



**OC PIZZA**

***SPECIFICATIONS FONCTIONNELLES ET SOLUTION TECHNIQUE***

# **SOMMAIRE**

## **1. INTRODUCTION**

1.1 Le client .....	4
1.2 Les besoins .....	4
1.3 Les acteurs .....	4

## **2. MODELISATION GENERALE** .....

2.2 Diagramme de contexte .....	5
2.2.1 Description .....	5
2.1 Diagramme de packages .....	6
2.1.1 Description - Partie : Client .....	6
2.1.2 Description – Partie : Pizzeria .....	6
2.3 Diagramme de cas d'utilisation – Partie : Client .....	7
2.3.1 Description .....	7
2.4 Diagramme de cas d'utilisation – Partie : Pizzeria .....	8
2.4.1 Description .....	8

## **3. SYSTEME D'AUTHENTIFICATION** .....

3.1 Diagramme de cas d'utilisation .....	9
3.1.1 Description .....	9
3.2 <b>PARTIE : AUTHENTIFICATION</b> .....	10
3.2.1 Diagramme de séquence .....	10
3.2.1.1 Description .....	10
3.2.2 Diagramme d'activité .....	11
3.2.2.1 Description .....	11
3.3 <b>PARTIE : CREATION DE COMPTE</b> .....	12
3.3.1 Diagramme de séquence .....	12
3.3.1.1 Description .....	12
3.3.2 Diagramme d'activité .....	13
3.3.2.1 Description .....	13

# **SOMMAIRE**

<b>4. GESTION DES COMMANDES</b>	14
4.1 <b>PARTIE : CLIENT</b>	14
4.1.1 Diagramme de cas d'utilisation	14
4.1.1.1 Description	14
4.1.2 Diagramme de séquence	15
4.1.2.1 Description	15
4.1.3 Diagramme d'activité	16
4.1.3.1 Description	16
4.2 <b>PARTIE : PIZZAIOLO</b>	17
4.2.1 Diagramme de cas d'utilisation	17
4.2.1.1 Description	17
4.2.2 Diagramme de séquence	18
4.2.2.1 Description	18
4.2.3 Diagramme d'activité	19
4.2.3.1 Description	19
4.3 <b>PARTIE : LIVREUR</b>	20
4.3.1 Diagramme de cas d'utilisation	20
4.3.1.1 Description	20
4.3.2 Diagramme de séquence	21
4.3.2.1 Description	22
4.3.3 Diagramme d'activité	23
4.3.3.1 Description	23
<b>5. CYCLE DE VIE D'UNE COMMANDE</b>	24
5.1 Diagramme	24
5.1.1 Description	25
<b>6. SOLUTION TECHNIQUE</b>	26
6.1 Diagramme	26
6.1.1 Description	27

# 1. INTRODUCTION

## 1.1 **LE CLIENT**

**OCPIZZA** est un jeune groupe de pizzeria en plein essor. Créé par Franck et Lola, le groupe est spécialisé dans les pizzas livrées ou à emporter. Il compte déjà 5 points de vente et prévoit d'en ouvrir au moins 3 de plus d'ici 6 mois.

Le système informatique actuel ne correspond plus aux besoins du groupe car il ne permet pas une gestion centralisée de toutes les pizzerias.

De plus, il est très difficile pour les responsables de suivre ce qui se passe dans les points de ventes. Enfin, les livreurs ne peuvent pas indiquer « en live » que la livraison est effectuée.

## 1.2 **LES BESOINS**

Le client souhaite améliorer son système informatique pour pouvoir centraliser la gestion de ses futures pizzerias. Pour cela, il aimerait :

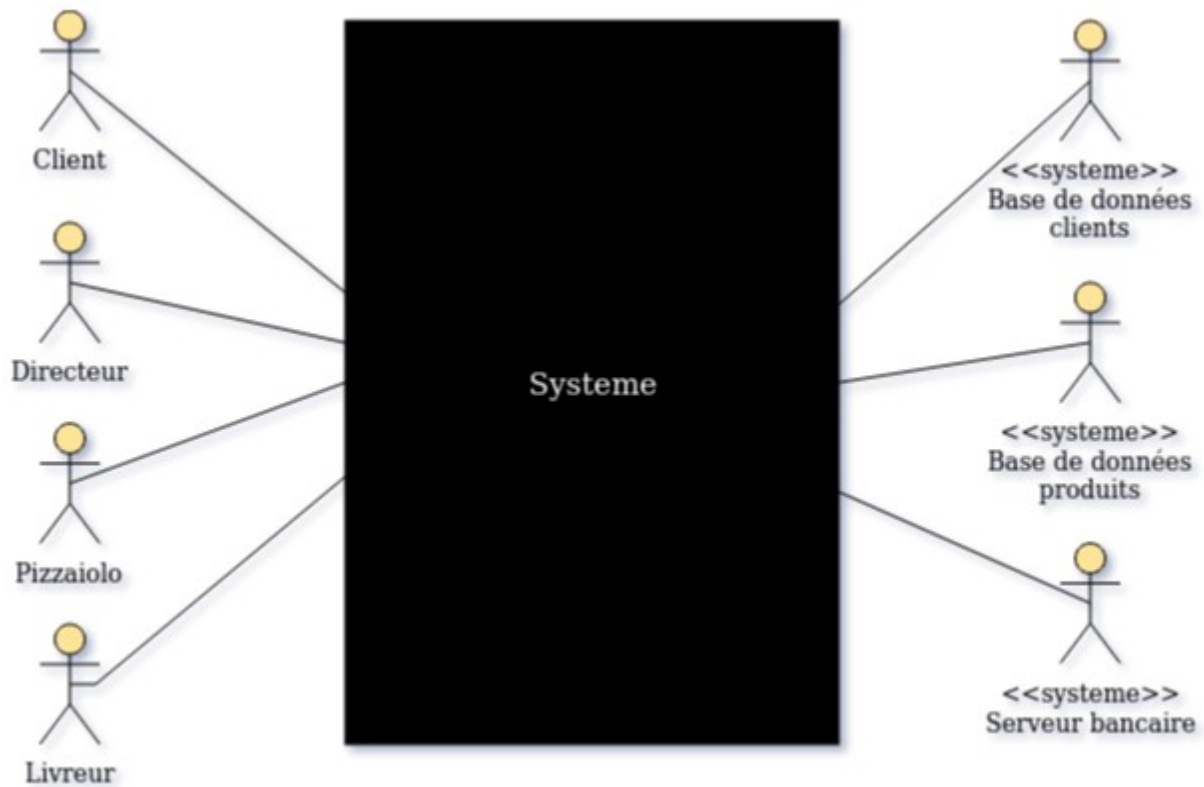
- Etre plus efficace dans la gestion des commandes
- Suivre en temps réel les commandes passées, en préparation et en livraison.
- Ajouter un aide-mémoire des recettes pour les pizaiolos
- Suivre en temps réel le stock d'ingrédients pour savoir quelles pizzas peuvent encore être réalisées.
- Permettre aux livreurs d'indiquer quand la livraison est effectuée
- Proposer un site Internet pour que les clients puissent :
  - Passer leurs commandes, en plus de la prise de commande par téléphone ou sur place.
  - Régler leur commande en ligne, sinon, ils régleront directement à la livraison.
  - Modifier ou annuler leur commande tant que celle-ci n'a pas été préparée.

## 1.3 **LES ACTEURS**

- Le **CLIENT** doit pouvoir :
  - S'authentifier ou créer un compte
  - Passer des commandes par téléphone ou sur Internet
  - Modifier ou annuler sa commande tant qu'elle n'est pas en préparation
  - Régler sa commande par Internet ou à la livraison
- Le **DIRECTEUR** doit pouvoir :
  - Suivre les points de vente
  - Consulter les commandes
- Le **PIZZAIOLO** doit pouvoir :
  - Consulter les commandes
  - Consulter l'aide-mémoire contenant les recettes des pizzas
  - Consulter et mettre à jour le stocks d'ingrédients
  - Mettre jour le statut de la commande pour la livraison
- Le **LIVREUR** doit pouvoir :
  - Mettre à jour les commandes quand elles ont été livrées

## 2. MODELISATION GENERALE

### 2.1 DIAGRAMME DE CONTEXTE

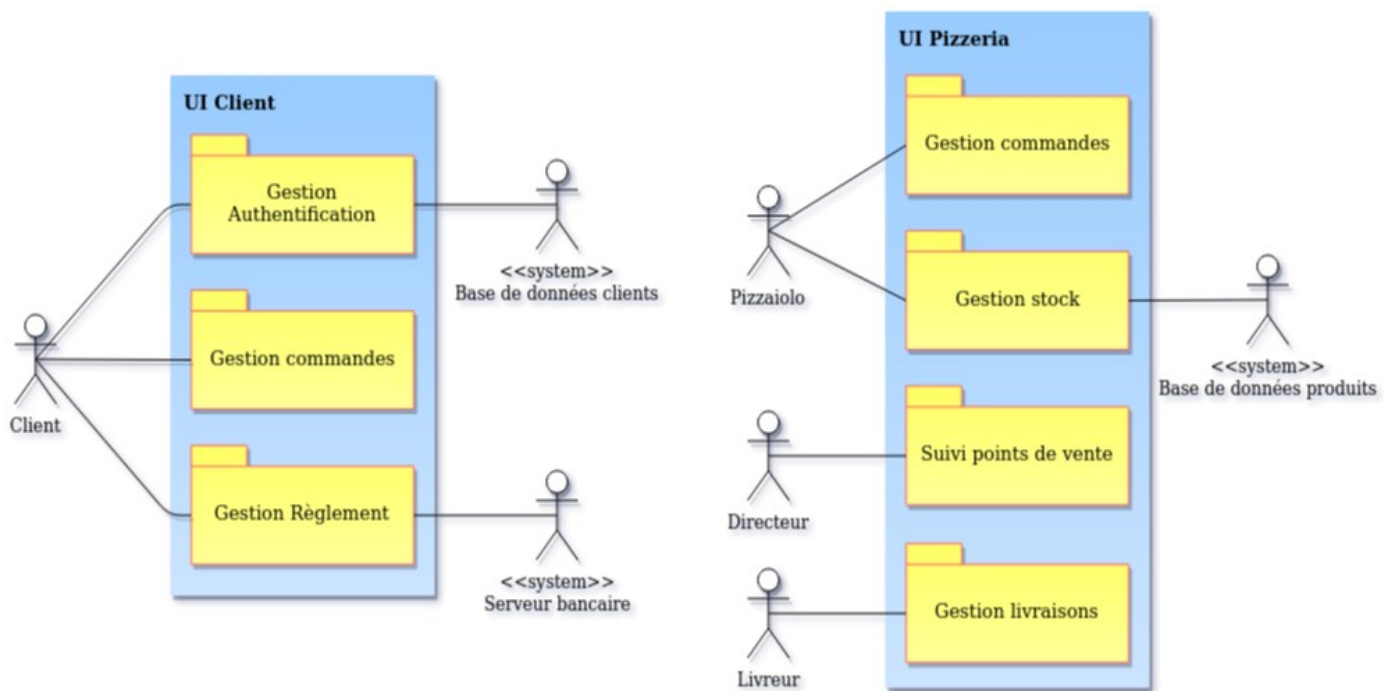


#### 2.1.1 DESCRIPTION

Ce diagramme illustre le système (boîte noire) et :

