

OC Pizza

Système de gestion de prise de commande

Dossier de conception fonctionnelle

Version 1.0

Auteur

Alix VOINOT

Developpeur

Sommaire

TABLE DES MATIÈRES

1 - Versions.....	3
2 - Introduction.....	4
2.1 - Objet du document.....	4
2.2 - Références.....	4
2.3 - Besoin du client.....	4
2.3.1 - Contexte.....	4
2.3.2 - Enjeux et Objectifs.....	4
2.3.3 - Contraintes.....	5
3 - Description générale de la solution.....	6
3.1 - Les acteurs.....	6
3.2 - Principes de fonctionnements.....	7
3.3 - Les cas d'utilisation généraux.....	8
3.3.1 - Use case n°1 Connexion (package authentication).....	8
3.3.1.1 - Diagramme use case package authentication.....	8
3.3.1.2 - Explication "Connexion".....	8
3.3.2 - Use case n°2 Effectuer une commande en ligne (package commande).....	9
4 - Le domaine fonctionnel.....	11
4.1 - Présentation diagramme de classe.....	11
5 - Les workflows.....	14
5.1 - Diagramme de séquence.....	14
5.2 - Diagramme d'état transition.....	14
6 - Glossaire.....	15

1 - VERSIONS

Auteur	Date	Description	Version
Alix VOINOT	23/02/2018	Première création du document	1.0
Alix VOINOT	25/02/2019	Harmonisation du document avec le dossier de conception technique	2.0

2 - INTRODUCTION

2.1 - Objet du document

Le présent document constitue le dossier de conception fonctionnelle de l'application www.ocpizza.com.

OC Pizza est un nouveau groupe de pizzerias en plein essor. Un de ses responsables, nous a contacté pour les aider à concevoir un système leur facilitant le travail et les accompagnant dans leur perspective d'évolution.

2.2 - Références

Pour de plus amples informations, se référer également aux éléments suivants :

1. Dossier d'exploitation
2. Dossier de conception technique

2.3 - Besoin du client

2.3.1 - Contexte

OC Pizza compte actuellement plusieurs points de vente et a prévu d'en ouvrir des nouveaux. Le système devra donc être conçu en prenant en compte cette évolution. OC Pizza a donc besoin d'un système permettant de faciliter la prise de commande.

2.3.2 - Enjeux et Objectifs

Le système devra répondre aux attentes suivantes:

- permettre la prise de commande en ligne
- simplifier la prise de commande sur place
- fournir un suivi en temps réel des différentes commandes
- fournir un suivi en temps réel du stock au personnel de la pizzeria
- mettre à jour en temps réel les produits disponible sur le site

