

# OC PIZZA

## Nouveau Système Informatique

Dossier de conception fonctionnelle

Version 1.0

**Auteur**

KhadijaCHTIOUI

*Analyste Programmeur*

## TABLE DES MATIERES

<b>1 - Versions .....</b>	<b>3</b>
<b>2 - Introduction.....</b>	<b>4</b>
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
<b>3 - Description générale de la solution .....</b>	<b>6</b>
3.1 - Les principes de fonctionnement .....	6
3.1.1 – Interface Client (Front Office) .....	6
3.1.2 – Interface Équipe Pizzeria (Back Office).....	6
3.2 - Les acteurs .....	6
3.2.1 - Diagramme de Contexte : .....	7
3.2.2 - Impact mapping : .....	8
3.3 - Les cas d'utilisation généraux.....	9
3.3.1 - Diagramme cas d'utilisation Client :.....	9
3.3.2 - Gestion de préparation d'une commande : .....	10
3.3.3 - Diagramme de gestion administrative : .....	11
3.3.4 - Les scénarios : .....	12
3.3.4.1 - Scenario authentication : .....	12
3.3.4.2 - Scenario prise des commandes : .....	13
3.3.4.3 - Scenario gestion administrative : .....	14
3.3.4.4 - Scenario préparation d'une commande : .....	15
<b>4 - Le domaine fonctionnel.....</b>	<b>16</b>
4.1 - Référentiel.....	16
4.1.1 - Diagramme de classes .....	16
4.2 - Package .....	17
4.2.1 - Diagramme de package : .....	18
<b>5 - Les workflows.....</b>	<b>19</b>
5.1 - Le workflow d'une commande .....	19
5.1.1 - Diagramme D'activité : .....	19
5.1.2 - Diagramme d'état- transition : .....	20
<b>6 - Application Web &amp; Mobile.....</b>	<b>21</b>
6.1 - Les acteurs .....	21
6.1.1 - Client : .....	22
6.1.2 - Directeur : .....	22
6.1.3 - Livreur : .....	23
6.1.4 - Pizzaiolos : .....	23
6.2 - Les règles de gestion générales.....	23
<b>7 - Glossaire .....</b>	<b>24</b>

# 1 - VERSIONS

Auteur	Date	Description	Version
Kh.CHTIOUI	01/03/2022	Création du document	1.0

## 2 - INTRODUCTION

### 2.1 - Objet du document

Le présent document constitue le dossier de conception fonctionnelle de l'application OC Pizza.

#### **Objectif du document :**

Présenter les fonctionnalités du Système « OC Pizza » telles qu'elles ont pu être définies avec le client lors des échanges et réunions préparatoires.

Les éléments du présent dossier découlent :

- De la présentation au client des spécifications fonctionnelles lors de l'entrevue de lancement du projet
- Du recueil des besoins et du cahier des charges donné par le client
- De la présentation de la solution technique

### 2.2 - Références

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

1. **DCT – 1.0** : Dossier de conception technique de l'application
2. **DE – 1.0** : Dossier d'exploitation de l'application

### 2.3 - Besoin du client

#### *2.3.1 - Contexte*

Le groupe OC Pizza est une chaîne de pizzerias en plein essor et 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 la fin de l'année. Le système informatique actuel ne répond plus aux besoins du client.

#### *2.3.2 - Enjeux et Objectifs*

Prendre en compte les demandes du client, afin de répondre à ces attentes.

Créer une application avec une interface qui soit la plus lisible, compréhensible et facile d'utilisation.

Le système informatique déployé dans toutes les pizzerias devra lui permettre :

- D'être plus efficace dans la gestion des commandes, de leur réception à leur livraison en passant par leurs préparations ;
- De suivre en temps réel les commandes passées et en préparation ;

- De suivre en temps réel le stock d'ingrédients restants pour savoir quelles pizzas sont encore réalisables ;
- De proposer un site internet pour que les clients puissent :
  - Passer leurs commandes, en plus de la prise de commande par téléphone ou sur place
  - Payer en ligne leur commande s'il le souhaite, sinon, ils paieront directement à la livraison
  - Modifier ou annuler les commandes tant que celle-ci n'a pas été préparée
- De proposer un aide-mémoire aux pizzaiolos indiquant la recette de chaque pizza
- D'informer ou notifier les clients sur l'état de leur commande

## 3 - DESCRIPTION GENERALE DE LA SOLUTION

La solution proposée sera donc composée d'un site Web et d'une application disponible sous Android.

La solution respecte le responsive de manière à s'adapter à tous types d'écrans.

### 3.1 - Les principes de fonctionnement

Le fonctionnement sera adapté en fonction du type d'utilisateur, nous aurons donc une interface Client et une Employé.

Les clients comme les employés se connectent à l'application via leur identifiants (identifiant et mot de passe).

Les clients peuvent consulter le site, cependant il est nécessaire de s'identifier afin de pouvoir valider une commande

#### 3.1.1 – Interface Client (Front Office)

Le client est automatiquement dirigé sur le pizzeria de référence en fonction de sa localisation.

Il peut choisir une autre pizzeria s'il le souhaite en arrivant sur le site.

Le client Ayant accès à la gestion de son compte il peut changer ses informations (son adresse, son mot de passe, sa pizzeria, supprimer son compte, ...)

Le client a accès au Menu avec les produits encore disponibles dans le stock.

Une fois son panier complet il a le choix du type de commande (En Livraison, A Emporter) et du type de règlement (en ligne ou sur place).

Après la validation de son panier, il accède au suivi de sa commande.

Tant que la commande n'est pas en préparation, il peut la modifier ou l'annuler

#### 3.1.2 – Interface Équipe Pizzeria (Back Office)

Le gérant est le seul à avoir la gestion des comptes employés et le suivi global du groupe OC Pizza.

Les accès des membres de l'équipe sont limités à leur pizzeria de rattachement. Il existe différents niveaux d'accès selon les postes des membres de l'équipe.

L'équipe du pizzeria doit être connecté pour accéder à la partie gestion.

Le membre de la pizzeria va faire évoluer le suivi d'une commande à chaque étape,

### 3.2 - Les acteurs

**Gérant :** Il accède sur l'application à la gestion des employés mais également aux stocks de tous les points de vente. Ils ont besoin d'une vision globale.

