

INFO-H-200 : Programmation orientée objet

TP 6- UML Diagramme de séquence

Professeur : Hugues Bersini

<http://cs.ulb.ac.be/public/teaching/infoh200>

Année académique 2015-2016

Exercice 6.1

Un restaurant décide de se munir d'un système informatique pour pouvoir gérer les commandes et faire un suivi de la clientèle. Lorsqu'un client passe commande, il fournit un identifiant (ex : son numéro de téléphone), ainsi que sa commande composée d'une série de plats et de leurs quantités respectives. En réponse à sa commande, le client va recevoir le prix de sa commande. S'il effectue le paiement demandé, le restaurant va lancer les préparations pour ensuite les livrer au client.

Pour suivre la clientèle, le restaurant sauvegarde dans son système séparément chaque commande pour ses différents clients. Il peut ainsi retrouver toutes les informations de ces derniers à partir de son identifiant et y ajouter toutes les informations relatives à sa nouvelle commande. Une commande se composera à son tour d'une liste de pizzas, identifiées à partir de leur nom, et de leur nombre.

Dessiner un diagramme de séquence répondant au problème, en supposant les interactions d'un acteur vis-à-vis d'objets de type `Restaurant`, `Client`, `Commande` et `Pizza`.

Exercice 6.2

Modélisez le diagramme de séquence correspondant au scénario suivant : Un client interagit avec un système d'achat en ligne dans le quel il parcourt un catalogue de produits et peut ajouter ceux-ci dans un panier d'achats avant de finaliser de celui-ci en demandant une facture. On vous demande de réaliser le diagramme de séquence des interactions client '`ajouterArticle()`' et '`voirFacture()`'. Attention, lors de l'ajout d'un article au panier, la catalogue doit être consulté pour en re-vérifier le prix. Comment pourriez-vous modéliser le fait que l'interaction '`ajouterArticle()`' peut être reconduite *n* fois ?

Exercice 6.3

Le scénario que l'on vous demande de modéliser est celui correspondant à l'usage d'un ascenseur par un acteur. On vous demande de réaliser le diagramme de séquence modélisant les interactions entre l'utilisateur et l'ascenseur entre le moment où le premier appelle le second, attends qu'il arrive, monte dans celui-ci, et sélectionne son étage. Les différentes opérations que l'on vous demande de décrire sont suivent votre intuition : '`appel()`', '`allumerVoyant()`', '`ouvrirPorte()`', '`fermerPorte()`', '`monter()`', '`descendre()`', '`entrer()`', '`attente()`'. Présenter le schéma de tel façon que cette succession d'opération puisse se répéter.

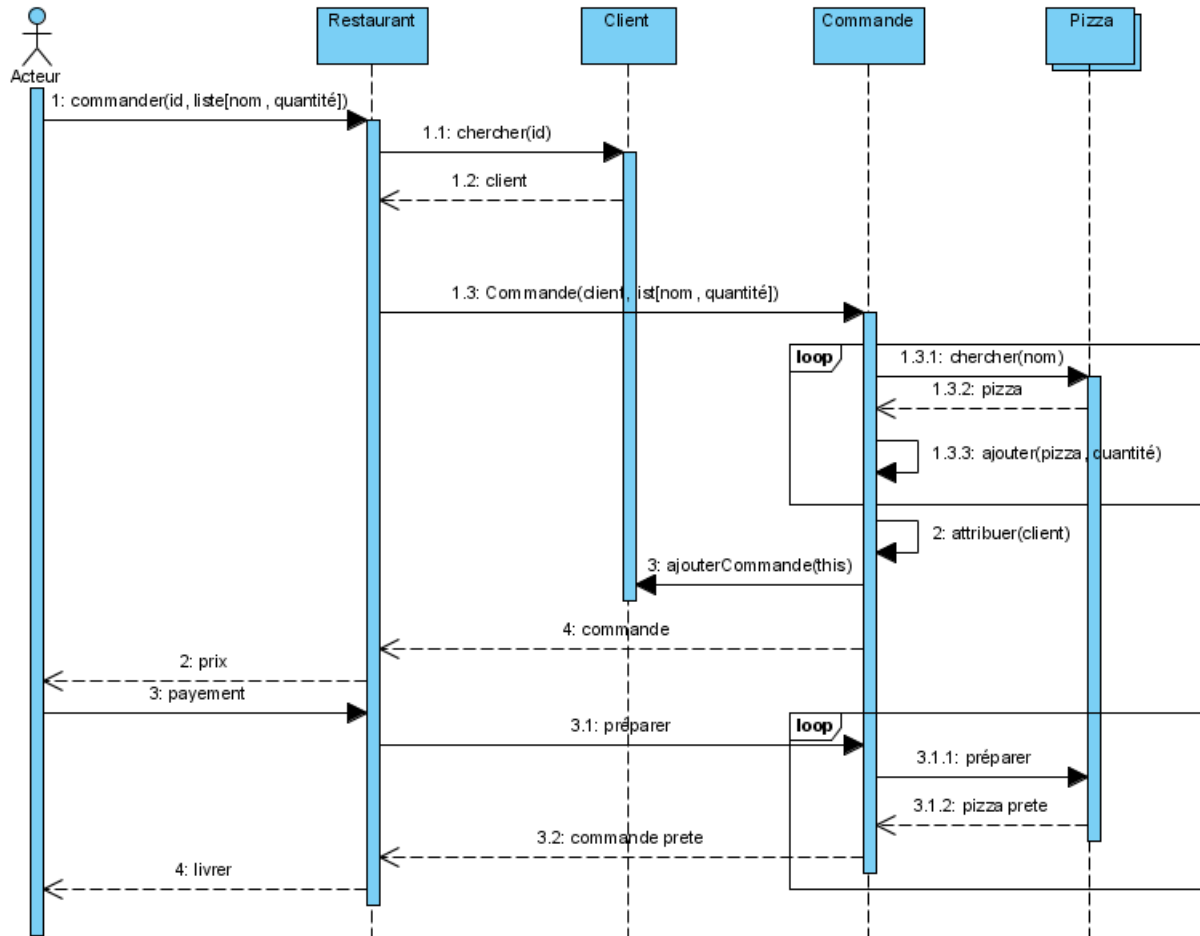
Modéliser ensuite le même scénario à l'aide d'un timer et sans faire mention de l'utilisateur. L'ascenseur est ainsi vu comme un système autonome. Cette fois, veillez à utiliser au mieux les blocs d'interactions pour ne pas répéter inutilement celles-ci.