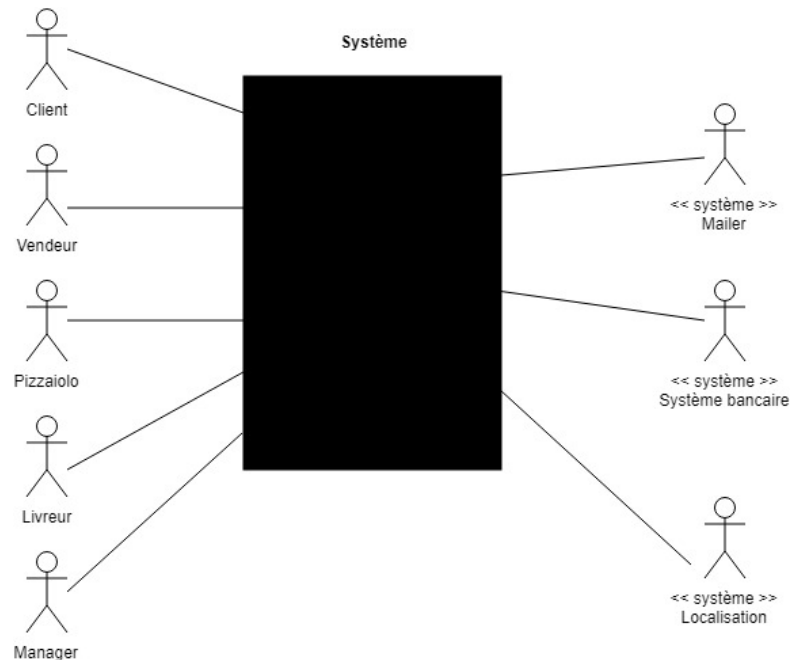
2.3.3 - Contraintes

Le site devra mettre à jour en permanence les produits disponibles en se basant sur les données du stock de sa pizzeria. Les produits ou les pizzas ayant un ingrédient en rupture de stock seront grisaillés sur le site et donc indisponible à l'achat.

Le système devra aussi fournir un aide-mémoire pour le pizzaiolo pour lui simplifier la réalisation de la pizza.

3 - DESCRIPTION GÉNÉRALE DE LA SOLUTION

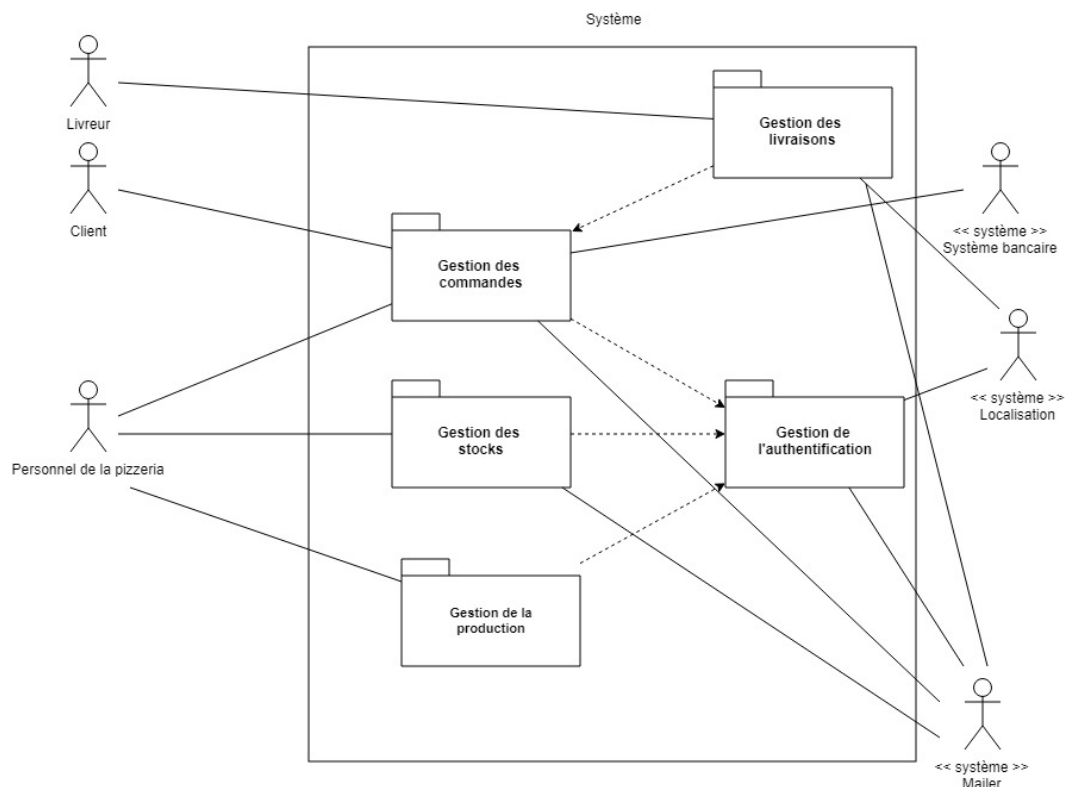
3.1 - Les acteurs



- **Client** : un client utilisateur du système
- **Vendeur** : un vendeur de la pizzeria
- **Pizzaiolo** : un pizzaiolo de la pizzeria
- **Livreur** : un livreur de la pizzeria
- **Manager** : le manager de la pizzeria
- **Mailer** : le service d'envoi de mail automatique
- **Système bancaire** : le service de paiement en ligne
- **Localisation** : le service de localisation

Les acteurs pizzaiolo, manager et vendeur seront regroupés dans l'acteur personnel de la pizzeria dans les futurs diagrammes. Le livreur n'est pas compris car son utilisation du système est peu importante.