**Client** : Il utilise l'application pour les parties authentification, consultation, vente des pizzas et suivi de la commande.

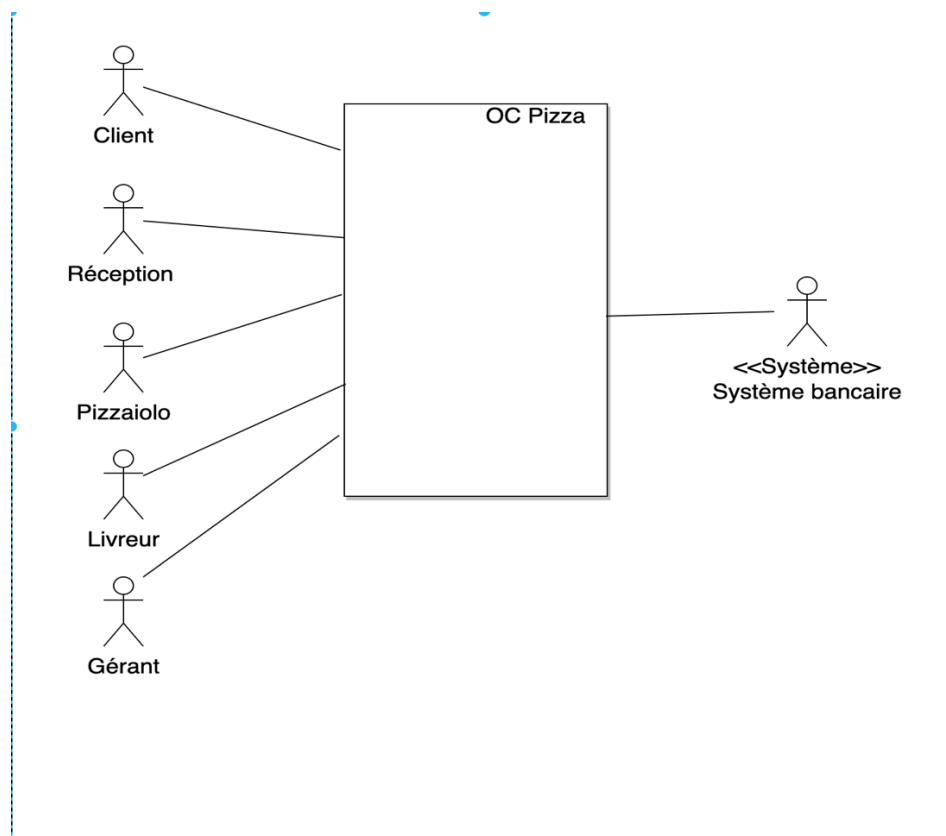
**Pizzaiolo** : Il utilise l'application pour consulter les recettes, les stocks et pour consulter les pizzas en préparation et la validation de celles-ci en préparation et une fois préparées. Il peut également gérer l'encaissement des clients et la validation du règlement sur l'application.

**Livreur** : Il utilise l'application pour la localisation du client et son numéro de téléphone, il renseigne le suivi de la livraison et du paiement et confirmer la livraison des pizzas.

**Réception** : Il accède aux mêmes fonctionnalités que le pizzaiolo dans l'application et peut également consulter le stock d'ingrédients restants ainsi qu'en modifier les quantités.

