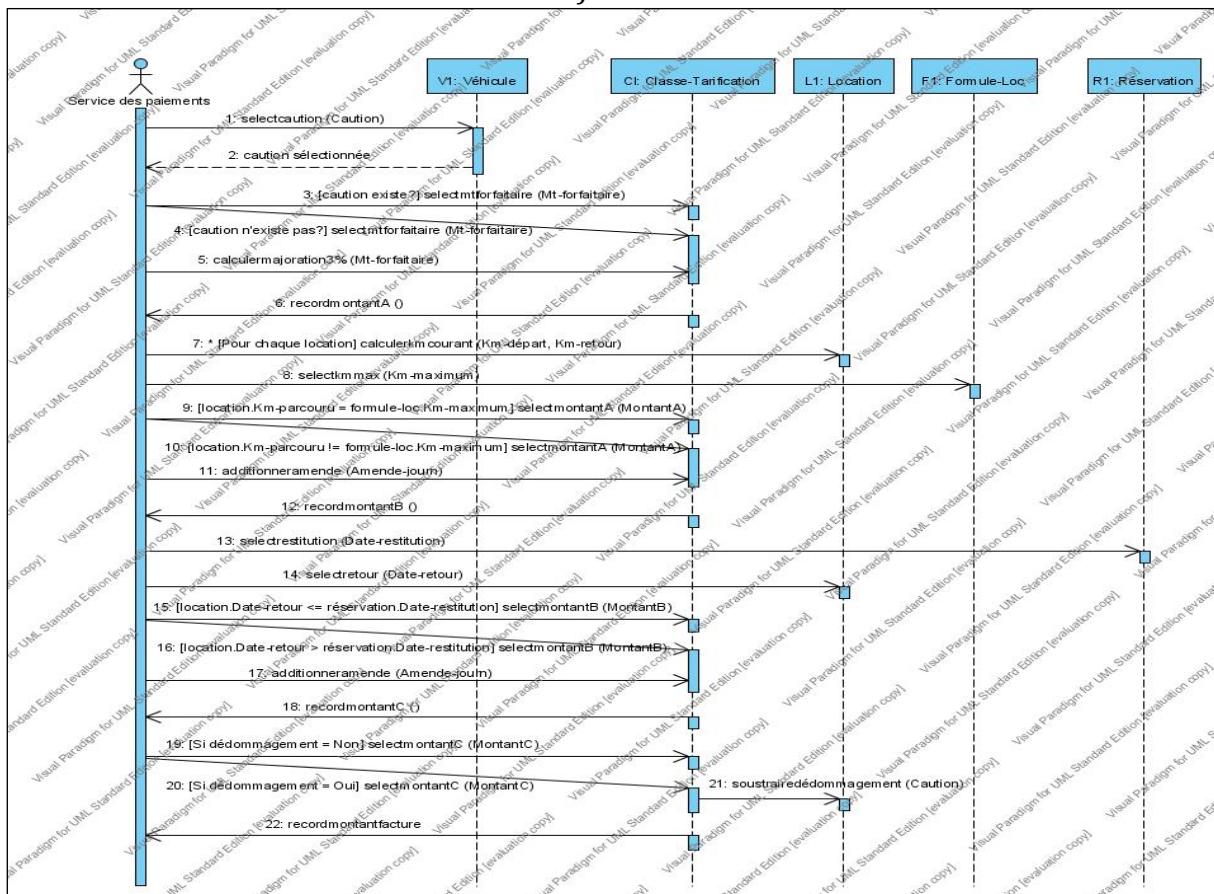


4. Diagramme de séquence

4.1. Pour le Use Case : Annuler une réservation



4.2. Pour le Use Case : Elaborer une facture



5. Diagramme d'état

Un diagramme d'état a été réalisé pour l'opération “recordétatréserv” de la classe Réservation. Cette opération va, à tout moment dans le système, enregistrer l'état de la

réservation (voir attribut : Etat-réserv de la classe Réservation). Cet attribut peut prendre quatre états différents selon certaines conditions. Il paraît pertinent de les visualiser par un graphique.