3.2 - Principes de fonctionnements

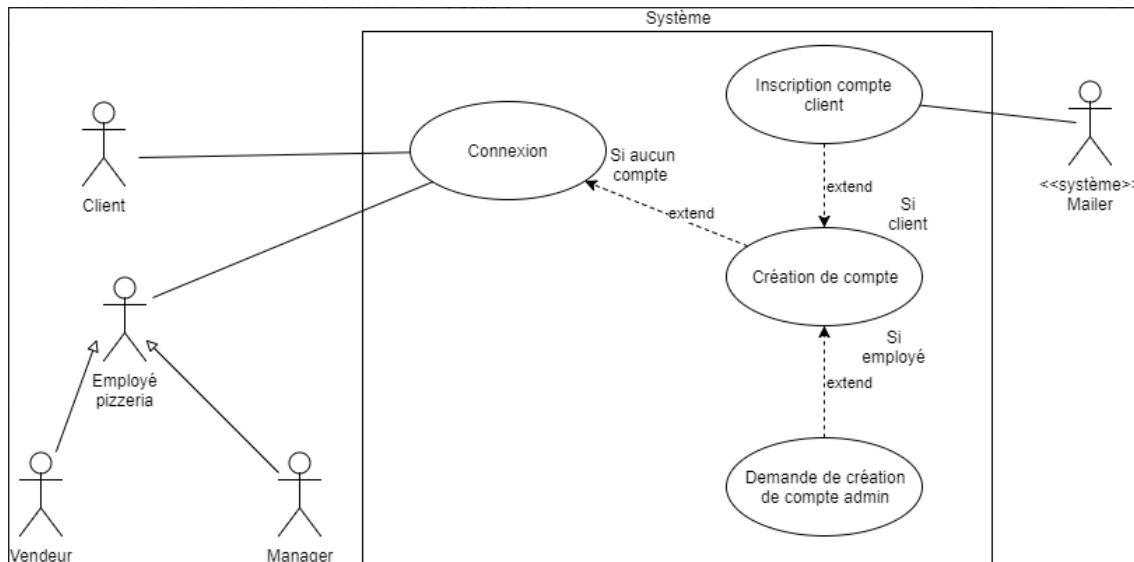


- **Package commande** : Contient les fonctionnalités liées à la commande (consulter la carte, passer une commande...)
- **Package livraison** : Contient les fonctionnalités liées à la livraison (livrer la commande, récupérer la commande en restaurant...)
- **Package authentification** : Contient les fonctionnalités liées à l'authentification (connexion, demande de nouveau mot de passe...)
- **Package stock** : Contient les fonctionnalités liées au stock (mise à jour en temps réel des stocks, seuil d'alerte, mise à jour de la carte...)
- **Package production** : Contient les fonctionnalités liées à la production (ajouter/retirer une pizza, statistique de production...)

3.3 - Les cas d'utilisation généraux

3.3.1 - Use case n°1 Connexion (package authentication)

3.3.1.1 - Diagramme use case package authentication



3.3.1.2 - Explication "Connexion"

Nom : Connexion (package « Authentification »)

Diagramme use case (Authentification)

Acteur(s) : Client et Employé Pizzeria

Description : Le client se connecte au système pour avoir accès aux fonctionnalités lui permettant de valider sa commande. Les employés de la pizzeria se connectent au système pour avoir accès aux fonctionnalités administrateur.