**Système Bancaire** : un acteur secondaire qui assure la gestion du règlement des commandes.

### 3.2.1 - Diagramme de Contexte :

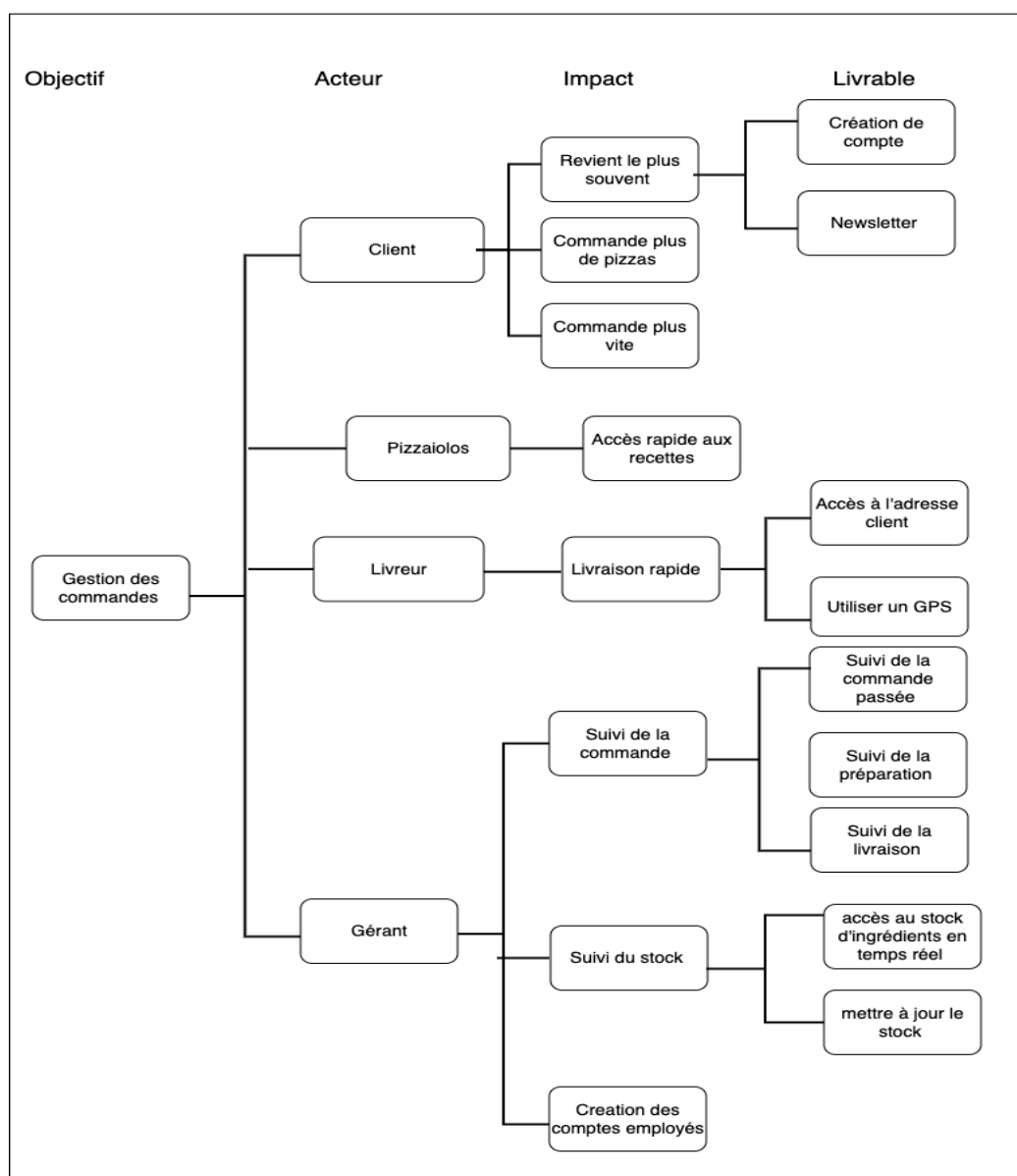


### 3.2.2 - Impact mapping :

Pour identifier les acteurs et fonctionnalités de notre projet nous avons utilisé un Impact Mapping.

L'Impact Mapping est construit comme suit :

- Définition de l'objectif principal
- Définition des acteurs majeurs
- Définition des différents impacts de ces acteurs
- Définition des fonctions des impacts





### 3.3 - Les cas d'utilisation généraux

Le diagramme des cas d'utilisations utilisés pour une représentation du comportement fonctionnel d'un système logiciel.

#### 3.3.1 - Diagramme cas d'utilisation Client :

Nous avons donc le cas d'utilisation pour le Front Office qui ne va concerner que les clients,

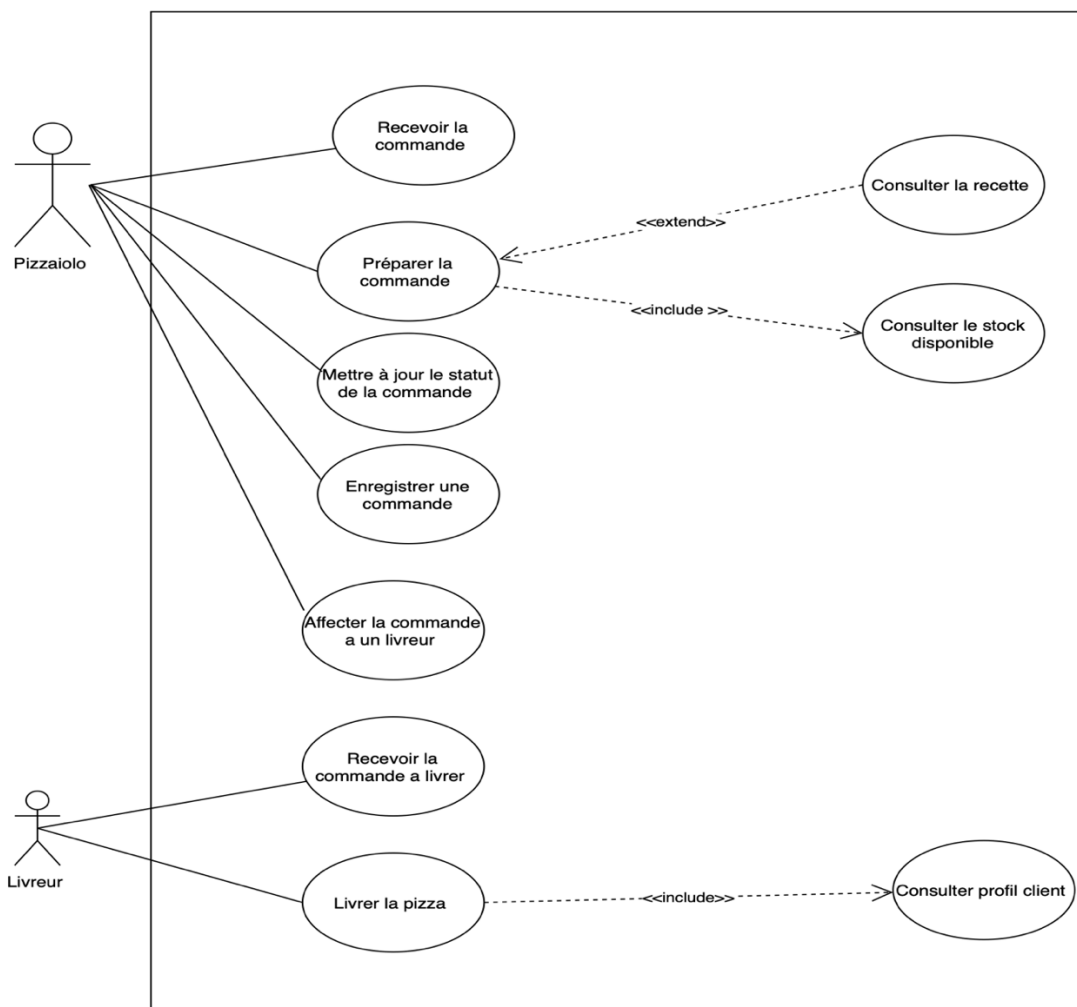


Le client peut :

1. S'inscrire et renseigner ses coordonnées.
2. S'authentifier en saisissant ses identifiants, avec sa localisation il sera dirigé vers la pizzeria à côté de lui.
3. Consulter le menu
4. Passer une commande en validant le panier la commande sera enregistrer.
5. Effectuer le paiement en cas de paiement en ligne.
6. Modifier ou annuler la commande.
7. Consulter la liste de commande.
8. Suivre le statut de la commande.
9. Règlement de la commande en cas de paiement a la livraison :
  - Paiement par chèque
  - Paiement par CB
  - Paiement en espèce

### 3.3.2 - Gestion de préparation d'une commande :

Ce diagramme représente de manière simplifiée le Back Office.



