



Compte-rendu de projet

Spécification des Systèmes Dynamiques

Samy ABDOUCHE
11606892

Table des matières

I	Brève introduction	3
II	Les fonctionnalités internes au système à développer	3
III	Les acteurs du système	4
IV	Le diagramme de cas d'utilisation associé	6
V	Les fiches détaillées des Use-Case	7
V.1	UC01 : Passer une commande	7
V.2	UC02 : Se connecter	8
V.3	UC03 : S'inscrire	8
V.4	UC04 : Envoyer demande de récupération produit à client physique . . .	9
V.5	UC05 : Recevoir commande	10
V.6	UC06 : Accepter demande de récupération d'un produit	10
V.7	UC07 : Déposer un produit dans une boîte de commande	11
V.8	UC08 : Valider un produit	12
V.9	UC09 : Passer en caisse	14
V.10	UC10 : Vérifier la commande d'un client	15
V.11	UC11 : Prendre en charge la récupération d'un produit	15
V.12	UC12 : Appliquer des réductions	16
V.13	UC13 : Chercher les caddies les plus proches	17
VI	Diagramme de séquence	18
VII	Propriétés de l'interaction	19
VIII	Description d'un observateur d'état sur le nombre de commandes d'un client	19

Section I

Brève introduction

Ce rapport se concentrera sur les parties d'analyse des besoins et de spécification. Carrérond souhaite réduire les coûts de façon innovante, et nous confie le développement du système informatique gérant sa future application. Ainsi, ces étapes seront réalisées, en respectant scrupuleusement le cahier des charges qui nous a été donné. Ce dernier contenant quelques zones d'ombre (et ne disposant pas d'un client avec qui échanger), de telles prises de décision seront explicitées, de façon à ce que le lecteur comprenne bien qu'il s'agit d'une prise d'initiatives.

Section II

Les fonctionnalités internes au système à développer

Le système à développer devra comporter certaines fonctionnalités natives, que voici :

- Le système sera doté d'un système de timers. Ces timers seront utilisés lors de l'envoi de demandes de récupération à des clients physiques, afin de déterminer un temps paramétrable au cours duquel les confirmations peuvent s'effectuer. Ils seront également utilisés à la confirmation, afin de déterminer un temps paramétrable au cours duquel la récupération d'un produit peut s'effectuer.
- Le système sera également doté d'observateurs d'état, qui lui permettront d'être averti dans le cas de la modification d'une donnée utilisée de façon globale. Typiquement,

ces observateurs d'état l'informeront des changements de position d'un caddie, par exemple, ou de nombre de produits dont s'occupe un client physique donné.

Section III

Les acteurs du système

Cette section est dédiée à l'apport de détails concernant les acteurs primaires et secondaires de notre système.

Acteur	Rôle	Description
Client internet	Primaire	Client commandant à partir d'internet
Client physique	Primaire	Client faisant ses courses à Carrérond
Site internet	Secondaire	Site internet de commande
Employé de magasin	Primaire	Employé de Carrérond
Tablette	Secondaire	Tablette posée sur chaque caddie
Ecran de scan	Secondaire	Ecran de scan
Serveur de bases de données	Secondaire	Serveur de Bases de Données
Boîte de commande	Secondaire	Boîte de commande
Caisse automatique	Secondaire	Caisse automatique

Décrivons désormais les objectifs de nos acteurs. Dans un souci de lisibilité, nous ne présenterons pas lesdits objectifs dans un tableau, mais sous forme d'une simple liste, ce qui nous permettra de gagner en clarté, tout en apportant un nombre suffisant de détails :

- Client internet : son objectif est simple, il veut pouvoir passer une commande en ligne, sur le site internet de Carrérond, et la récupérer par la suite.
- Client physique : le client physique est celui qui se déplace à Carrérond pour faire ses courses, ce qui reste son objectif premier. Cependant, il peut également, s'il l'accepte, s'occuper de la récupération d'un produit pour une commande internet