Pré-conditions : Le client doit avoir un compte. Les employés de la pizzeria doivent avoir un compte.

Démarrage : Le client doit se connecter avec son compte pour valider sa commande. Les employés se connectent pour accéder aux fonctionnalités dont ils ont besoin.

Scénario nominal

1. Le système affiche le formulaire d'authentification 2. L'utilisateur entre ses identifiants 3. Le système vérifie les données entrées et renvoie sur la page du système correspondant au type du compte

Scénario d'exception

3.a Les identifiants de l'utilisateur sont incorrectes Le système envoie un message d'erreur et demande à l'utilisateur de ressayer

3.b L'utilisateur n'a pas de compte Le système fait appel au cas d'utilisation interne « Création de compte »

Fin et post-conditions

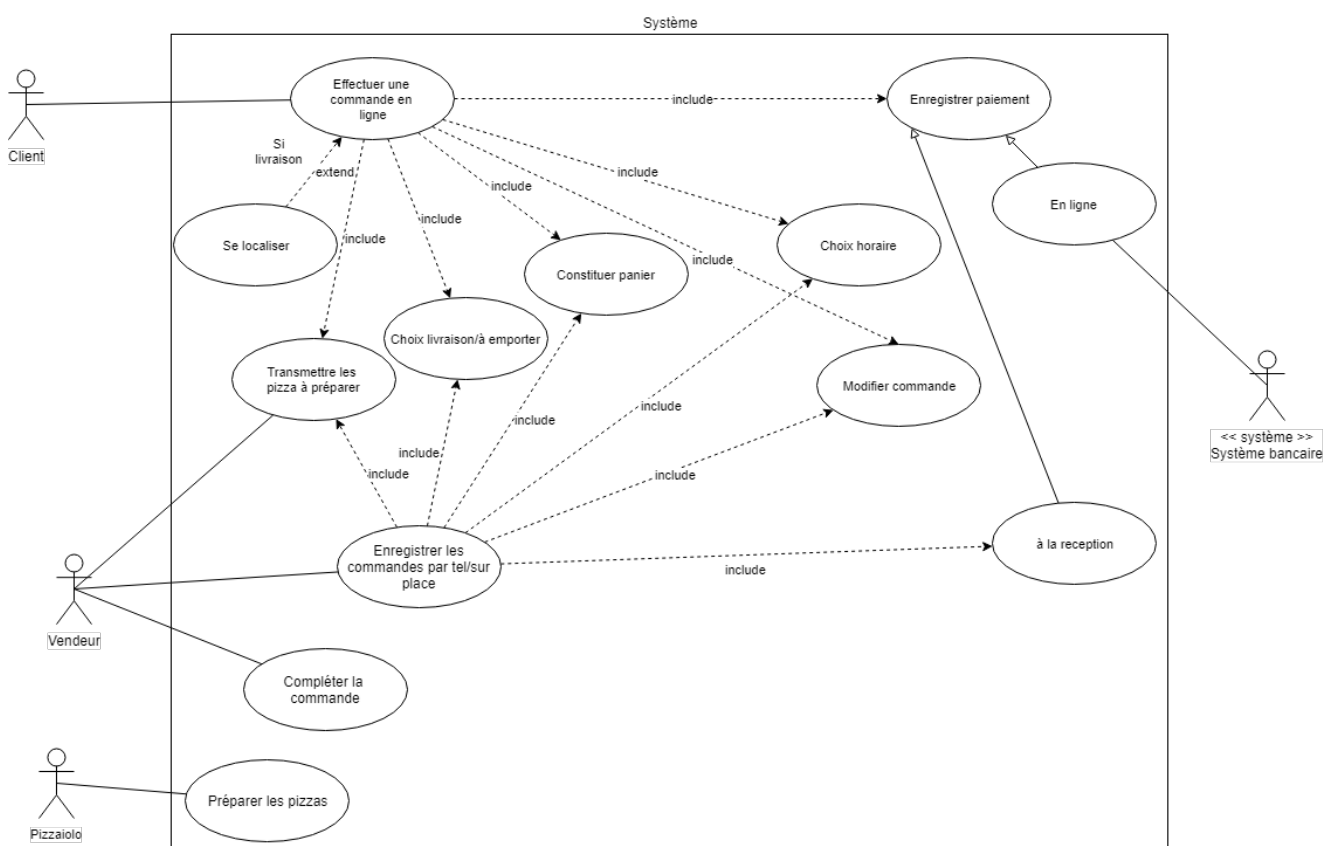
Scénario nominal : après le point 3 Scénario d'exception : après le point 3.a