Pour la préparation de la commande nous avons le livreur et le pizzaiolo :

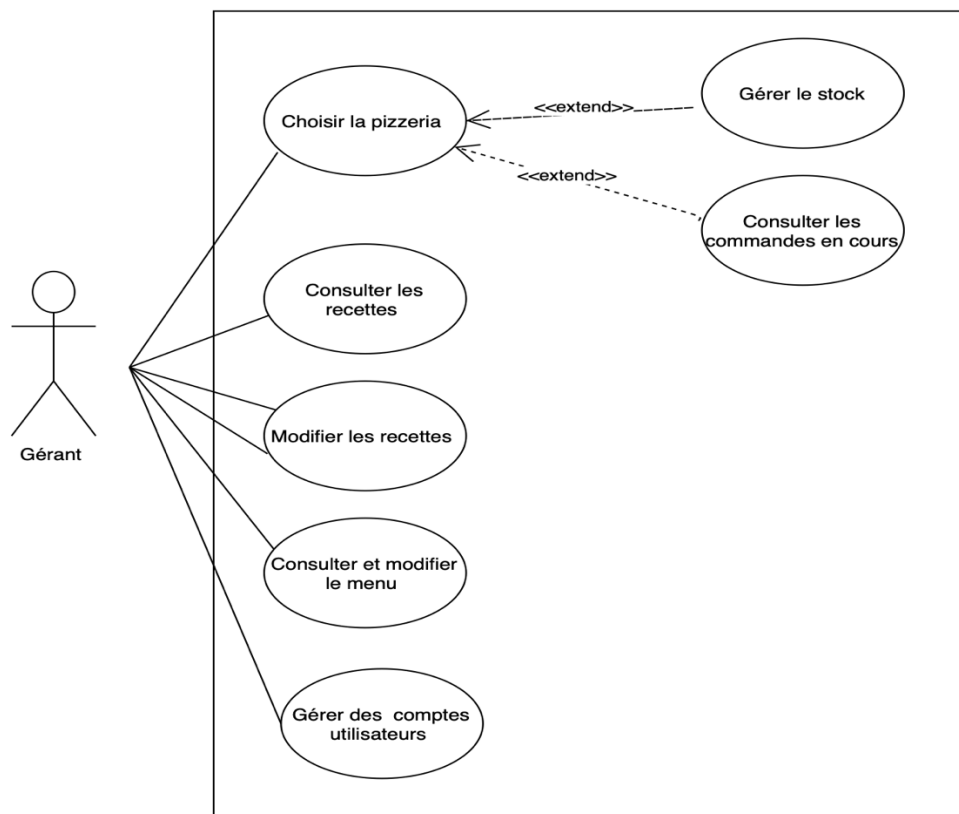
#### Le pizzaiolo :

- Recevoir la notification d'une nouvelle commande.
- Consulter le stock d'ingrédients restants.
- Consulter les recettes des pizzas.
- Mettre à jour le statut de la commande.
- Affecter la commande a un livreur.

#### Livreur :

- Recevoir la commande à livrer
- Consulter adresse et numéro de téléphone client.

### 3.3.3 - Diagramme de gestion administrative :



Le gérant peut :

- Choisir une pizzeria du groupe.

- Gérer le stock (Réaliser l'inventaire physique, Réconcilier le stock informatique et physique)
- Consulter les commandes en cours.
- Consulter les recettes
- Modifier les recettes
- Consulter et modifier les menus.
- Gérer des compte utilisateurs

### 3.3.4 - Les scénarios :

#### 3.3.4.1 - Scenario authentication :



**Nom :** authentication  
**Acteur :** client  
**Description :** le client doit s'inscrire ou se connecter pour pouvoir passer une commande  
**Auteur :** Khadija CHTIOUI  
**Pré-condition :** aucune  
**Démarrage :** le client visite la page d'authentification

#### Scenario nominal :

- 1\_ la page d'authentification affichée avec les choix (s'inscrire) et (se connecter)
- 2\_ le client clique sur s'inscrire
- 3\_ le client remplit son profil et choisit une pizzeria
- 4\_ le client valide son profil
- 5\_ le système enregistre les données du client
- 6\_ le client saisit ses identifiants
- 7\_ les identifiants validés par le système
- 8\_ le système affiche un message que la connexion est réussie et autorise l'accès au menu

#### Scenarios d'exception :

- 5\_ Le système refuse l'inscription car les identifiants existent déjà
- 7\_ les identifiants ne sont pas validés par le système
- 8\_ le système affiche un message que la connexion a échoué et n'autorise pas l'accès

### 3.3.4.2 - Scenario prise des commandes :

Nom : Prise des commandes  
Acteur : client  
Description : le client passe une commande  
Auteur : Khadija CHTIOUI  
Pré-condition : connexion obligatoire  
Démarrage : le client consulte le menu

**Scenario nominal :**

- 1 \_ le système affiche le menu
- 2 \_ le client consulte le menu
- 3 \_ le client fait son choix
- 4 \_ le client choisit livraison à domicile
- 5 \_ le client consulte le panier
- 6 \_ le client effectue le paiement bancaire
- 7 \_ le système affiche les pages des coordonnées bancaires
- 8 \_ le client saisit ses coordonnées bancaires
- 9 \_ le système valide les coordonnées saisies
- 10 \_ le système affiche un message qui confirme que le paiement est accepté

**Scenarios d'exception :**

- 9 \_ les coordonnées bancaires saisies ne sont pas validées par le système
- 10 \_ le système affiche un message disant que le paiement n'est pas passé

### 3.3.4.3 - Scenario gestion administrative :

Nom : Gestion administrative

Acteur : directeur

Description : le directeur gère les pizzerias

Auteur : Khadija CHTIOUI

Pré-condition : connexion obligatoire

Démarrage : le directeur consulte la page réservée à la direction

#### Scenario nominal :

- 1 \_ le système affiche toutes les pizzerias
- 2 \_ le directeur choisit une pizzeria
- 3 \_ le système affiche le menu (suivi des commandes), (suivi du stock), (gestion compte utilisateurs) ou (quitter)
- 4 \_ le directeur choisit (suivi des commandes)
- 5 \_ le système affiche la liste des commandes
- 6 \_ le directeur choisit une commande
- 7 \_ le système affiche toutes les informations sur la commande
- 8 \_ le directeur quitte la liste des commandes
- 9 \_ le système affiche le menu
- 10 \_ le directeur choisit (suivi du stock)
- 11 \_ le système affiche les informations du stock
- 12 \_ le directeur consulte le stock
- 13 \_ le directeur quitte le stock
- 14 \_ le système affiche le menu
- 15 \_ le directeur choisit (gestion des comptes)
- 16 \_ le système affiche les comptes clients
- 17 \_ le directeur clique sur un compte
- 18 \_ le système affiche les informations du compte
- 19 \_ le directeur quitte la page (gestion des comptes)
- 20 \_ le système affiche le menu
- 21 \_ le directeur clique sur (quitter)