- *Ses acteurs principaux :*
  - Le client
  - Le directeur
  - Le pizzaiolo
  - Le livreur
- *Ses acteurs secondaires :*
  - La base de données clients
  - La base de données produits
  - Le serveur bancaire

## 2.2 DIAGRAMME DE PACKAGES



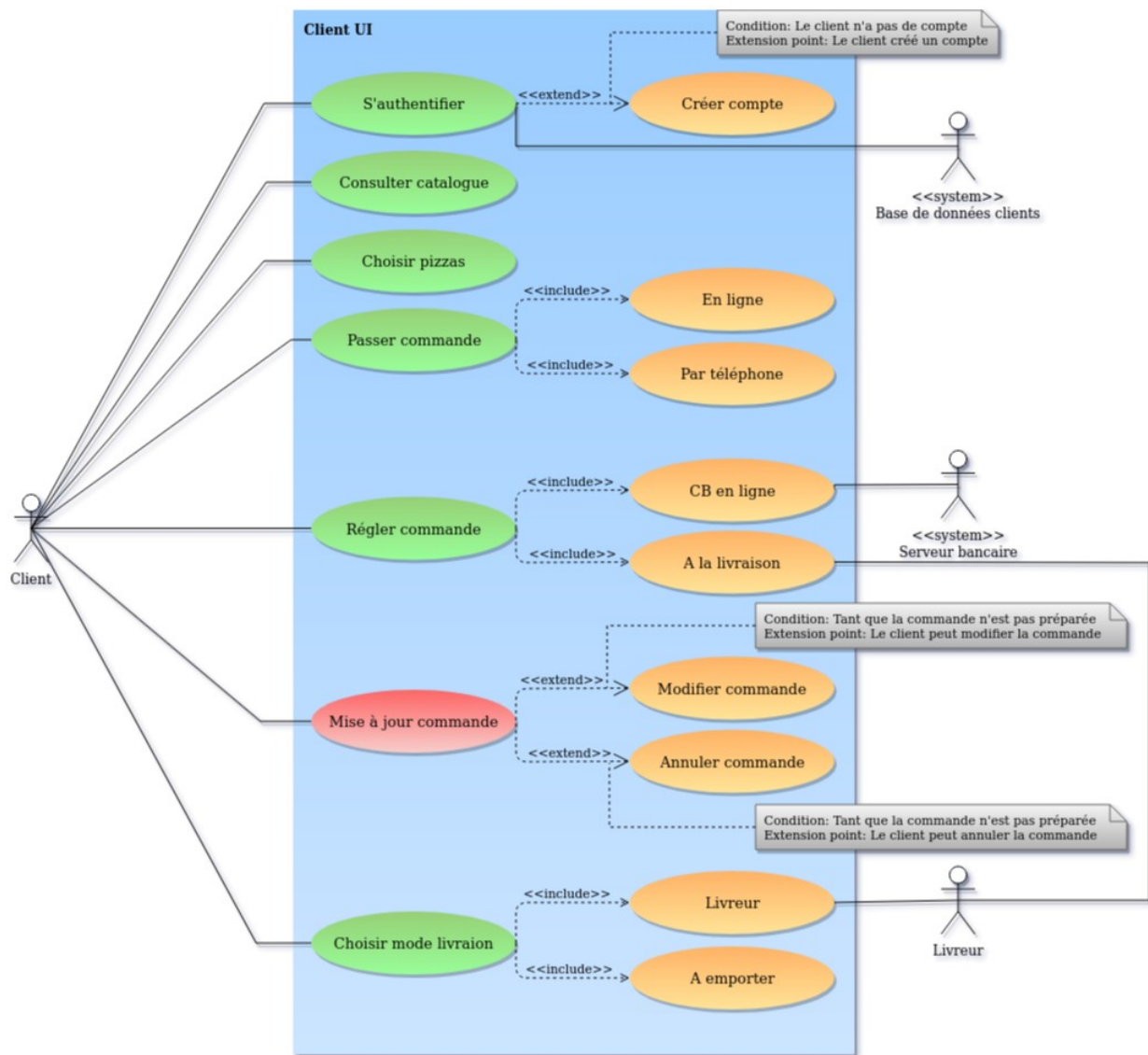
### 2.2.1 Description – Partie : CLIENT

- *Gestion authentification*
  - Identifiant (mail client)
  - Mot de passe
  - Vérification de l'existence du client dans la base de données
  - Renseignement des informations personnelles (Nom, prénom, adresse postale...)
- *Gestion des commandes*
  - Consultation du catalogue des pizzas
  - Choix des pizzas
  - Passage de commande
- *Gestion du règlement*
  - Choix de la méthode de règlement (CB en ligne ou à la livraison)
  - Interrogation serveur bancaire si règlement en ligne

### 2.2.2 Description – Partie : PIZZERIA

- *Gestion des commandes (PIZZAIOLOS)*
  - Réception des commandes
  - Consultation de l'aide-mémoire
  - Préparation des commandes
  - Mise à jour de l'état des commandes
- *Gestion du stock produits (PIZZAIOLOS)*
  - Mise à jour de la base de données produits une fois la commande terminée
- *Suivi des points de vente (DIRECTEUR)*
  - Choix du point de vente à suivre
- *Gestion des livraisons (LIVREUR)*
  - Mise à jour de l'état de la commande une fois livrée

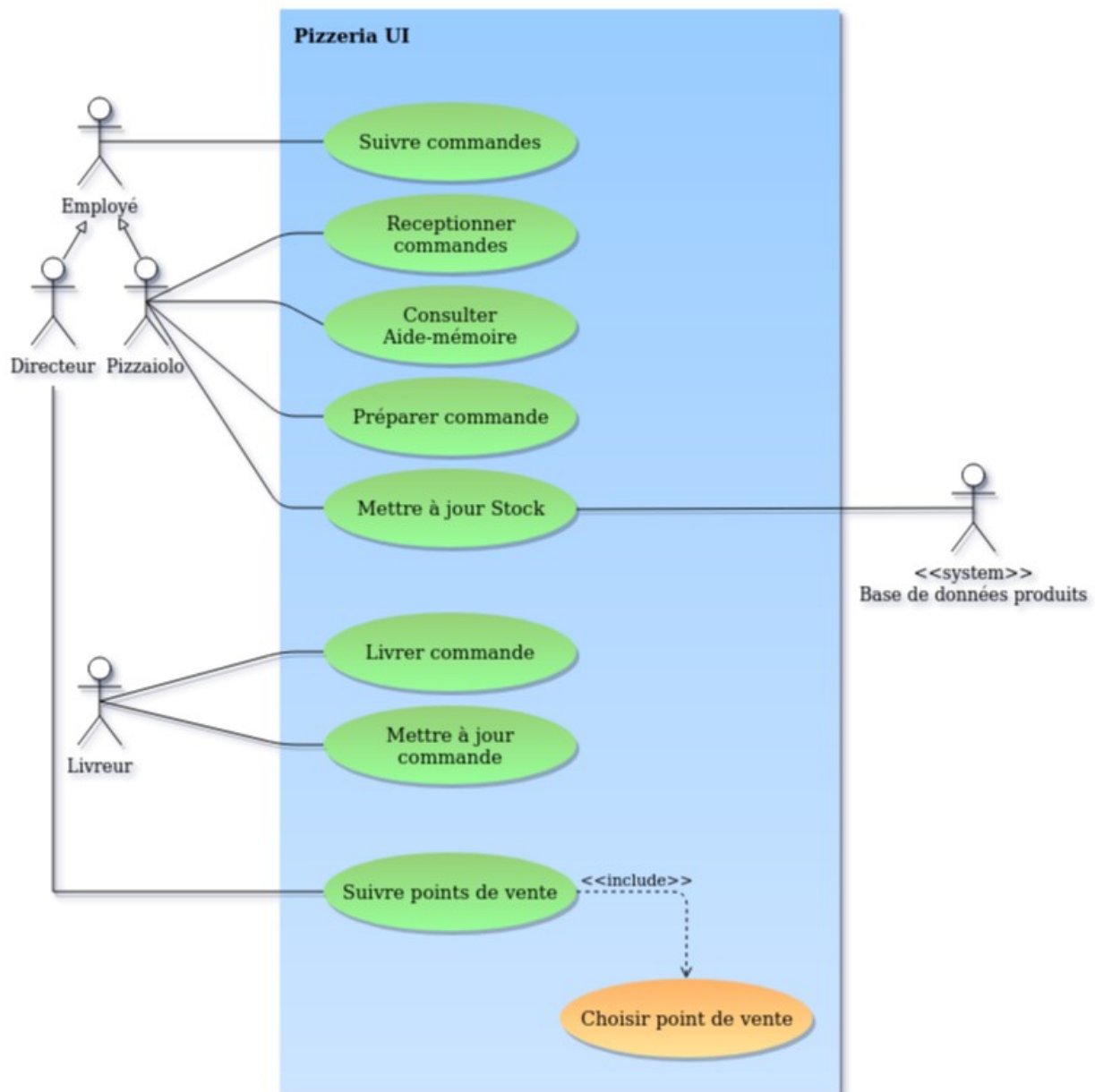
## 2.3 DIAGRAMME DE CAS D'UTILISATION - PARTIE : CLIENT