Exercice 6.4

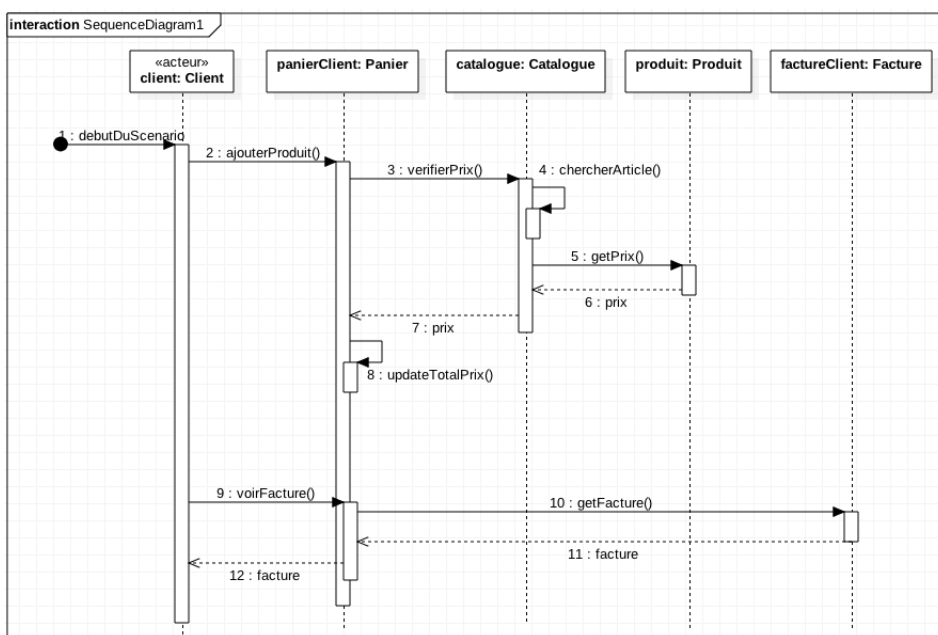
On vous demande de réaliser le diagramme de séquence du scénario suivant... Dans le jeu Bomberman, un joueur dépose une bombe là où il se trouve et ne se déplace plus ensuite. La bombe décompte un temps donné (e.g : 3 secondes) et explose en tuant le joueur. Voici l'ordre dans lequel les diverses opérations interviennent :

1. Le Joueur (e.g : l'acteur), appuye sur la barre d'espacement.
2. Le Jeu reçoit cette information et, pour le bomberman en question, crée une bombe.
3. Le Jeu met à jour la vue du plateau de jeu
4. La Bombe est initialisée avec un timeout donné.
5. Le compte à rebour de la bombe atteint zéro et celle-ci explose.
6. La Bombe notifie le Jeu de sa propre explosion avant de s'auto-supprimer.
7. Le Jeu calcule ensuite les dégats.
8. Le Bomberman du joueur est donc touché par l'explosion et le Jeu indique au bomberman concerné qu'il est touché.
9. Le nombre de vies du bomberman est vérifié et il s'avère qu'il est nul. Le bomberman signifie donc au jeu qu'il est mort.
10. Le Jeu met à jour sa liste de Bomberman puis met à jour à la vue du plateau de jeu.