#### Scenario alternatif :

- 12 \_ le directeur effectue des modifications sur le stock
- 12' \_ le système affiche un message de confirmation
- 12'' \_ le directeur confirme les modifications
- 19 \_ le directeur modifie le compte client
- 19' \_ le système envoie un message de validation
- 19'' \_ le directeur valide les modifications
- 19''' \_ le directeur crée un compte employé

**Scenario d'exception :**

#### 3.3.4.4 - *Scenario préparation d'une commande :*

Nom : Préparation d'une commande  
Acteur : pizzaiolo  
Description : le pizzaiolo prépare les pizzas  
Auteur : Khadija CHTIOUI  
Pré-condition : connexion obligatoire  
Démarrage : le pizzaiolo consulte les commandes et les recettes

**Scenario nominal :**

- 1 \_ le système affiche la liste des commandes à préparer
- 2 \_ le pizzaiolo clique sur la première commande sur la liste
- 3 \_ le système demande la confirmation que le pizzaiolo prend en charge la commande
- 4 \_ le pizzaiolo clique sur (confirmer)
- 5 \_ le système envoie une notification au client disant que la commande est en cours de préparation
- 6 \_ le pizzaiolo clique pour afficher la recette de la pizza
- 7 \_ le système affiche la recette
- 8 \_ le pizzaiolo appuie sur (commande terminée)
- 9 \_ le système envoie un message au client disant que la commande est en cours de livraison
- 10 \_ le système notifie le livreur d'une nouvelle commande à livrer

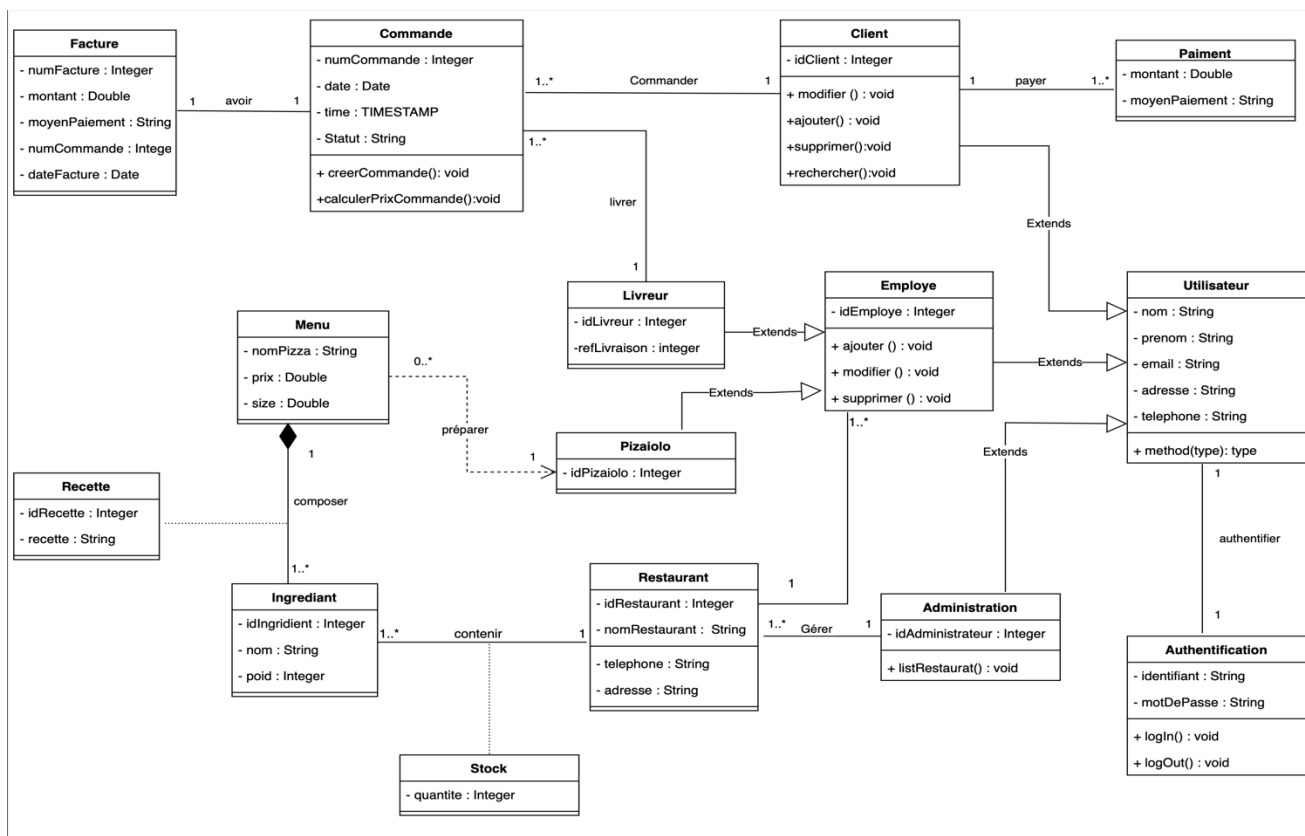
**Scenario d'exception :**

- 4 \_ le pizzaiolo affirme ne peut pas pouvoir faire la commande par manque d'ingrédients nécessaires

## 4 - LE DOMAINE FONCTIONNEL

### 4.1 - Référentiel

#### 4.1.1 - Diagramme de classes



Afin de modéliser ce domaine fonctionnel, nous avons élaboré un diagramme de classes qui va représenter les relations entre les différents concepts de notre application.