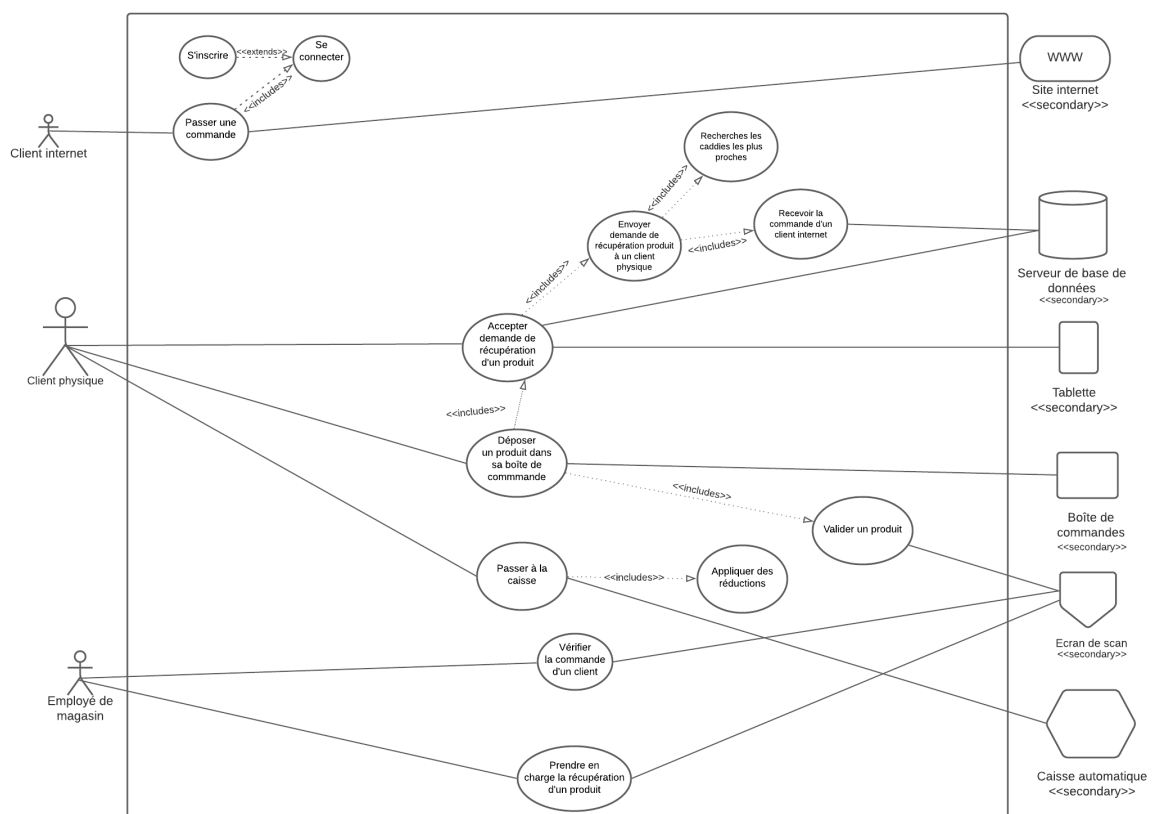
d'un autre client Internet. Dans ce cas, lorsque ce client Physique se rendra à la caisse, il bénéficiera de réductions proportionnelles au nombre de produits qu'il a récupérés.

- Site internet : le site internet doit pouvoir offrir une interface claire au client internet. C'est de par son truchement que le client sélectionnera des articles et passera une commande.
- Employé de magasin : l'employé de magasin, outre ses missions basiques d'achalandage, doit superviser le bon déroulement d'une commande Internet. Il se charge donc de vérifier, une fois que les produits figurant sur la commande d'un client Internet ont bien été récupérés, si des produits en trop figurent. Il peut également se charger lui-même de la récupération d'un produit, si un client physique ayant accepté la récupération ne s'en charge pas en un certain temps, ou si aucun client n'accepte de prendre en charge la récupération d'un produit.
- Tablette : tablette accrochée sur un caddie. Le client physique peut l'activer ou la désactiver. Si elle est activée, le client physique peut recevoir des demandes de récupération de produit d'une commande Internet, et les accepter. La tablette affiche également un message lorsque le client physique n'a pas récupéré le produit à temps. Elle est dotée d'un module GPS, ce qui peut transmettre sa position à un appareil tiers.
- Ecran de scan : un écran de scan se trouvera à proximité d'une boîte de commande. Il s'agit d'un écran tactile doté d'un appareil de scan. Il sera utilisé pour scanner et valider des produits.
- Serveur de bases de données : notre système utilisera un serveur de bases de données, qui servira à consigner plusieurs informations relatives aux commandes internet, ainsi qu'à leur prise en charge par des clients physiques ou des employés du magasin.
- Boîte de commande : il s'agit d'une boîte dans laquelle seront déposés des produits d'un client internet. Ces produits auront été récupérés par un client physique, ou un employé du magasin. On imagine que chaque boîte correspond à une commande précise.