Exercice 6.1



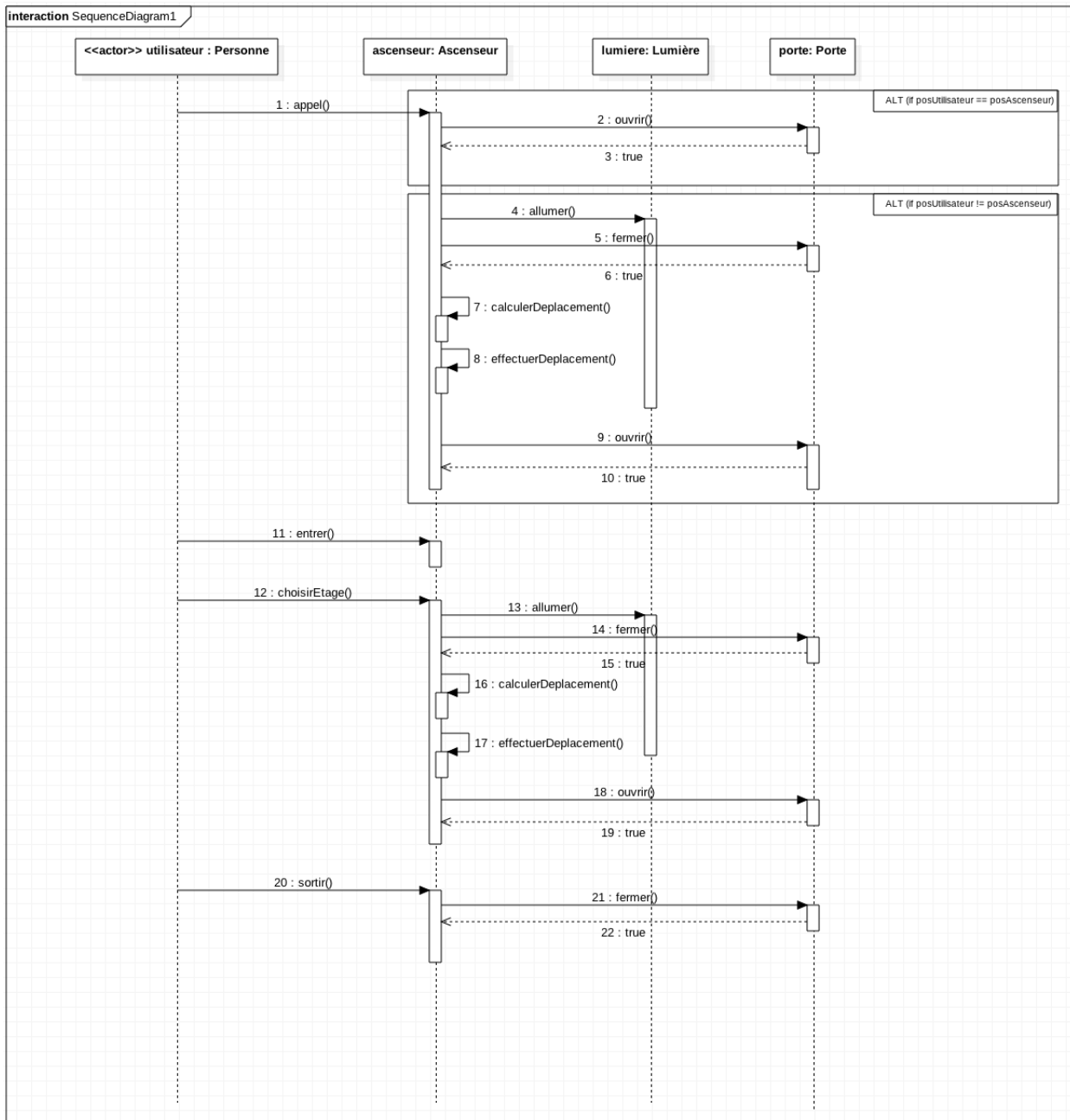
Exercice 6.2



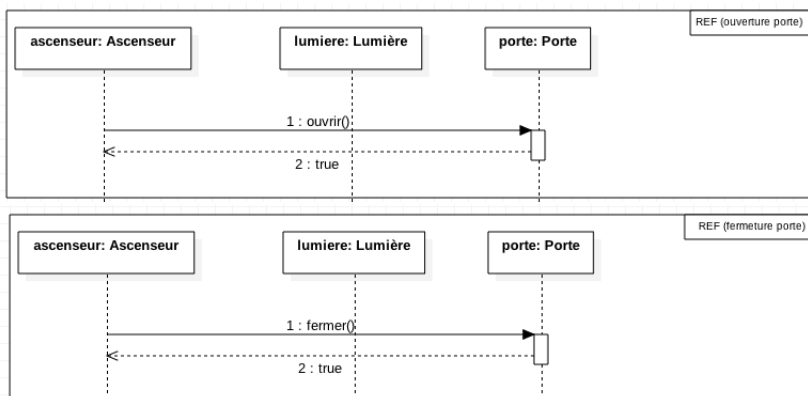
Dans le cas où l'interaction 'ajouterArticle()' peut être reconduite n fois, les opérations 2 à 8 