### 2.3.1 DESCRIPTION

- Le client doit s'authentifier. S'il n'a pas de compte, le client doit créer un compte
- Le client consulte le catalogue des pizzas
- Le client choisit sa ou ses pizzas
- Le client passe sa commande
  - Le client choisit de payer sa commande en ligne → Appel serveur bancaire
  - Le client choisit de payer sa commande à la livraison
- La commande est en attente de préparation (Action non effectuée par le client)
  - Tant qu'elle n'est pas préparée, le client peut modifier sa commande
  - Tant qu'elle n'est pas préparée, le client peut annuler sa commande
- Le client choisit le mode de livraison
  - Le client peut être livré
  - Le client peut aller chercher sa commande au comptoir

## 2.4 DIAGRAMME DE CAS D'UTILISATION - PARTIE : PIZZERIA



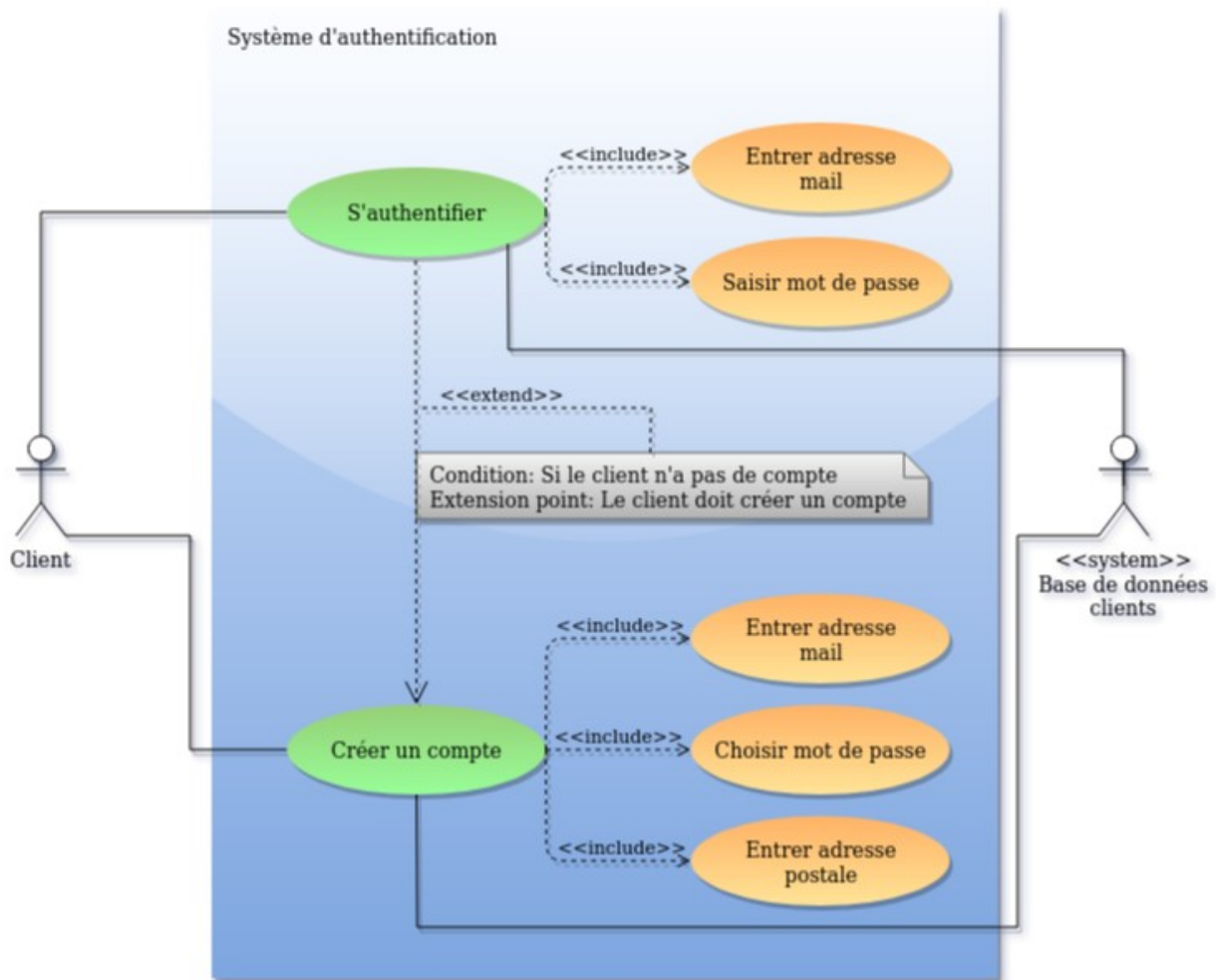
### 2.4.1 DESCRIPTION

- Le pizzaiolo et le directeur suivent l'arrivée des commandes
- Le pizzaiolo prend connaissance de la commande suivante
- Le pizzaiolo consulte l'aide-mémoire des recettes des pizzas
- Le pizzaiolo prépare les commandes
- Le pizzaiolo met à jour la base de données des produits
- Le livreur livre la commande
- Le livreur met à jour le statut de livraison de la commande
- Le directeur suit les ventes de chaque point de vente
  - Le directeur choisit un point de vente à suivre



### 3. SYSTEME D'AUTHENTIFICATION

#### 3.1 DIAGRAMME DE CAS D'UTILISATION

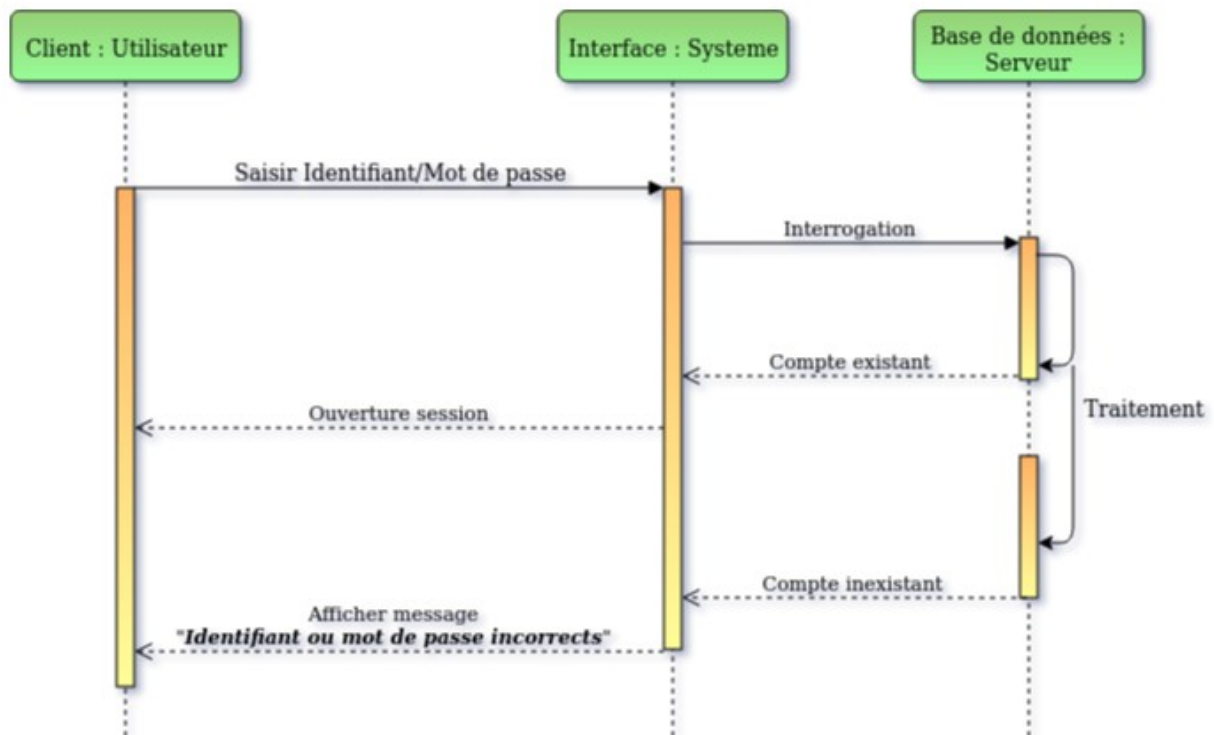


##### 3.1.1 DESCRIPTION

- Le client doit s'authentifier avec son adresse mail et son mot de passe (Uniquement si le compte est créé)
- Le système vérifie l'existence du compte dans la base de données et autorise l'accès aux commandes
- Si le compte n'existe pas, le client crée un compte
  - Le client entre son adresse mail comme identifiant
  - Le client choisit un mot de passe (alphanumérique et 8 caractères minimum)
  - Le client renseigne son adresse postale comme adresse de livraison
- Le système enregistre le compte du client dans la base de données