Chacun de ces objets est donc représenté par des classes possédant leurs spécificités qu'on nomme attributs :

- Les niveaux de visibilité des attributs (public, private, etc...)
- Les types d'attributs (String, Integer, boolean, etc...)
- Les associations (one to one, one to many, many to many, etc...)
- La multiplicité des associations
- Nom des associations

Le domaine fonctionnel va nous servir à la création de notre base de données PostgreSQL.



**Utilisateur :**

Les tables client, employé et administrateur hérite de la table utilisateur celle-ci représentent la gestion des utilisateurs On y retrouve les coordonnées des utilisateurs avec un rattachement à la table authentification.

Les tables Livreur et pizzaiolo hérite de la table Employe.

Les employés sont également associés au restaurant ou ils exercent leurs fonctions.

**Authentification :**

Elle comprend les identifiants de connexions des utilisateurs.

Elle est associée avec la table Utilisateur (un a un).

**Commandes :**

Gestion des commandes s'organise de la manière suivante.

La table principale 'Commande' regroupe les informations générales de la commande (numéro, date, etc...) :

- Elle est associée avec la table Facture (une commande a une seule facture et une facture a une seule commande).
- Elle est associée avec la table livreur (une commande et livrer par un livreur et un livreur peut livrer plusieurs commande).
- Elle est associée avec la table client (une commande est commandée par un client et un client peut commander plusieurs commande).
- La table client associée avec la table paiement (un client paie une a plusieurs commande et une commande est payée par un client).

**Administration :**

La table Administration est associée avec la table Restaurant (un administrateur peut gérer plusieurs restaurant et un restaurant peut être géré par un seul administrateur).

**Menu :**

La table Ingrédient est une composante de la table Menu, c'est deux tables est associées par la table Recette.

La table Menu est dépendante de la table pizzaiolo c'est le pizzaiolo qui prépare les menus.

**Stock :**

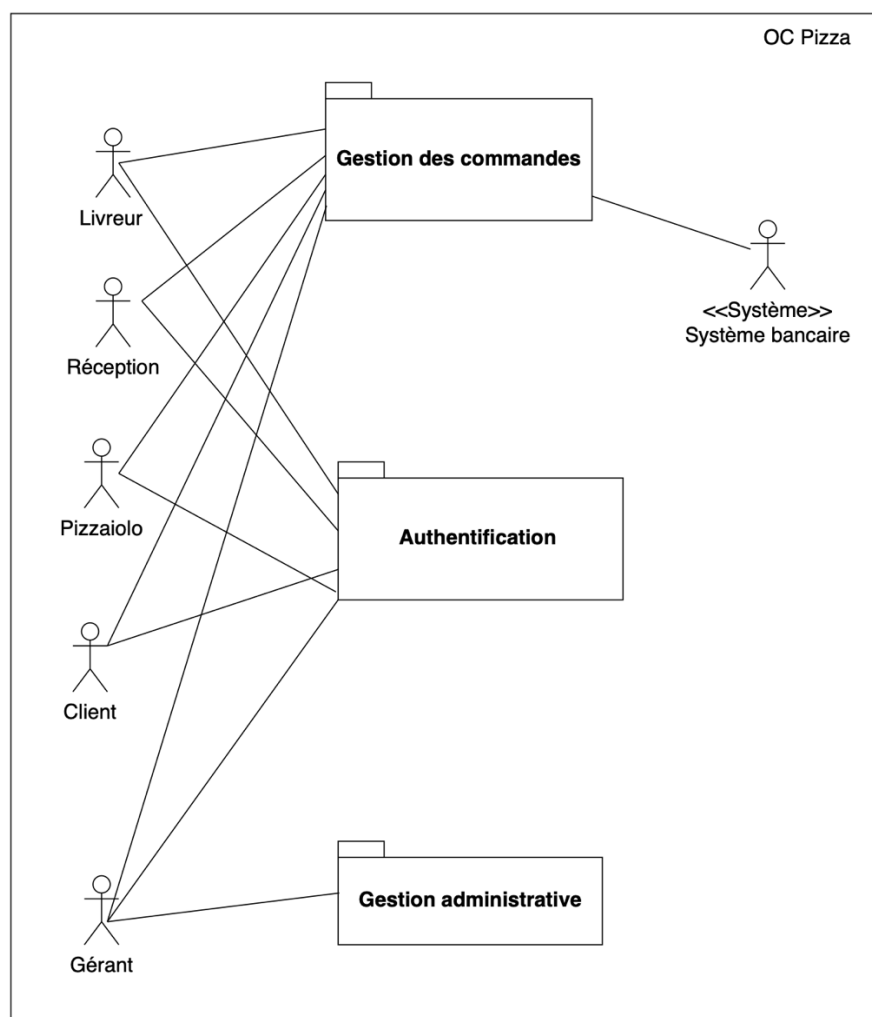
Le stock est une association entre la table Restaurant et la table Ingrédient.

Un restaurant contient un seul stock et le stock peut contenir un a plusieurs ingrédients.

## 4.2 - Package

Le découpage de l'application permettant de rassembler dans chaque package les éléments lui correspondant et de lui en attribuer ses différents acteurs.