Ergonomie : La page contiendra seulement 2 champs ID et mot de passe ainsi que deux liens « Création de compte » et « mot de passe oublié ».

Performance attendues : Se connecter en moins de 20 secondes pour un client, en 5 secondes si ses informations sont sauvegardées. Pour les employés, la connexion est automatique lorsque l'on lance le système sur leur borne.

3.3.2 - Use case n°2 Effectuer une commande en ligne (package commande)

3.3.2.1 Diagramme



3.3.2.2 Explication “effectuer une commande en ligne”

Nom : Effectuer une commande en ligne (« package commande »)

Acteur(s) : Client

Description : Le client se rend sur le site de la pizzeria, constitue un panier, renseigne ses informations et valide sa commande.

Démarrage : Le client arrive sur le site internet de la pizzeria

Scénario nominal:

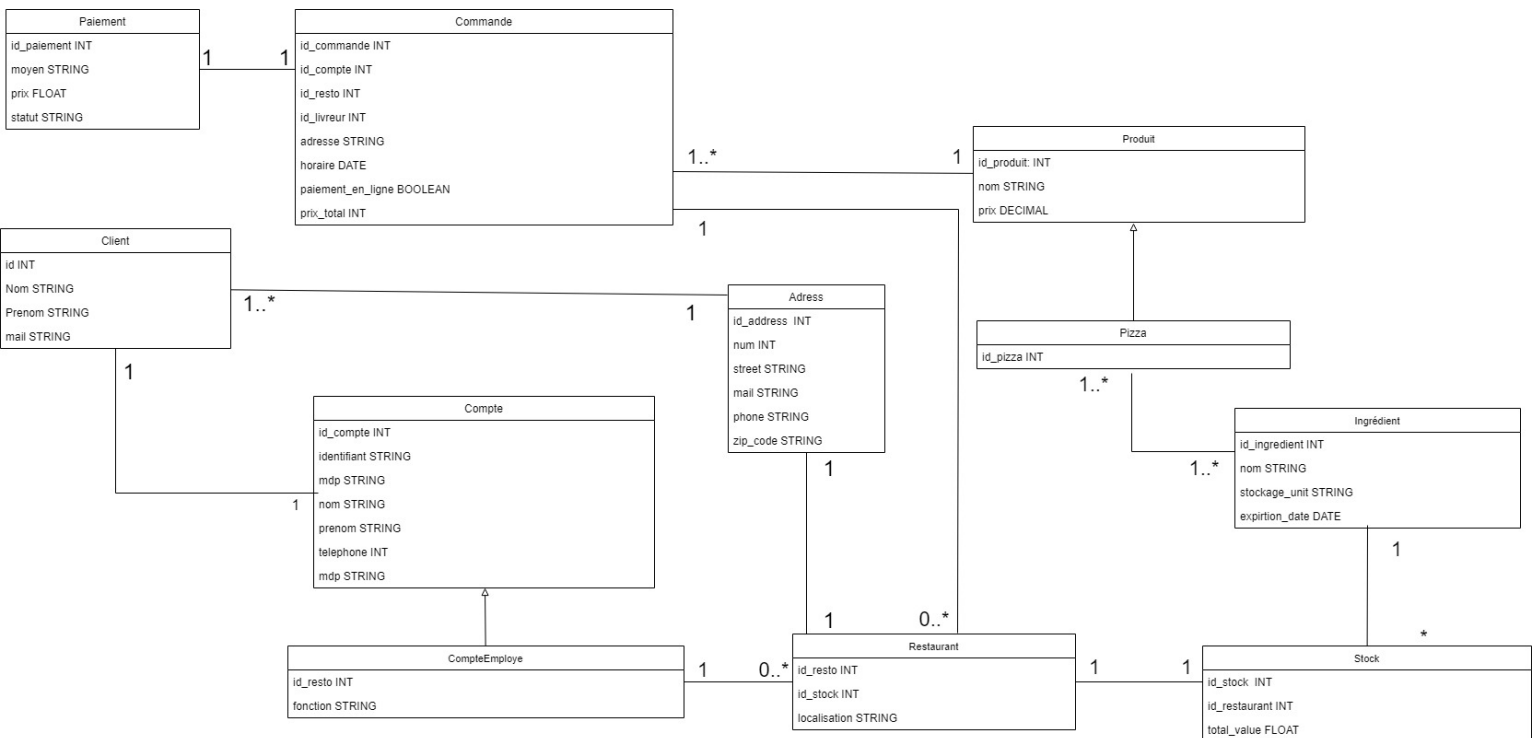
1. Le système fait appel au cas d'utilisation interne « Choix livraison /à emporter »
2. Le système fait appel au cas d'utilisation interne « Se localiser »
3. Le système fait appel au cas d'utilisation interne « Choix horaire »
4. Le système fait appel au cas d'utilisation interne « Constituer panier »
5. L'utilisateur se connecte (ou le système fait appel au cas d'utilisation connexion?)
6. Le système fait appel au cas d'utilisation interne « Enregistrer paiement »