## 3.2 PARTIE : AUTHENTIFICATION

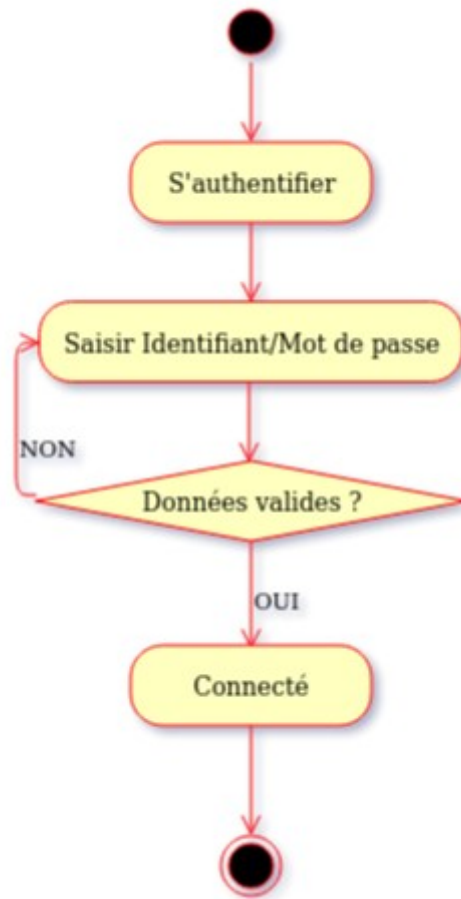
### 3.2.1 DIAGRAMME DE SEQUENCE



#### 3.2.1.1 DESCRIPTION

- Le client doit saisir son identifiant et son mot de passe
- Le système interroge la base de données clients
- La base de données vérifie que le compte existe
  - ✓ Le compte existe
    - Le système affiche un message indiquant au client qu'il est connecté
  - ✗ Le compte n'existe pas
    - Le système affiche un message invitant le client à créer un compte

### 3.2.2 **DIAGRAMME D'ACTIVITE**

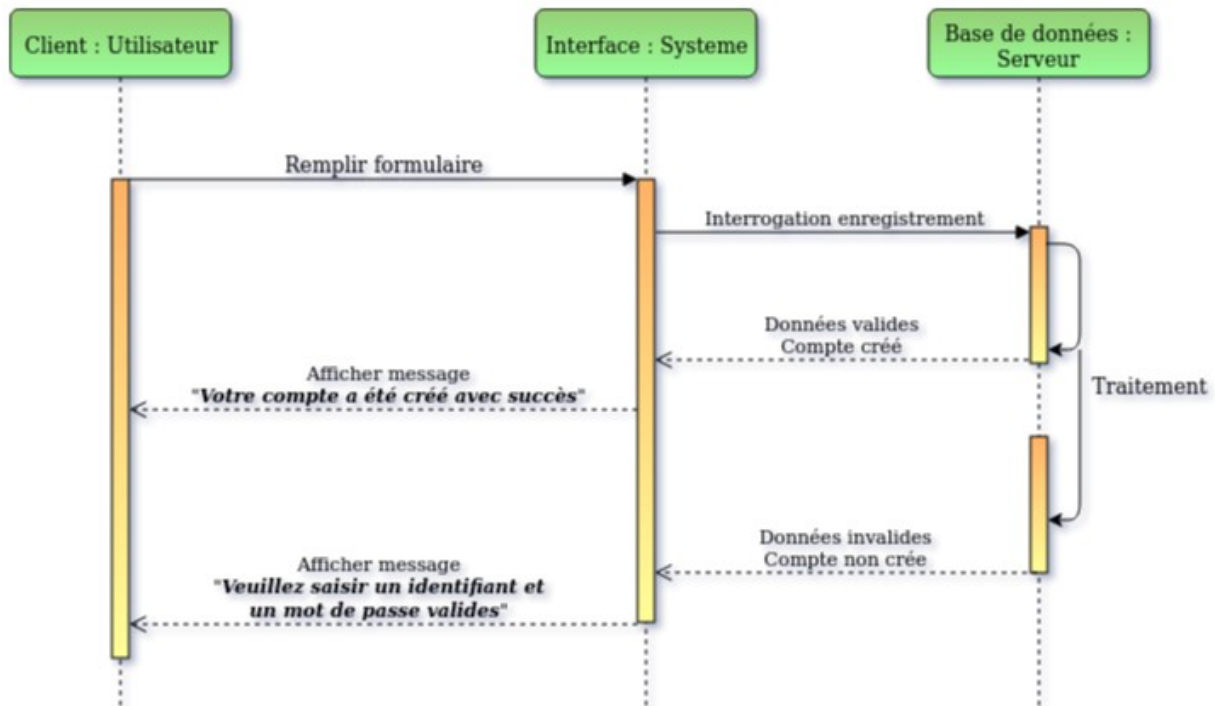


#### 3.2.2.1 **DESCRIPTION**

- Le client souhaite s'authentifier
- Le client entre son identifiant et son mot de passe
  - ✓ Les informations sont enregistrées dans la base de données, sa session est ouverte
  - ✗ Les informations ne sont pas enregistrées dans la base de données, il doit les resaisir
- Le client est connecté et peut passer sa commande
- Authentification terminée

### 3.3 PARTIE : CREATION DE COMPTE

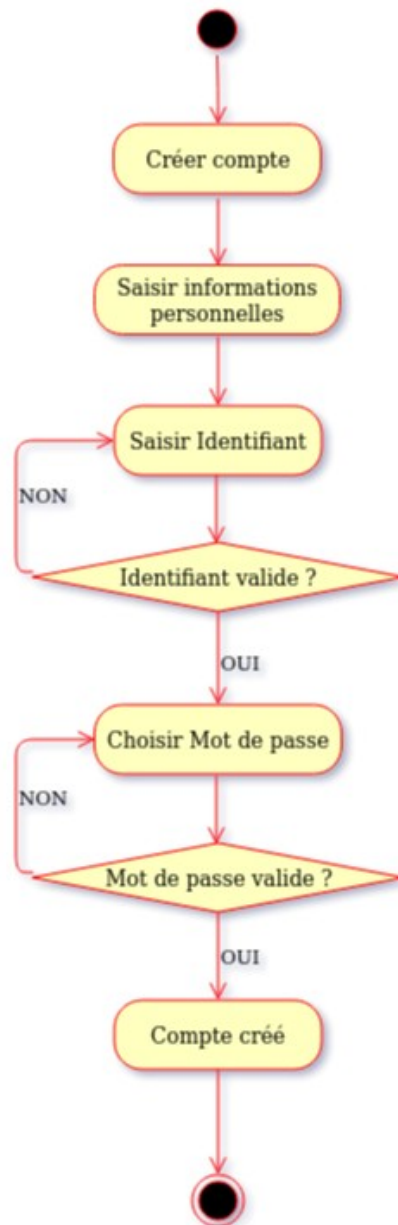
#### 3.3.1 DIAGRAMME DE SEQUENCE



##### 3.3.1.1 DESCRIPTION

- Le client souhaite créer un compte et remplit le formulaire
- Le système interroge la base de données pour enregistrer le compte
  - ✓ Les données Identifiant/Mot de passe sont valides, le compte est créé
    - Le système envoie un message indiquant au client que le compte est créé
  - ✗ Les données Identifiant/Mot de passe ne sont pas valides, le compte n'est pas créé
    - Le système envoie un message invitant le client à saisir des données valides

### 3.3.2 **DIAGRAMME D'ACTIVITE**



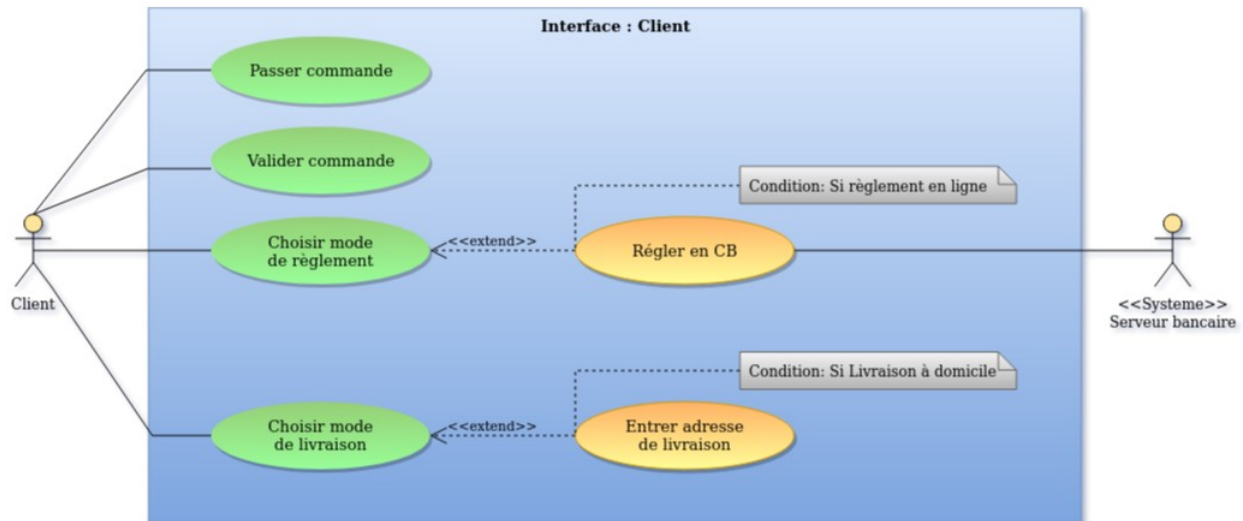
#### 3.3.2.1 **DESCRIPTION**

- Le client souhaite créer un compte
- Le client saisit ses informations personnelles (Nom, prénom, adresse de livraison)
- Le client entre son adresse mail comme identifiant
  - ✓ L'identifiant est une adresse mail valide, il peut choisir son mot de passe
  - ✗ L'identifiant n'est pas une adresse mail valide, il doit saisir une adresse mail valide
- Le client choisit son mot de passe (alphanumérique – 8 caractères minimum)
  - ✓ Le mot de passe est valide, le client peut se connecter
  - ✗ Le mot de passe n'est pas valide, il doit choisir un mot de passe valide
- Le compte client est créé et enregistré dans la base de données
- Création de compte terminée