seraient à placer dans un sous-cadre d'interaction 'loop (n)'.

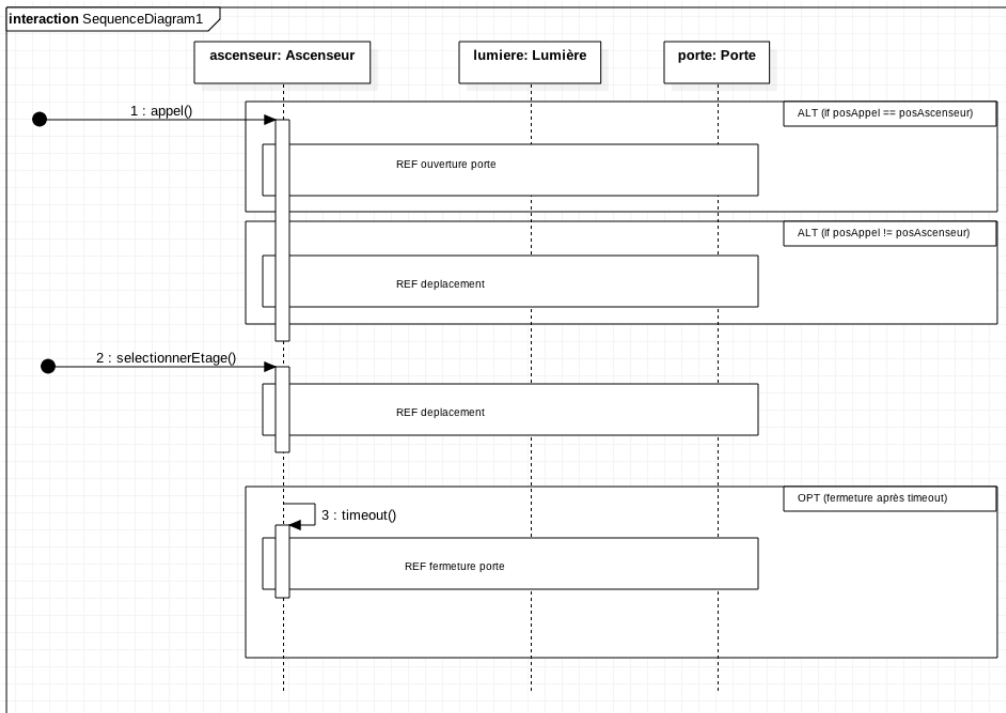
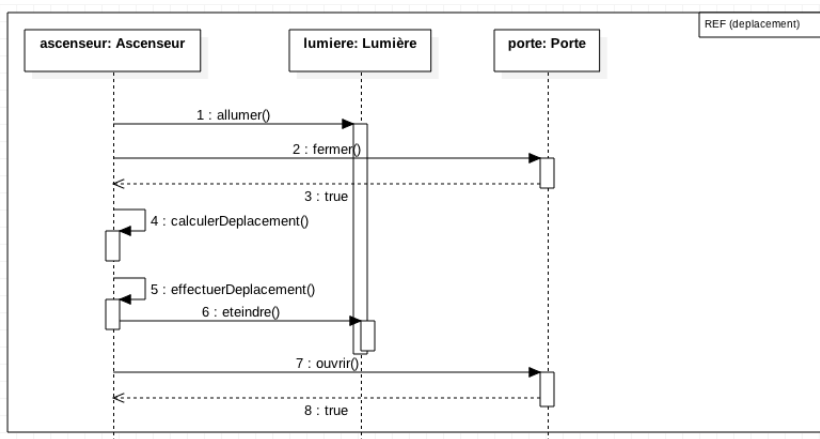
Exercice 6.3

Première partie



Seconde partie





Exercise 6.4