- Caisse automatique : la caisse automatique sera utilisée par un client physique ayant fini de faire ses emplettes. Grâce au numéro de caddie fourni par le client physique, elle récupérera les informations requises pour appliquer de potentielles réductions au total du client physique.

Section IV

Le diagramme de cas d'utilisation associé



Et voici le diagramme de cas d'utilisation associé. Je vous conseille vivement de copieusement zoomer, pour pouvoir correctement voir le diagramme.

Dans un souci de lisibilité, j'ai cherché à fournir un diagramme aux liens qui s'entrecroisent le moins, tout en gardant les informations essentielles. Tout sera consigné avec précision dans les fiches détaillées.

Section V

Les fiches détaillées des Use-Case

V.1 UC01 : Passer une commande

UseCase 01

Nom : Passer une commande

Acteur principal : Client internet

Acteur secondaire : Site internet

Scénario nominal :

1. Le client se rend sur le site internet de Carrérond
2. Le client se connecte : **Appel** : UC02
3. Le client constitue son panier en sélectionnant des articles
4. Le client valide son panier.
5. Le site informe le client que la commande a bien été passée : **Website_Order_Confirmation**
6. Le site internet affiche l'horaire auquel le client pourra venir chercher sa commande au drive-in.
7. Le site internet envoie la commande au système : **Website_Send_Order**

Remarques :

Le cahier des charges ne présentait aucune mention d'un quelconque paiement de la commande internet (et ce bien qu'il explicite le renseignement des informations de paiement, lors de l'inscription). Par conséquent, j'ai décidé de ne pas traiter ce cas, dans la mesure où beaucoup de solutions existent, et que certaines nécessitent des traitements et explications qui, à mon sens, nuiraient au travail demandé. Succinctement, les options les plus sensées sont les suivantes : le client internet paie sa commande au moment de son passage sur le site internet ; cela soulève divers problèmes de gestion

des stocks (que se passe-t-il si un produit figurant sur la commande arrive à rupture de stock lors de la préparation de celle-ci?). On peut aussi présumer que le client est automatiquement crédité dès lors que la commande est validée (encore une fois, il faudrait gérer les cas où certains produits figurant sur la commande manquent à l'appel, ou l'éventualité d'une impossibilité de paiement pour plusieurs raisons). Je tenais à vous en faire part, et me suis, dans ce cas précis, tenu aux éléments donnés dans le cahier des charges, bien que dans le cas d'une modélisation professionnelle, cela n'aurait aucunement été le cas.

V.2 UC02 : Se connecter

UseCase 02

Nom : Se connecter

Acteur principal : Client internet

Acteur secondaire : Site internet

Scénario nominal :

1. Le client clique sur le bouton de connexion
2. Le client renseigne son adresse mail ainsi que son mot de passe, puis valide
3. Le site internet affiche une confirmation de connexion, et redirige le client vers la page dédiée aux commandes

Post-conditions :

Le client se trouve désormais sur la page principale du site, où il peut sélectionner des produits.

Alternatives/Exceptions :

E1 : Client inconnu

3a. Les informations données par le client ne correspondent pas à celles d'un compte existant. Le site internet affiche un message d'erreur. Retour à l'étape 2.