## 4. GESTION DES COMMANDES

### 4.1 PARTIE : CLIENT

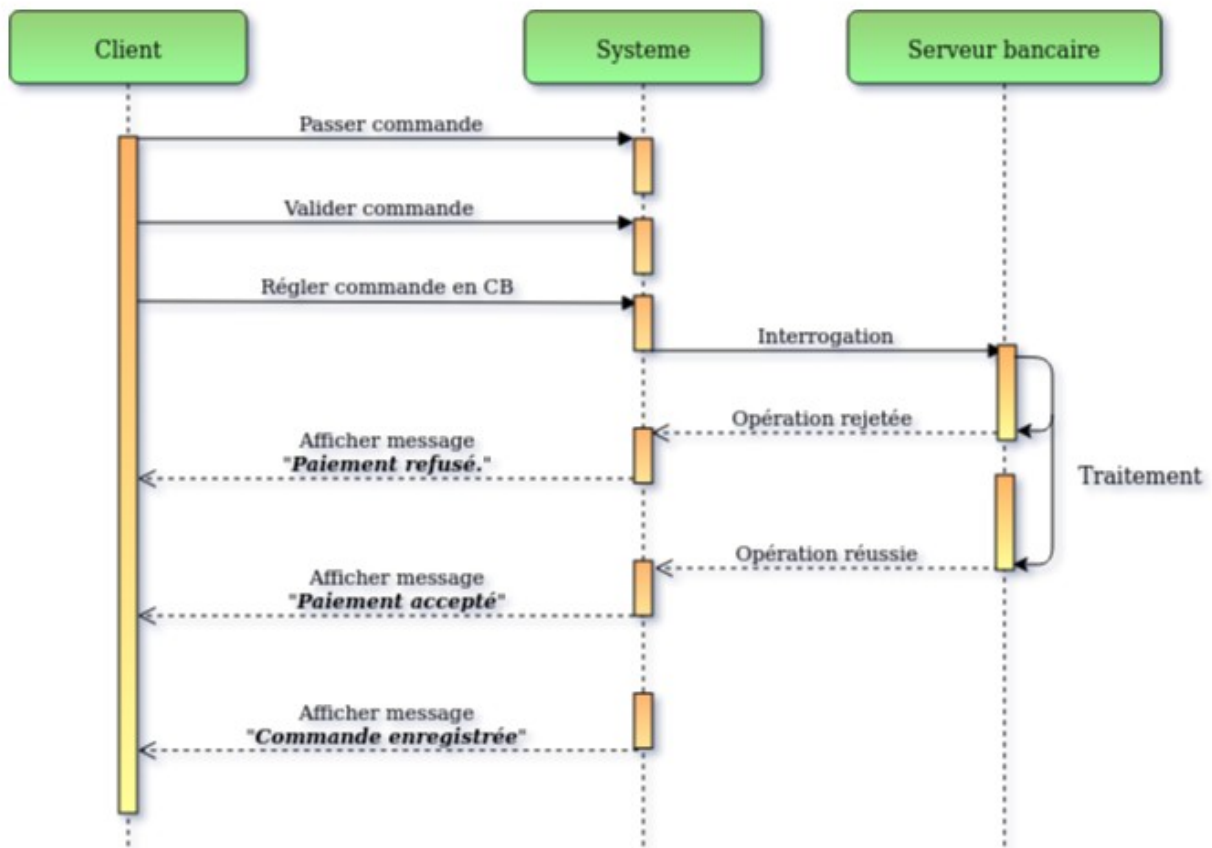
#### 4.1.1 DIAGRAMME DE CAS D'UTILISATION



##### 4.1.1.1 DESCRIPTION

- *Le client passe une commande*
  - Le client consulte le catalogue des pizzas
  - Le client choisit ses pizzas puis passe à l'étape suivante
- *Le client valide sa commande*
- *Le client choisit son mode de règlement*
  - Le client choisit le mode de règlement par CB, le système envoie la requête au serveur bancaire
  - Le client choisit le règlement à la livraison, la transaction se fera en espèces avec le livreur
- *Le client choisit son mode de livraison*
  - Le client peut choisir la livraison à domicile, il entre donc une adresse postale
  - Le client peut choisir la livraison à emporter

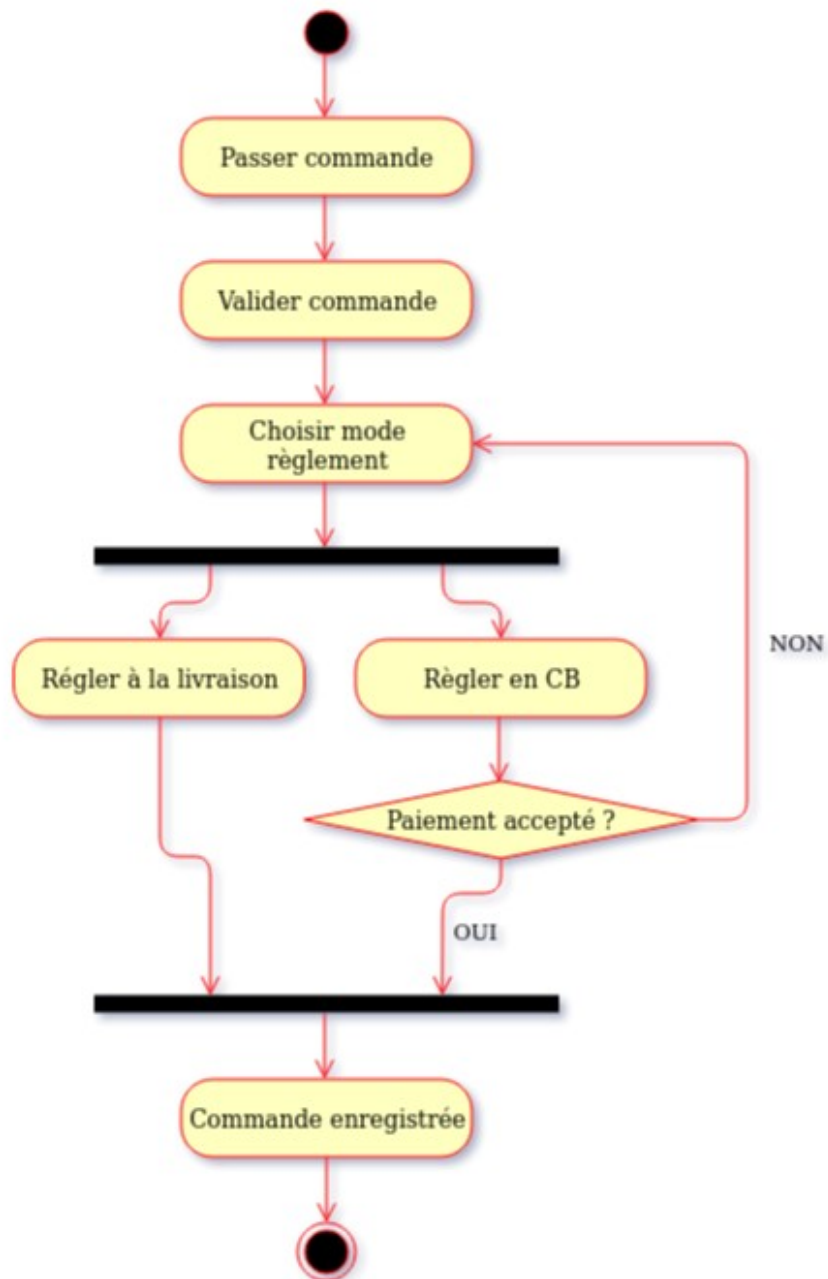
#### 4.1.2 DIAGRAMME DE SEQUENCE



##### 4.1.2.1 DESCRIPTION

- Le client passe sa commande
- Le client valide sa commande
- Le client choisit le règlement en CB
  - Le système interroge le serveur bancaire
    - ✗ L'opération n'a pas aboutie, le système affiche un message d'information
    - ✓ L'opération a aboutie, le système affiche un message d'information
- La commande a été enregistrée, le système affiche un message d'information

#### 4.1.3 **DIAGRAMME D'ACTIVITE**



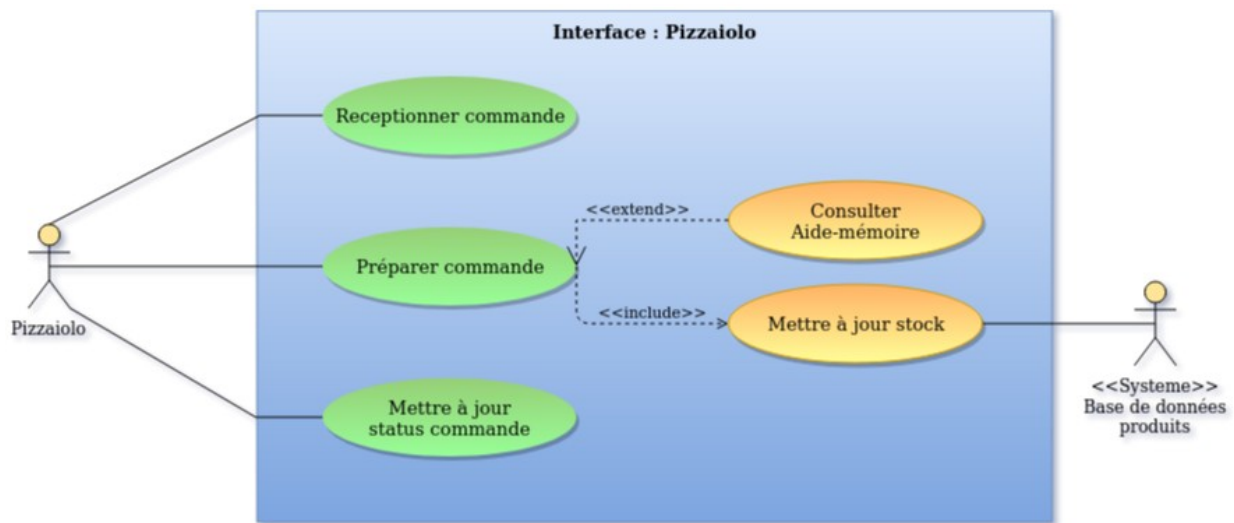
##### 4.1.3.1 **DESCRIPTION**

- Le client passe sa commande
- Le client valide sa commande
- *Le client choisit le règlement en CB*
  - Le système interroge le serveur bancaire
    - ✗ L'opération n'a pas réussie, le client est invité à choisir le règlement à la livraison
    - ✓ L'opération a réussie, la commande est validée et enregistrée
- *Le client choisit le règlement à la livraison*
  - ✓ La commande est automatiquement validée et enregistrée
- Commande terminée