Fin : scénario nominal, à l'étape 6 ou sur la volonté de l'utilisateur.

Performance attendu : Une commande doit prendre entre 5 et 10 minutes.

4 - LE DOMAINE FONCTIONNEL

4.1 - Présentation diagramme de classe



Compte : Classe contenant les différentes informations qu'a un compte du système

- *id_compte* : numéro identifiant du compte
- *identifiant* : identifiant permettant de se connecter au système
- *mdp* : mot de passe permettant de se connecter au système
- *nom* : nom de l'utilisateur du compte
- *prénom* : prénom de l'utilisateur du compte
- *téléphone* : téléphone de l'utilisateur du compte

Les informations de cette classe sont stockées dans la table *Account* du **MPD**.

CompteEmploye : Classe héritant de la classe **Compte**. Représente un type de compte particulier (employé)

- *id_resto* : numéro identifiant de la pizzeria à laquelle est associé ce compte
- *fonction* : fonction au sein de la pizzeria de l'utilisateur. Définit les privilèges du compte

Les informations de cette classe sont stockées dans la table *Account* du **MPD**.

Restaurant : Classe contenant les différentes informations d'un restaurant

- *id_resto* : numéro identifiant de cette pizzeria
- *id_stock* : le numéro d'identifiant du stock de cette pizzeria
- *localisation* : adresse de la pizzeria

Les informations de cette classe sont stockées dans la table *Restaurant* du **MPD**.

Stock : Classe contenant les différentes informations d'un stock

- *id_stock* : numéro identifiant du stock
- *id_restaurant* : Numéro de la pizzeria à laquelle est associé ce stock

Les informations de cette classe sont stockées dans la table *Stock* du **MPD**.

Ingrédient : Classe contenant les différentes informations d'un ingrédient

- *id_ingrédient* : numéro identifiant de cette ingrédient
- *nom* : nom de l'ingrédient
- *stockage_unit* : unité de stockage de cet ingrédient (kg, l...)
- *expiration_date* : date d'expiration de l'ingrédient

Les données de cette classe sont stockées dans la table *Ingrédient*, *PizzaIngrédient* et *StockIngredient* du **MPD**.