A1 : Création de compte

2a. Le client décide de se créer un compte : **Appel** : UC03

V.3 UC03 : S'inscrire

UseCase 03

Nom : S'inscrire

Acteur principal : Client internet

Acteur secondaire : Site internet

Scénario nominal :

1. Le client clique sur le bouton "s'inscrire".
2. Le client renseigne ses nom, prénom, adresse postale, téléphone, adresse mail, mot de passe ainsi que ses informations de paiement, puis valide.
3. Le site internet affiche une confirmation d'inscription, et redirige le client vers la page de connexion : **aller au UseCase 02 : 2.**

Alternatives/Exceptions :

E1 : Informations déjà utilisées

3a. L'adresse mail choisie par le client est celle d'un compte existant. Le site affiche un message d'erreur. Retour à l'étape 2.

3b. Le numéro de téléphone choisi par le client est celui d'un compte déjà existant. Le site affiche un message d'erreur. Retour à l'étape 2.

E2 : Informations invalides

3a. L'adresse postale choisie est invalide. Le site affiche un message d'erreur, retour à l'étape 2.

3b. Les informations de paiement choisies sont invalides. Le site affiche un message d'erreur, retour à l'étape 2.

V.4 UC04 : Envoyer demande de récupération produit à client physique

UseCase 04

Nom : Envoyer demande de récupération produit à client physique

Acteur secondaire : Tablette

Scénario nominal :

1. Le système reçoit une commande : **Appel à UC05.**
2. Le système cherche les caddies les plus proches : **Appel UC13**
3. Le système envoie une demande de récupération de produit aux tablettes situées sur chaque caddie : **Sys_Send_Item_Retrieval**

V.5 UC05 : Recevoir commande

UseCase 05

Nom : Recevoir commande

Acteur secondaire : Serveur de BD indiquant le numéro d'une commande internet et le nombre de produits de celle-ci

Scénario nominal :

1. Le système reçoit une commande du site internet : **Sys_Receive_Order**
2. Le système insère la commande au sein d'une des tables du serveur de BDD : **System_DB_Order_Insertion**

Alternatives/Exceptions :

E1 : Commande vide

2a. La commande envoyée par le site internet est vide. Le système renvoie une erreur **Sys_Empty_Order** au site internet, qui renvoie la commande.

V.6 UC06 : Accepter demande de récupération d'un produit

UseCase 06

Nom : Accepter demande de récupération d'un produit

Acteur principal : Client physique

Acteurs secondaires : Tablette, serveur de BD indiquant les produits dont chaque client physique s'occupe

Préconditions : la tablette est activée.

Scénario nominal :

1. Le système envoie des demandes de récupération aux clients physiques : **Appel à UC04**
2. La tablette affiche un message proposant au client physique d'accepter une demande de récupération de produit de client internet : **Tablet_Display_Item_Retrieval**
3. Le système arme un timer associé à la confirmation : **System_Set_Confirmation_Timer**
4. Un client physique appuie sur le bouton de confirmation avant la fin du timer.
5. La tablette envoie au système le signal **Tablet_Retrieval_Confirmation**
6. Le système désarme le premier timer.
7. Le système arme un timer associé à la récupération : **System_Set_Retrieval_Timer**

8. Le système envoie une requête vers le serveur de bases de données afin d'ajouter le produit à ceux dont s'occupe le client physique : **System_SQL_INSERT_Item**
9. Un observateur d'état se charge de répercuter la mise à jour du nombre de commandes acceptées par ce client, et ce dans le système : **Observer_Notify_System**
10. La tablette affiche le produit dont le client physique devra se charger : **Tablet_Display_Item**

Post-conditions :

- Le nombre de produits dont s'occupe un client physique est, après confirmation de la prise en charge d'un produit, incrémenté. Ainsi, on peut voir cette postcondition comme une assertion : $new.produitsRécupérés = old.produitsRecupérés + 1$
- Un timer de récupération est désormais déclenché.