## 4.2 PARTIE : PIZZAIOLO

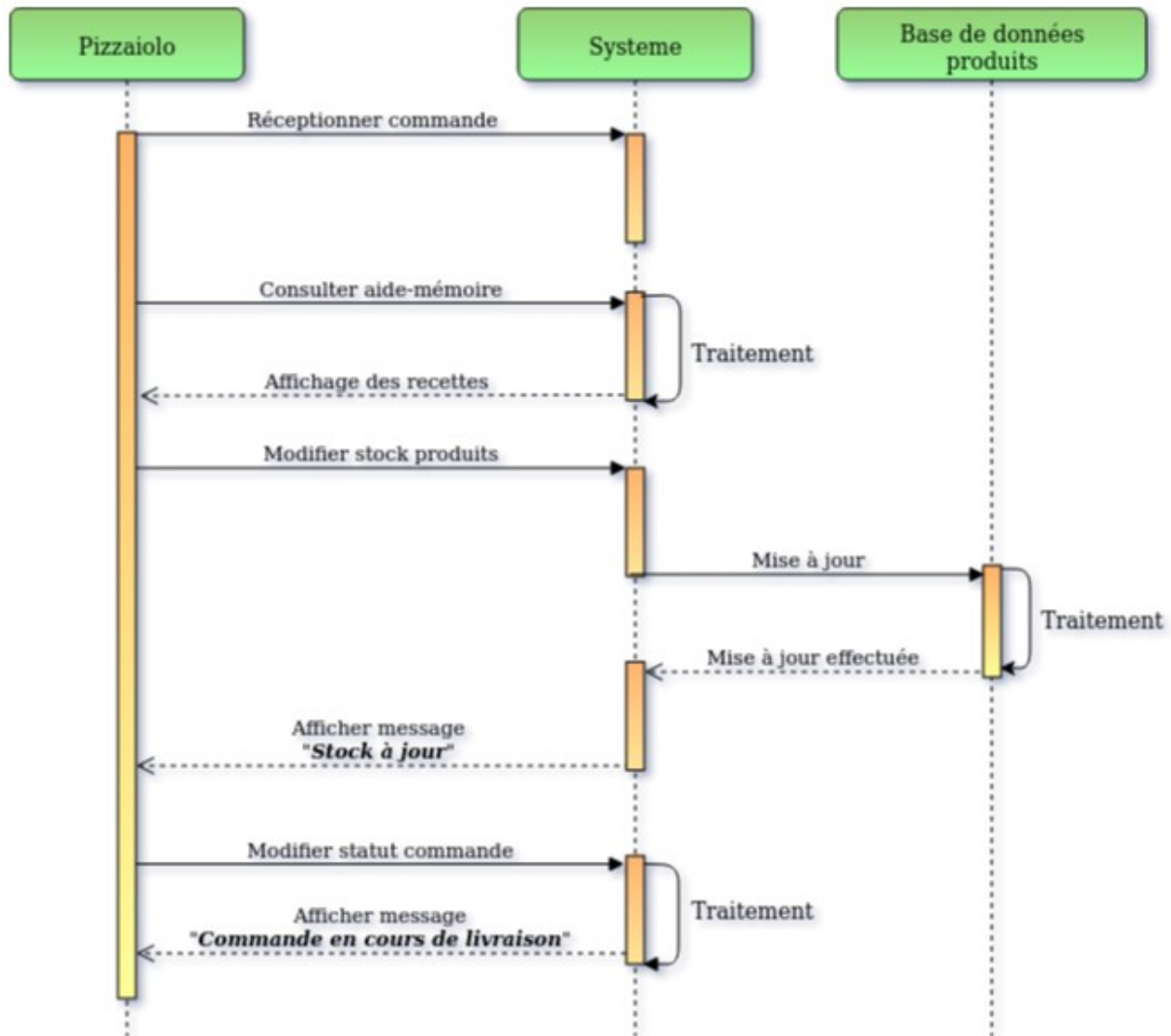
### 4.2.1 DIAGRAMME DE CAS D'UTILISATION



#### 4.2.2.1 DESCRIPTION

- Le pizzaiolo réceptionne la commande
- *Le Pizzaiolo prépare la commande*
  - Il peut consulter l'aide-mémoire, si nécessaire
  - Une fois la commande terminée, il met le stock produits à jour
- Le pizzaiolo modifie le statut de la commande quand elle est terminée et prête à être livrée

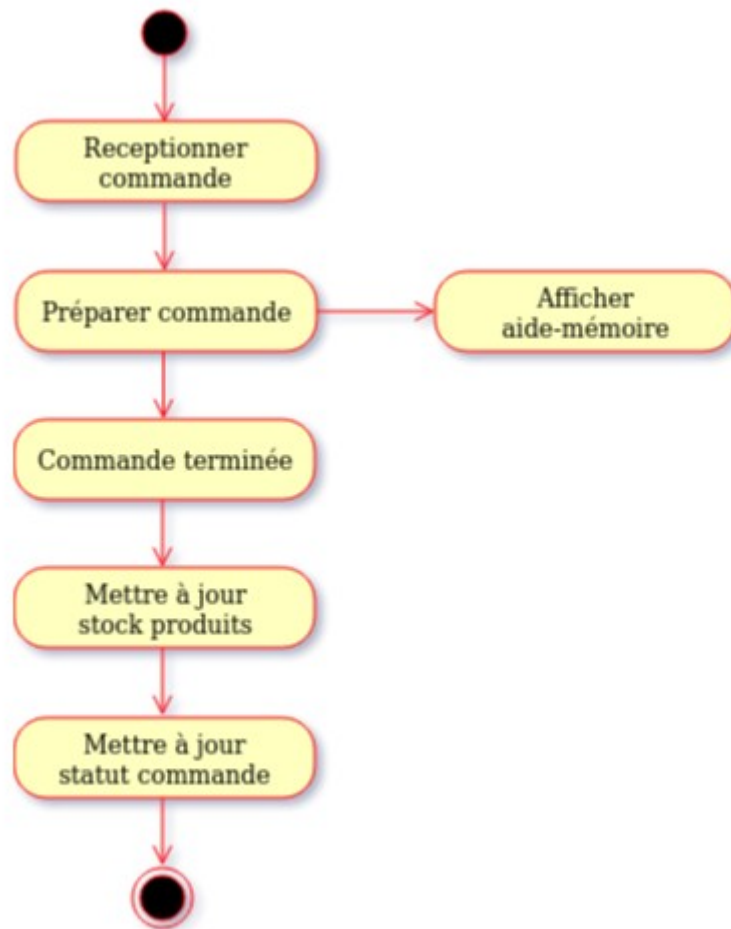
#### 4.2.2 DIAGRAMME DE SEQUENCE



##### 4.2.2.1 DESCRIPTION

- Le pizzaiolo réceptionne la commande
- *Le pizzaiolo prépare la commande*
  - Le pizzaiolo peut consulter l'aide-mémoire, si nécessaire
    - Le système affiche les recettes des pizzas
  - Une fois la commande terminée, il met le stock produits à jour
    - Le système envoie la requête à la base de données
    - La base de données met le stock à jour
    - Le système affiche un message informant que la mise à jour a été effectuée
- *Le pizzaiolo modifie le statut de la commande quand elle est terminée et prête à être livrée*
  - Le système affiche un message indiquant le nouveau statut de la commande

#### 4.2.3 **DIAGRAMME D'ACTIVITE**

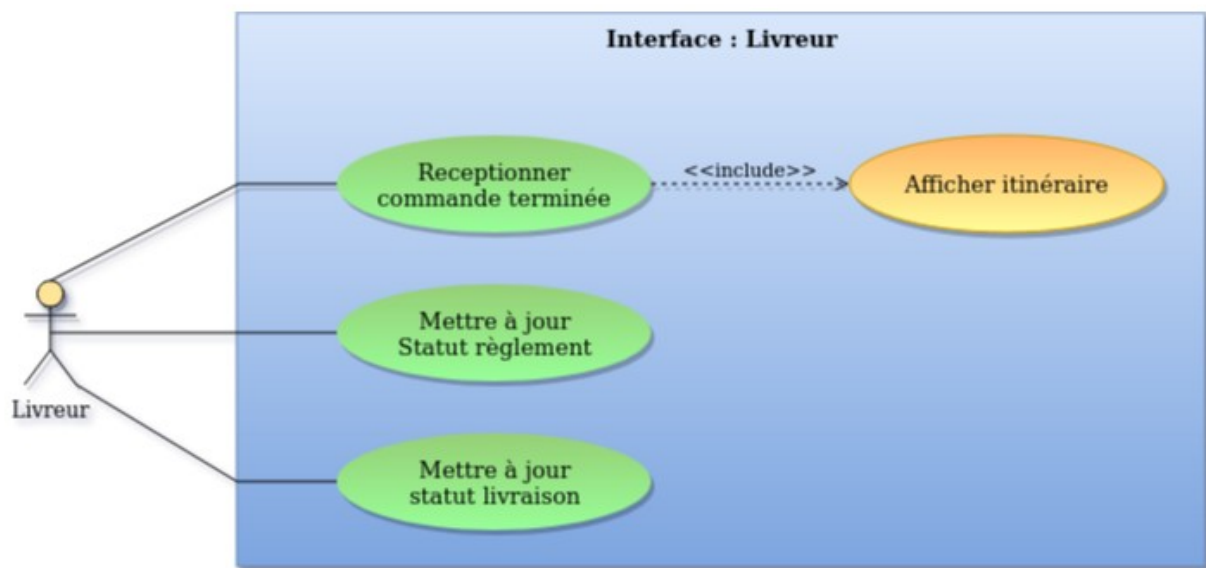


##### 4.2.3.1 **DESCRIPTION**

- Le pizzaiolo réceptionne la commande
- Le pizzaiolo peut consulter l'aide-mémoire, si nécessaire
- Le pizzaiolo prépare la commande
- La commande est terminée
- Le pizzaiolo met le stock produits à jour
- Le pizzaiolo met le statut de la commande à jour : Elle peut être livrée
- Préparation commande terminée