Produit : Classe contenant les différentes informations d'un produit

- *id_produit* : numéro identifiant de ce produit
- *nom* : nom du produit
- *prix* : prix du produit

Les données de cette classe sont stockées dans la table *Product* et *OrderProduct* du **MPD**.

Pizza : Classe contenant les différentes informations d'une pizza

- *id_pizza* :numéro identifiant de cette pizza

Les données de cette classe sont stockées dans la table *Pizza* et *PizzaIngrédient* du **MPD**.

Client : Classe contenant les différentes informations d'un client

- *id_client* : numéro identifiant de ce client
- *nom* : nom du client
- *prenom* : prénom du client
- *mail* : mail du client

Les informations de cette classe sont stockées dans la table *Customer* du **MPD**.

Adresse : Classe contenant les différentes informations d'une adresse

- *id_address* : numéro identifiant de cette adresse
- *num* : numéro de rue de l'adresse
- *street* : rue de l'adresse
- *mail* : mail associé à l'adresse
- *phone* : numéro de téléphone associé à l'adresse
- *zip_code* : code postal associé à l'adresse

Les informations de cette classe sont stockées dans la table *Address* du **MPD**.

Paiement : Classe contenant les différentes informations du paiement d'une commande

- *id_paiement* : numéro identifiant de ce paiement
- *moyen* : moyen de paiement (carte bancaire, paypal...)
- *prix* : prix de la commande
- *statut* : état du paiement (payé/ impayé)

Les informations de cette classe sont stockées dans la table *Payment* du **MPD**.

Commande : Classe contenant les différentes informations d'une commande

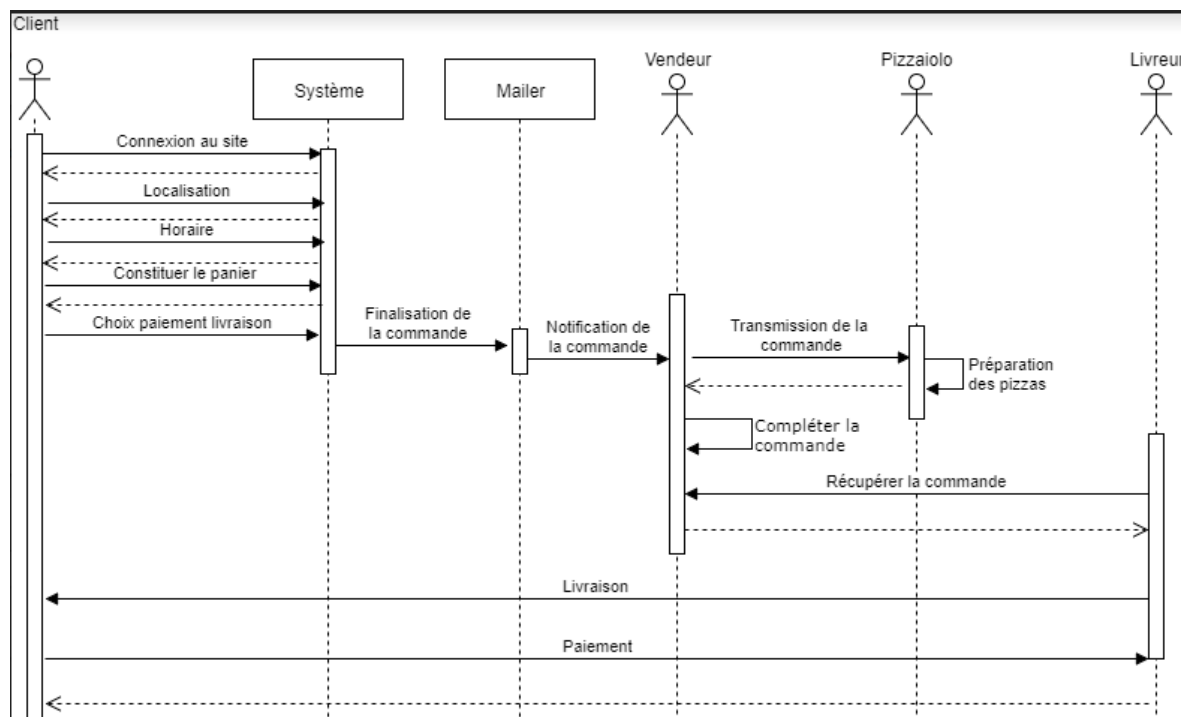
- *id_commande* : numéro identifiant de cette commande
- *id_compte* : numéro du compte associé à la commande
- *id_resto* : numéro de la pizzeria associé à la commande
- *id_livreur* : numéro du livreur associé à la commande
- *horaire* : horaire de reception de la commande (sur place ou par livraison)
- *paiement_en_ligne* : le boolean annonce si le paiement s'effectue en ligne ou non
- *livraison* : le boolean annonce si la commande est livrée à domicile ou non
- *prix_total* : prix total de la commande, livraison comprise