Alternatives/Exceptions :

E1 : Confirmation tardive

4a. Le système a déjà reçu un signal **Tablet_Retrieval_Confirmation** pour ce produit, signifiant qu'un client a déjà confirmé la récupération. La tablette affiche un message d'erreur : **Tablet_Retrieval_Error**

E2 : Refus de prise en charge

3a. Le client physique clique sur le bouton d'information. La tablette envoie au système le signal **Tablet_Retrieval_Refusal**

E3 : Aucune confirmation

4a. Aucun client physique ne confirme la prise en charge d'un produit, et ce durant le temps imparti par le premier timer. Le système reçoit un signal **CONFIRMATION_TIME_OUT**, et c'est un employé de magasin qui se chargera du produit : **Appel à UC11**

Exigences non-fonctionnelles

Le temps de réponse requis sera paramétrable au sein de l'application. Il convient de choisir une durée significative, pour permettre aux clients de répondre aux demandes.

V.7 UC07 : Déposer un produit dans une boîte de commande

UseCase 07

Nom : UC07 : Déposer un produit dans une boîte de commande

Acteur principal : Client physique

Acteurs secondaires : Boîte de commande

Scénario nominal :

1. Le client physique accepte une demande de récupération d'un produit : **Appel : UseCase 05.**
2. Le client physique récupère le produit.
3. Le client physique se dirige vers une boîte de commande.
4. Le client physique rentre le numéro du caddie.
5. **Appel : UC08.**
6. Le client physique dépose le produit dans la boîte.

Alternatives/Exceptions :

E1 : Numéro de caddie inconnu

5a. Le numéro de caddie renseigné par le client physique est inconnu. L'écran affiche un message d'erreur : **ScanningScreen_Wrong_Caddie_Number**, retour à l'étape 4.

V.8 UC08 : Valider un produit

UseCase 08

Nom : Valider un produit

Acteur principal : Client physique

Acteur secondaire : écran de scan

Hypothèse : un timer de récupération associé au produit à valider est armé, et son temps imparti ne s'est pas entièrement écoulé.

Précondition : Le client physique a rentré son numéro de caddie.

Scénario nominal :

1. Le client scanne son produit.
2. Le client valide le produit avant que le timer ne soit désarmé.
3. L'écran de scan envoie au système un signal de confirmation de la bonne prise en charge du produit : **Item_Retrieval_Success**
4. Le système retire, dans la BDD, le produit de la commande en question : **System_DB_Item_Validation**
5. La BDD envoie un signal au système l'informant de la bonne suppression de la ligne : **DB_Notify_System_Deletion**
6. Le système désarme le timer : **System_Set_Retrieval_Timer_Off**

Post-conditions :

La commande dont le produit vient d'être validé est désormais "lestée" d'un produit :

ainsi, dans la DB, une ligne correspondant au produit en question, pour cette commande, sera retirée.

Alternatives/Exceptions :

A1 : le client change de produit

2a. Le client change de produit en appuyant sur le bouton retour. Retour à l'étape 1.

A2 : la commande du client internet est complète

5a. La BDD ne comporte plus, après suppression du produit, de ligne associée à la commande : celle-ci est complète. La BDD envoie un signal **Order_Complete** au système, ce qui déclenche **UC10**. La BDD envoie ensuite le signal de suppression, puis retour à l'étape 6.

E1 : fin du timer de récupération

3a. Le système reçoit un signal du timer de récupération **Retrieval_Timer_Time_Out**.

3b. Le système envoie un message d'annulation à la tablette, qui l'affiche.

3c. Le système envoie une requête à la BDD pour supprimer la ligne correspondant au produit dont s'occupe le client physique : **System_DB_Delete_Item**

3d. Un observateur d'état interne au système lui retransmet le nouveau nombre de commandes dont s'occupe le client physique : **Observer_Notify_System**

3e. L'employé de magasin s'occupe par la suite de ce produit : **Appel à UC11**

E2 : mauvais produit

5a. Le produit qui a été scanné ne correspond pas à un produit contenu dans la commande. Dans ce cas, rien ne se passe : la suppression n'a pas lieu au sein de la BDD, qui envoie au système un signal d'erreur : **DB_Item_Not_Found**

Exigences non-fonctionnelles

Le délai de récupération sera paramétrable au sein de l'application. Il convient de choisir une durée significative et réaliste, pour permettre au client de pouvoir récupérer le produit sans sentiment de contrainte, tout en garantissant la bonne préparation d'une commande.