### 4.3 PARTIE : LIVREUR

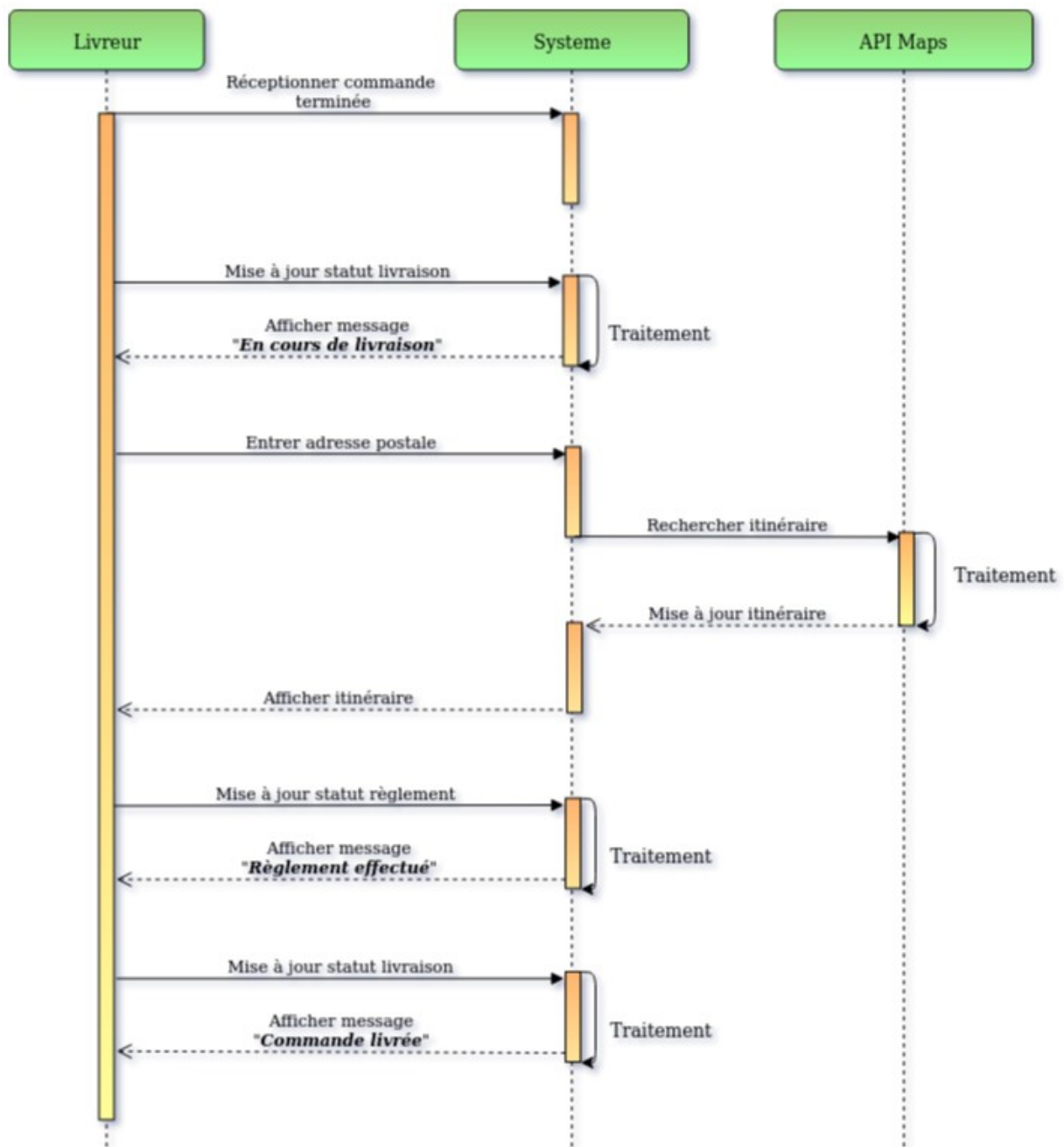
#### 4.3.1 DIAGRAMME DE CAS D'UTILISATION



##### 4.3.1.1 DESCRIPTION

- Le livreur réceptionne la commande terminée
  - Le livreur affiche les coordonnées de livraison dans l'API Maps
  - Le livreur met le statut de la commande à jour : **"En cours de livraison"**
- Le livreur met le statut de règlement à jour, si règlement à la livraison
- Le livreur met le statut de la commande à jour : **"Commande livrée"**

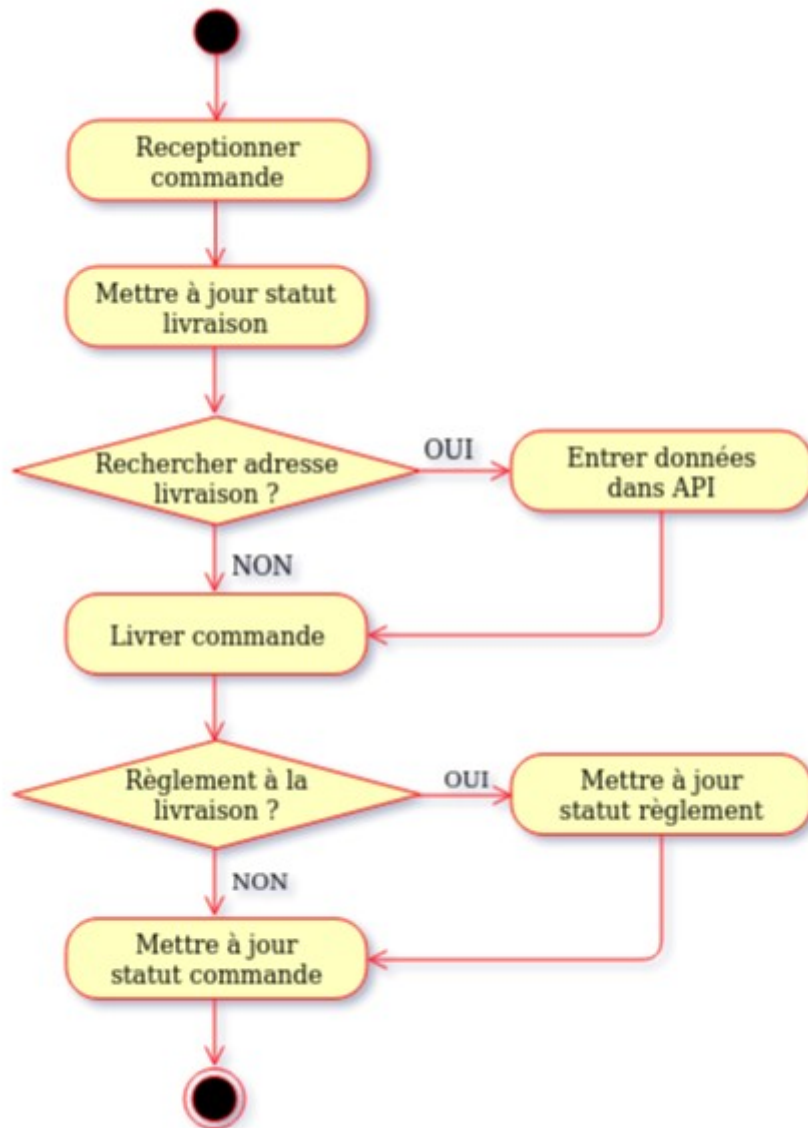
#### 4.3.2 DIAGRAMME DE SEQUENCE



#### 4.3.2.1 **DESCRIPTION**

- Le livreur réceptionne la commande terminée
- *Le livreur met le statut de la livraison à jour*
  - Le système affiche le message : “**En cours de livraison**”
- *Le livreur passe à l'étape de livraison de la commande*
  - Le livreur saisit l'adresse de livraison
    - Le système envoie une requête au serveur MAPS
    - Le serveur MAPS met à jour l'itinéraire
    - Le système affiche l'itinéraire
  - Si le règlement à la livraison a été choisi, le livreur encaisse les espèces
    - Le livreur met le statut du règlement à jour
    - Le système affiche le message : “**Règlement effectué**”
- *Le livreur met le statut de la livraison à jour*
  - Le système affiche le message : “**Commande livrée**”