Les informations de cette classe sont stockés dans la table *Order* et *OrderProduct* du **MPD**.

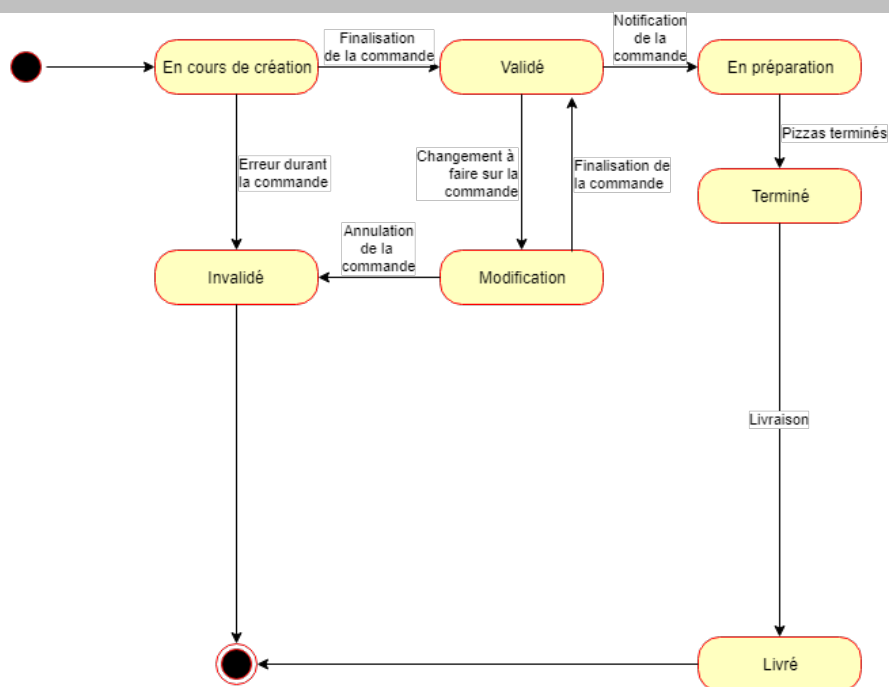
Note : Le MPD est disponible dans le dossier associé à ce fichier ou dans le dossier de conception technique.

5 - LES WORKFLOWS

5.1 - Diagramme de séquence



5.2 - Diagramme d'état transition



6 - GLOSSAIRE

UML	Le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système.
Diagramme use case	Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel.
Diagramme de classes	Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Une classe regroupe des membres, méthodes et propriétés (attributs) communs à un ensemble d'objets.
Diagramme de séquences	Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique.
Diagramme d'état transition	Le diagramme d'état-transition représente tout les différents états que va prendre un objet dans son cycle de vie. Dans notre cas, le diagramme d'état transition illustre le cycle de vie d'une commande.