Remarques :

De cette façon, scanner des produits non-contenus dans une commande ne changera rien. En effet, le scan sert simplement, dans le cas d'une validation correcte (donc d'un produit bien contenu dans une commande), à tenir compte de l'avancement de la préparation. Le prix de la commande internet étant prévu à l'avance, j'ai pris cette

liberté, d'autant plus qu'il ne nous était pas demandé de gérer le cas où un article commandé par un client internet s'avère être en rupture de stock.

V.9 UC09 : Passer en caisse

UseCase 09

Nom : Passer en caisse

Acteur principal : Client physique

Acteur secondaire : Caisse automatique

Hypothèse : Le client physique s'est occupé d'au moins une récupération de produits.

Scénario nominal :

1. Le client se dirige vers une caisse automatique.
2. La caisse automatique affiche un message de bienvenue invitant le client à scanner ses courses.
3. Le client physique scanne ses courses.
4. Le client physique clique sur le bouton "Total"
5. La caisse invite le client physique à saisir son numéro de caddie.
6. Le client saisit son numéro de caddie.
7. **Appel UC12.**
8. La caisse affiche le montant de la réduction, ainsi que les totaux avant et après réduction.
9. Le client choisit une option de paiement
10. Le client procède au paiement
11. La caisse valide le paiement
12. La caisse envoie au système un signal l'informant de la complétion des courses du client physique : **Paystation_Shopping_Complete**
13. Le système supprime de la base de données les lignes correspondant au numéro de caddie du client : **System_Delete_Client_Items**
14. Un observateur d'état répercute le changement du nombre de produits récupérés par le caddie du client physique à l'ensemble du système : **Observer_Notify_System**

Post-conditions :

- Le nombre de produits dont s'occupe le client est désormais nul (concrètement, dans la BDD, le nombre de produits associé au numéro de caddie du client est nul).

V.10 UC10 : Vérifier la commande d'un client

UseCase 10

Nom : UC10 : Vérifier la commande d'un client

Acteur primaire : Employé de magasin

Acteurs secondaires : écran de scan, serveur de BD indiquant les commandes internet

Scénario nominal :

1. Le système reçoit un signal **Order_Completed**.
2. Un employé de magasin se rend à la boîte de commande.
3. L'employé scanne sa carte d'employé
4. L'écran de scan affiche un menu de contrôle des commandes
5. L'employé passe en revue les produits dans la boîte de commande
6. L'employé clique sur le bouton "Valider la commande"
7. L'écran affiche le message "La commande internet est désormais validée".
8. L'écran de scan envoie un signal au système : **Order_Validated**
9. Le système envoie une requête à la base de données pour supprimer la commande :
System_Delete_Internet_Order
10. L'écran de scan affiche de nouveau une page d'accueil.

Post-conditions :

La commande traitée par le client physique n'apparaît plus dans la base de données.

Alternative/Exceptions :

A1 : Produit en trop

6a. L'employé, en vérifiant la commande, a trouvé un produit en trop. Dans ce cas, il le retire de la boîte. Retour à l'étape 5.

V.11 UC11 : Prendre en charge la récupération d'un produit

UseCase 11

Nom : UC11 : Prendre en charge la récupération d'un produit

Acteur primaire : Employé de magasin

Acteurs secondaires : écran de scan

Scénario nominal :

1. Le système reçoit un signal **Appel UC11**
2. L'employé de magasin récupère le produit.

3. L'employé de magasin se rend à la boîte de commande.
4. L'employé de magasin scanne sa carte d'employé.
5. L'écran de scan affiche un menu de contrôle des commandes.
6. L'employé clique sur le bouton "Déposer un produit"
7. L'écran de scan affiche un message demandant à l'employé de rentrer le numéro de la commande.
8. L'employé saisit le numéro de la commande.
9. L'employé scanne le produit.
10. L'employé appuie sur le bouton de validation.
11. L'écran de scan affiche un message de confirmation de la validation du produit.
12. Le système envoie une requête à la BDD pour rajouter ce produit dont s'occupe l'employé. **System_DB_Add_Clerk_Item**