#### 4.3.3 DIAGRAMME D'ACTIVITE

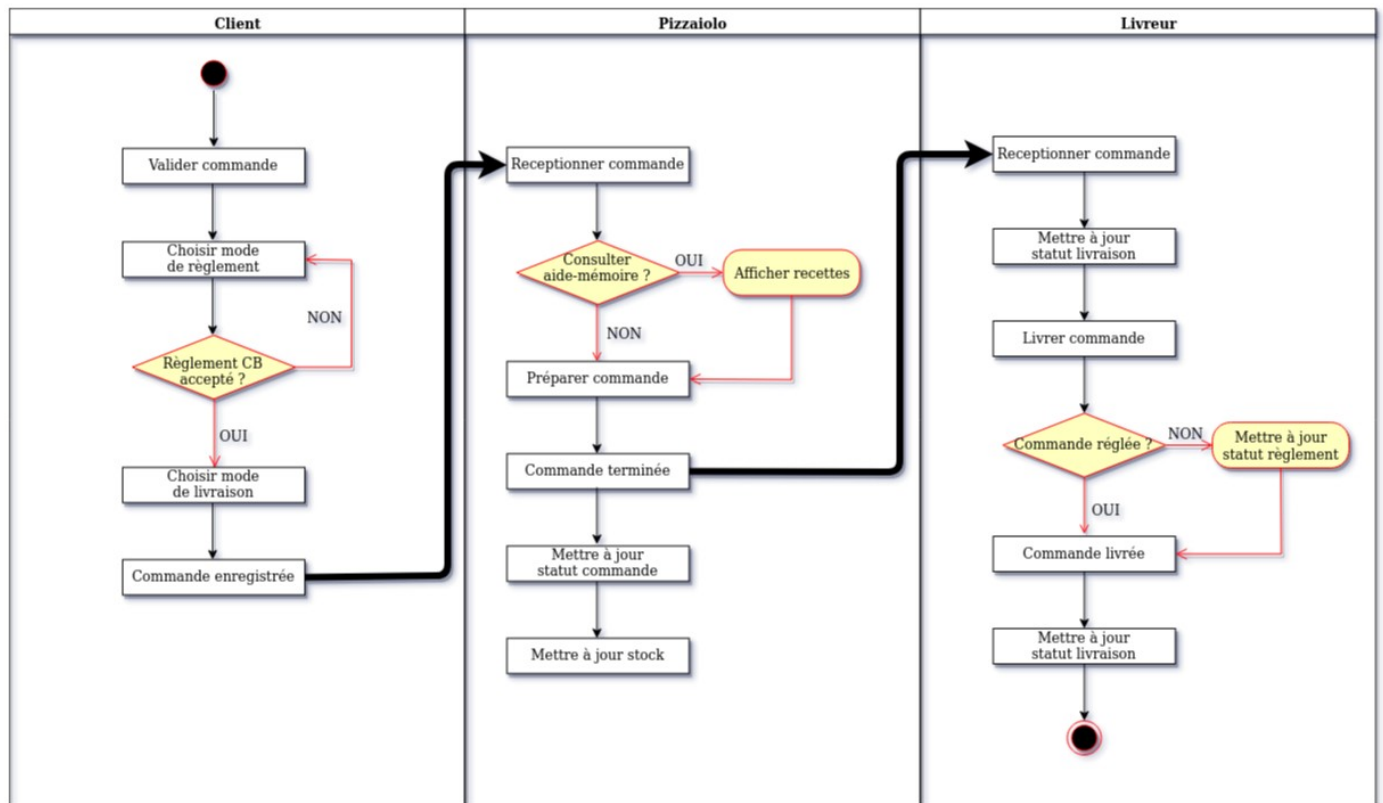


##### 4.3.3.1 DESCRIPTION

- Le livreur réceptionne la commande terminée
- Le livreur met le statut de la livraison à jour
- *Le livreur passe à l'étape de livraison de la commande*
  - Si nécessaire, le livreur entre les coordonnées de livraison dans l'API
  - Le livreur livre la commande
  - Si le règlement à la livraison a été choisi, le livreur met le statut du règlement à jour
- Le livreur met le statut de la livraison à jour
- Livraison terminée

## 5. CYCLE DE VIE D'UNE COMMANDE

### 5.1 DIAGRAMME





### 5.1.1 DESCRIPTION

#### DEBUT DE CYCLE DE VIE DE LA COMMANDE

##### Le CLIENT :

- Le client valide sa commande
- Le choisit son mode de règlement, s'il choisit le règlement par CB
  - Le règlement est accepté, la commande se poursuit
  - Le règlement n'est pas accepté, il doit choisir un autre mode de règlement
- Le client choisit son mode de livraison
- La commande est enregistrée et est envoyée vers le pizzaiolo

##### Le PIZZAIOLO :

- Le pizzaiolo réceptionne la commande
- Le pizzaiolo peut consulter l'aide-mémoire contenant les recettes
- Le pizzaiolo a terminé la commande, elle est envoyée à la livraison
- Le pizzaiolo met le statut de la commande à jour
- Le pizzaiolo met les tock à jour

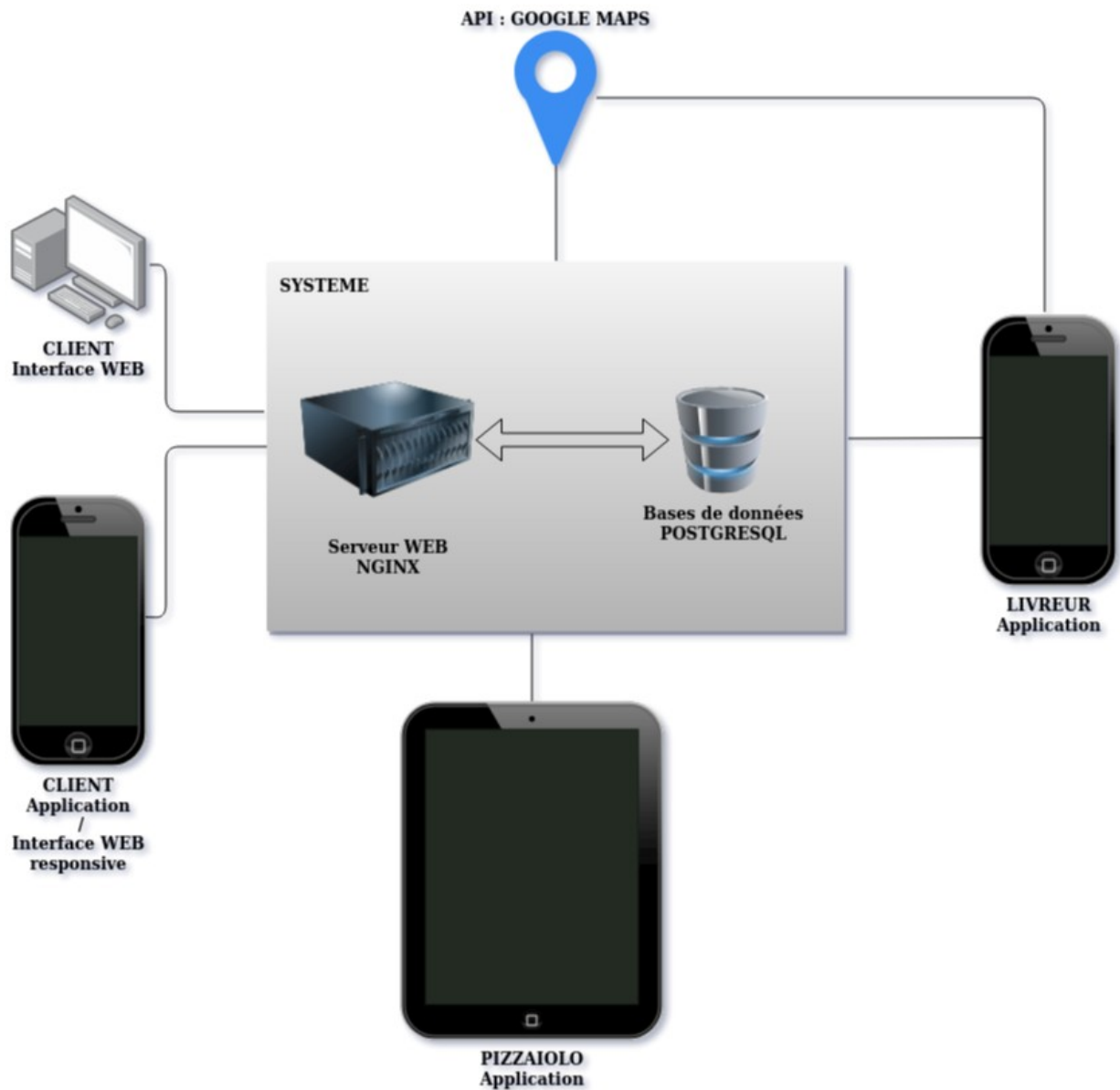
##### Le LIVREUR :

- Le livreur réceptionne la commande
- Le livreur met à jour le statut de la commande
- Le livreur livre la commande
- La commande a été réglée en ligne
  - Le livreur finalise la livraison et met le statut de la commande jour
- La commande n'a pas été réglée en ligne
  - Le livreur encaisse le règlement
  - Le livreur met le statut du règlement à jour
  - Le livreur finalise la livraison et met le statut de la commande jour

#### CYCLE DE VIE DE LA COMMANDE TERMINE

## 6. SOLUTION TECHNIQUE

### 6.1 *DIAGRAMME*



### 6.1.1 DESCRIPTION

#### LANGAGE ET FRAMEWORKS :

L'application WEB sera développée avec **PYTHON 3.8** et son framework **DJANGO** en version **3.0**.

La partie Interface Utilisateur sera développée avec **REACT 16.13**.

Cette bibliothèque Javascript et très complète permet la création de composants réutilisables, ce qui évitera de les réécrire à chaque fois que l'on en aura besoin, ce qui fera gagner un temps précieux.

Elle s'intégrera parfaitement dans le système et pourra être utilisée pour l'application mobile grâce à **REACT NATIVE**.

#### API MAPS :

L'API choisie pour les itinéraires de livraison sera celui de **GOOGLE MAPS**. Il est complet, facile d'utilisation et s'intègre parfaitement à l'application.

#### BASES DE DONNEES :

**POSTGRESQL** sera utilisé pour la gestion des bases de données (clients et produits).

Ce système est stable et très bien adapté à Python. Il est l'un des systèmes les plus avancés sur les normes SQL.

Son optimisation de ressources en cas de faibles charges est estimée entre 40% et 60% ce qui en fait un choix économique très intéressant et judicieux.

#### SERVEUR WEB :

Le serveur WEB sera administré avec **NGINX**. Il est le serveur WEB le plus utilisé au monde.

Comme **POSTGRESQL**, il est optimisé pour les cas de faibles charges grâce à sa structure de données minimales favorisant l'utilisation des caches ou du processeur.

Un point particulièrement intéressant est la possibilité de mettre à jour les fichiers de configuration sans avoir à redémarrer le serveur grâce à un processus maître gérant les processus serveur.

#### APPAREILS :

L'utilisateur pourra accéder à l'interface WEB depuis son ordinateur, sa tablette ou son smartphone.

Le pizzaiolo sera doté d'une tablette pour interagir plus rapidement avec l'application.

Le livreur sera, lui, doté d'un smartphone pour pouvoir afficher l'itinéraire avec un bon confort visuel et interagir avec l'application.

Les appareils des employés seront de marque Apple, l'application mobile sera donc installée nativement sur le système **iOS** et sera développée avec **SWIFT**.

L'application WEB affichée dans le navigateur, sera accessible depuis n'importe quel appareil (ordinateurs, smartphones, tablettes), cependant, elle devra être "**responsive**" pour s'adapter aux petites diagonales des tablettes ou smartphones.

Il est à noter que pour une utilisation, côté client, plus ergonomique sur ces appareils, il serait préférable de développer une application mobile **iOS**.