#### 4.2.1 - Diagramme de package :

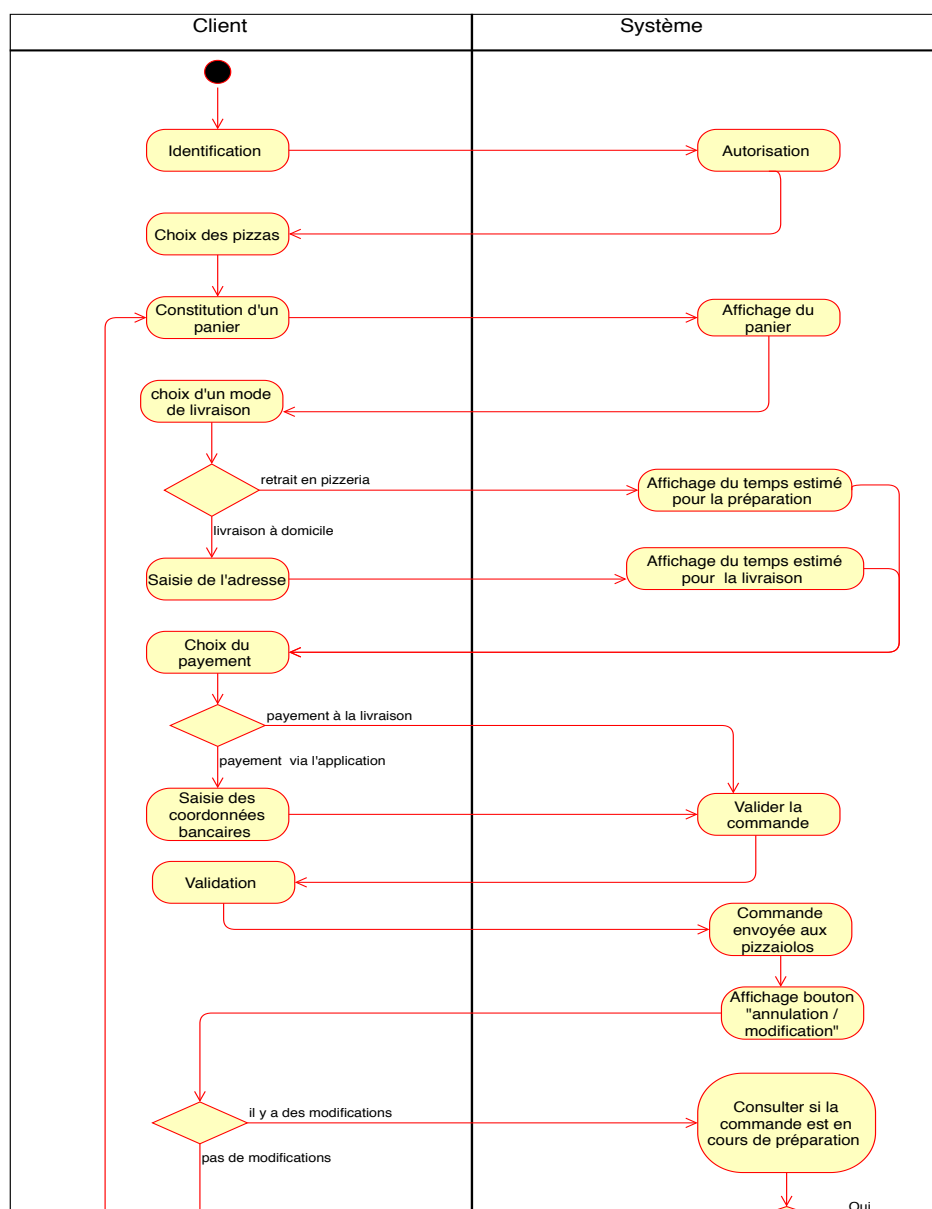


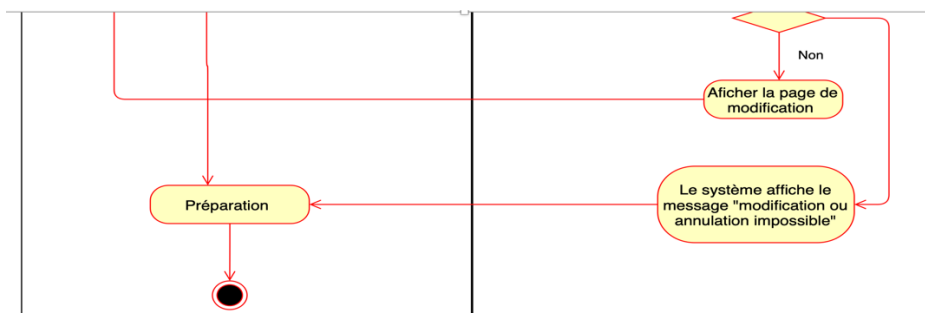
## 5 - LES WORKFLOWS

Nos workflows modélisent les flux de données lors des différentes interactions effectuées par le client ou le personnel de la Pizzeria. Nos workflows sont au nombre de quatre et détaillent chacun une situation précise

### 5.1 - Le workflow d'une commande

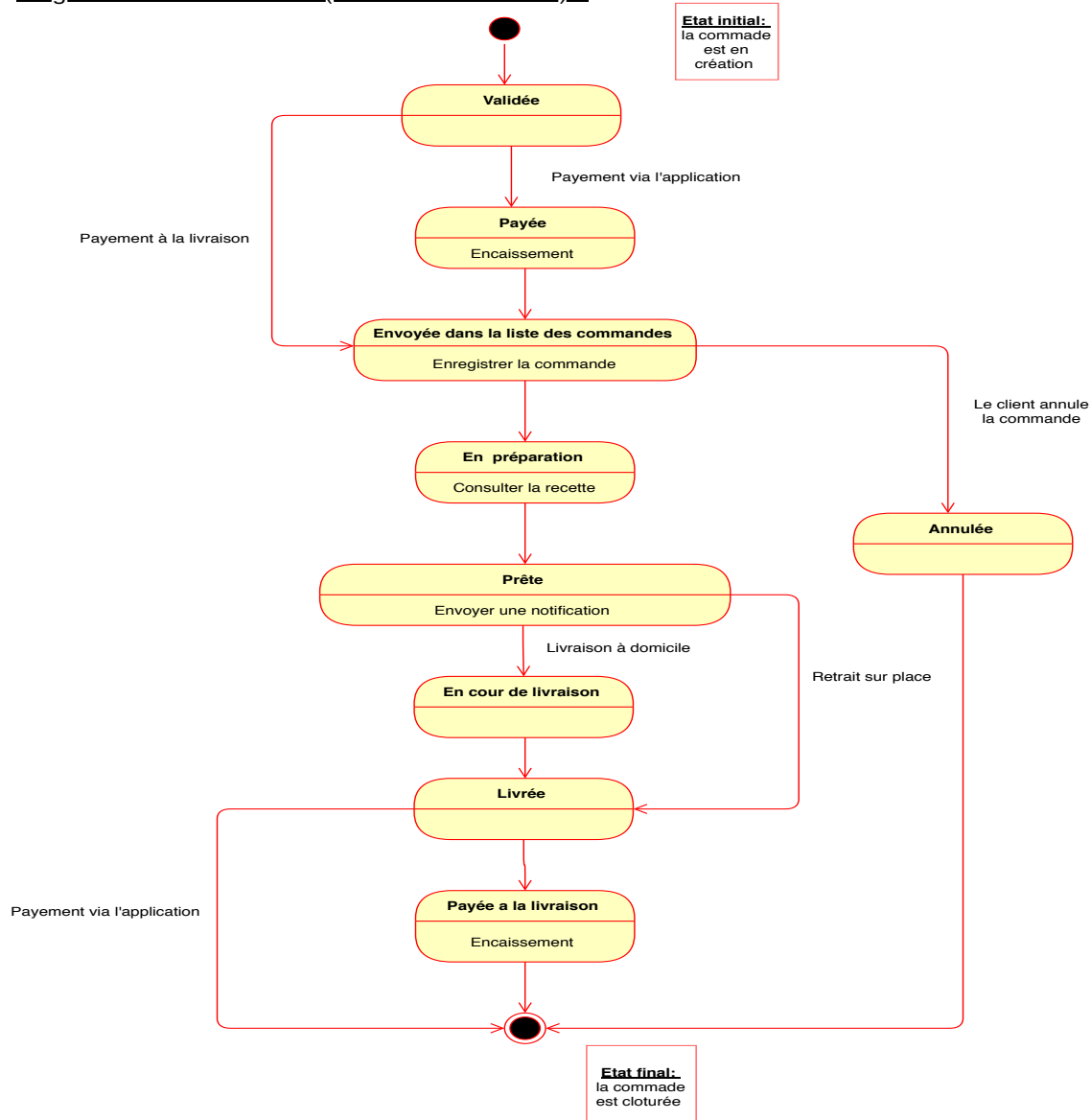
#### 5.1.1 - Diagramme D'activité :