V.12 UC12 : Appliquer des réductions

UseCase 12

Nom : UC12 : Appliquer des réductions

Acteur primaire : Client physique

Acteurs secondaires : Serveur de BD, caisse automatique

Scénario nominal :

1. Le client physique rentre son numéro de caddie sur l'écran de la caisse.
2. Le client physique valide sa saisie.
3. La caisse automatique envoie au système un signal lui demandant de récupérer la somme du nombre de produits dont s'est occupé le client, et du nombre de produits récupérés par les employés, pour ces mêmes commandes : **Paystation_Request_Item_Number**
4. Le système interroge les tables pour récupérer cette donnée : **System_SQL_Request_COUNT_Item_Number**
5. Le système envoie la donnée à la caisse automatique : **System_Send_Item_Number**
6. La caisse calcule le montant de la réduction, à partir de cette donnée.

Alternatives/Exceptions

E1 : numéro de caddie incorrect

3a. Le numéro de caddie est incorrect. La caisse automatique affiche un message d'erreur.
Retour en 1.

V.13 UC13 : Chercher les caddies les plus proches

UseCase 13

Nom : UC13 : Chercher les caddies les plus proches

Acteurs secondaires : serveur de BD, tablette

Hypothèse : un observateur d'état transmet, à un intervalle régulier, la position de chaque caddie au système. Un autre se charge de transmettre à "ce cas d'utilisation" le nombre de commandes déjà acceptées par un client.

Scénario nominal :

1. Le système récupère la position du produit.
2. Le système choisit, en fonction de cette position, les caddies les plus proches, en ne considérant que ceux dont le nombre de commandes est inférieur ou égal à la limite instaurée par le système.

Post-conditions :

Le système a récupéré au moins un caddie éligible à la prise en charge d'un produit.

Alternatives/exceptions :

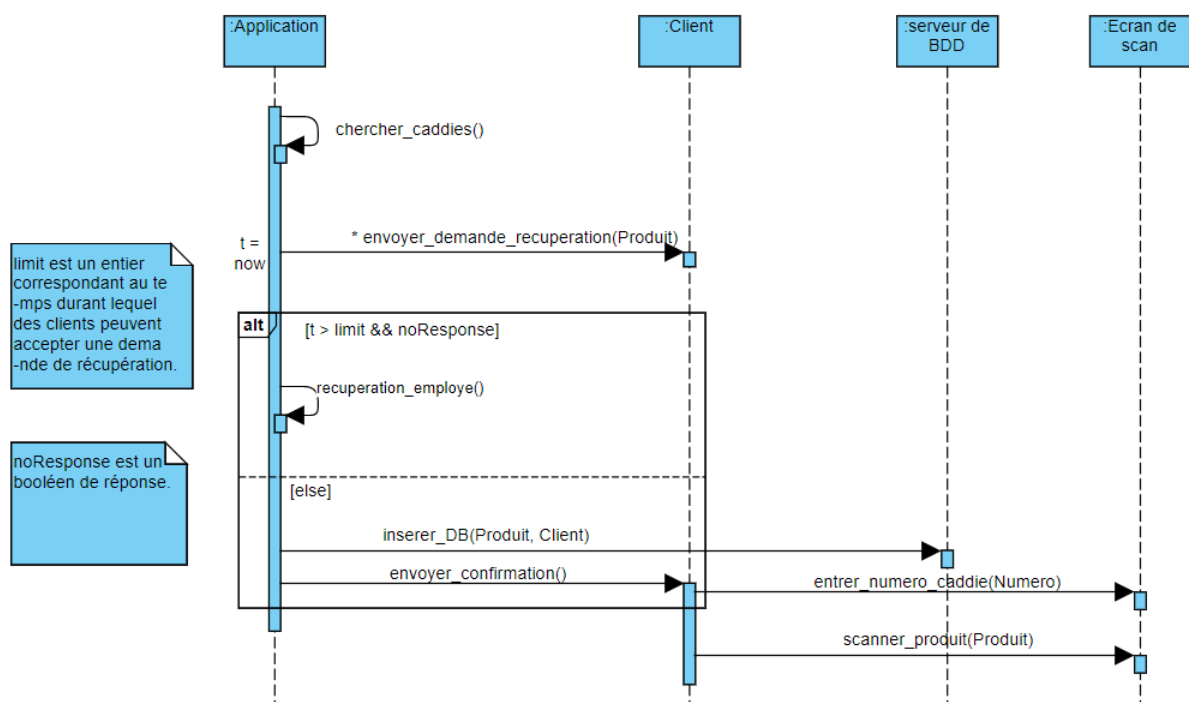
E1 : Pas de caddie trouvé

3a. La recherche de caddie ne renvoie aucun résultat. Dans ce cas, le système retourne à l'étape 2, après un certain délai. Après trois tentatives infructueuses, le système génère une erreur **Sys_No_Caddie_Found**, et un employé s'occupe de la récupération du produit en question : **Appel : UC11** .

Section VI

Diagramme de séquence

La question 5 nous demandait de réaliser un diagramme de séquence donnant l'ordre des tâches accomplies par les Use-Case "Envoyer demande de récupération aux clients" et "Valider le produit".



L'aspect principal que j'ai cherché à expliciter, dans ce diagramme de séquence, est l'absence de réponse de clients dans un temps imparti. Pour ce faire, j'ai utilisé un fragment alt, séparant les deux cas : le premier, où le temps imparti pour accepter une demande de récupération est arrivé à sa fin ($t > \text{limit}$), sans qu'au moins client physique n'ait répondu OK (noResponse). Dans ce cas, le système génère un signal, et c'est un employé qui se charge de la récupération.

Dans l'autre cas, on insère des éléments propres à la commande et au client physique dans la base de données, et on envoie une confirmation de récupération au client. A

partir de là, les deux actions sont simples : le client va entrer son numéro de caddie, puis scanner le produit.

Section VII

Propriétés de l'interaction

L'interaction élémentaire à laquelle on s'intéresse est "envoyer_demande_client", ou **Sys_Send_Item_Retrieval**, dans UC04.

Ce signal est provoqué par l'obtention des caddies les plus proches d'un produit donné (ultimement, on peut remonter et dire que l'envoi ne se fait que si l'application a reçu une commande provenant d'un site internet). On ne peut donc envoyer des demandes que si l'on a au préalable reçu une commande non-vide, et cherché des caddies à qui envoyer des demandes.

Cependant, intéressons-nous davantage à la recherche de caddies.

Section VIII

Description d'un observateur d'état sur le nombre de commandes d'un client

Dans le cahier des charges qui nous a été donné, il était spécifié qu'un client physique ne pouvait dépasser une certaine limite de produits à récupérer. C'est là qu'un observateur d'état interne au système intervient. Initialement, dès qu'un client se saisit d'un caddie,

son nombre de produits récupérés est nul. Cependant, à chaque acceptation de demande de récupération, ce nombre va se voir incrémenté. Analogiquement, ce nombre se voit décrémenté si le client physique n'a pas récupéré le produit à temps, et est remis à zéro dès qu'un client a effectué ses courses.

La mise à jour va donc se faire dans ces deux cas : lorsqu'un client acceptera une demande de récupération, un observateur d'état se chargera de transmettre cette information au système (et plus précisément à UC13, le cas d'utilisation de recherche des caddies les plus proches d'un produit donné. Analogiquement, ce même UseCase sera "averti" dès lors qu'un client physique ne récupère pas un produit donné à temps (ce qui amènera une décrémentation du nombre de produits dont s'occupe ledit client).

Notre observateur fonctionne donc sur des mises à jour. Cette façon de procéder, bien que très efficace, me paraît dangereuse, car le bon fonctionnement du système dépend donc de l'intégrité et du fonctionnement de cet observateur d'état, qui devra donc bénéficier de maintenances régulières, au risque de voir des clients s'occuper d'un nombre trop important de produits.