## 5.1.2 - Diagramme d'état- transition :

Diagramme d'état-transition (vie d'une commande) :



## 6 - APPLICATION WEB & MOBILE

L'application est multi-utilisateur, elle s'adapte et affiche les onglets selon l'utilisateur qui est connecté.

- Les utilisateurs ont accès à la carte des pizzas et au contact, leurs permettant de commander par téléphone, directement au point de vente souhaité.
- Les clients ayant un compte et qui sont authentifiés, peuvent commander directement sur l'application et choisir une pizzeria pour récupérer la commande ou peuvent se la faire livrer.
- Les pizzaiolos connectés ont accès à toute l'application ou le site. Des onglets supplémentaires leurs permettent de consulter les recettes des pizzas, les commandes à préparer. Tout ça, à partir de l'application.
- Les gérants connectés ont accès aux mêmes fonctionnalités que le pizzaiolo. Ils peuvent également consulter le stock d'ingrédients restant dans leur pizzeria. Et réajuster les stocks lors d'une livraison. Ils ont également accès à la recette du jour pour effectuer les comptes
- Le gérant connecté peut ajouter ou supprimer des employés dans la base de données par l'intermédiaire d'un onglet réservé et accède à tous les stocks des points de vente. Il accède également à la recette de tous les points de vente pour la gestion de la comptabilité.

Pour la création du site web on utilisera les langages : **JavaScript, HTML, CSS, Python** avec le framework **Django** qui va nous permettre de gérer un site web avec une architecture bien organisée et qui peut être liée à une base de données **PostgreSQL**.

**L'API** pour communiquer et échanger mutuellement des services et des données avec d'autres applications.

Pour la création de l'application mobile on utilisera le langage **Java** et elle sera développée sous Android.

Il est préférable de créer des applications mobiles pour tout type d'appareils mobiles qui permettront l'accès avec connexion ou sans connexion si nécessaire pour le directeur et le pizzaiolo.

Le système de paiement sera géré avec la fonctionnalité Stripe.

### 6.1 - Les acteurs

Les acteurs ont été définis précédemment.

Afin d'être plus proche de la réalité nous avons créé des Personas qui sont des avatars de nos acteurs

Voici la liste des utilisateurs potentiels de cette application.

### 6.1.1 - Client :

Contexte	Besoins	Implications
<ul style="list-style-type: none"> <li>- Jean</li> <li>30 ans</li> <li>- célibataire</li> <li>- n'aime pas cuisiner</li> <li>- hyper connecté</li> </ul>	<ul style="list-style-type: none"> <li>- commander une pizza depuis son téléphone</li> <li>- payer en ligne</li> <li>- être livré à domicile</li> <li>- être informé en temps réel</li> </ul>	<ul style="list-style-type: none"> <li>- commande via l'application</li> <li>- système d'encaissement bancaire</li> <li>- livraison possible</li> <li>- notifier de l'avancée d'une commande</li> </ul>

### 6.1.2 - Directeur :

Contexte	Besoins	Implications
<ul style="list-style-type: none"> <li>_ Jack</li> <li>_ 46 ans</li> <li>_ Marié</li> <li>_ Directeur</li> </ul>	<ul style="list-style-type: none"> <li>-la gestion centralisée de toutes les pizzerias</li> <li>- être plus efficace dans la gestion des commandes</li> <li>- suivre en temps réel les commandes de la préparation à la livraison</li> <li>- suivre en temps réel le stock d'ingrédients</li> </ul>	<ul style="list-style-type: none"> <li>-pouvoir modifier et consulter le stock</li> <li>-pouvoir modifier et ajouter les comptes pizzaiolos et livreurs</li> <li>-consulter l'état d'une commande en temps réel</li> </ul>

### 6.1.3 - Livreur :

Contexte	Besoins	Implications
- Martin - 19 ans - Étudiant	- être informé d'une nouvelle livraison - avoir les coordonnées du client - savoir si le paiement est effectué ou pas encore - livrer rapidement - notifier que la livraison est effectuée	- accès au compte client pour l'adresse et le téléphone - avoir un GPS sur l'application - notifier de l'heure de son arrivée chez le client

### 6.1.4 - Pizzaiolos :

Contexte	Besoins	Implications
- David - 40 ans - Marié - cuisinier	- recevoir les commandes en ordre - accéder à ses recettes facilement - informer de la fin de la préparation de la commande - avoir accès au stock disponible	- notifier de l'arrivée d'une nouvelle commande - rubrique recette sur son écran - option fin de préparation sur la commande

## 6.2 - Les règles de gestion générales

- Le directeur pourra gérer les pizzerias via le site web et l'application mobile.
- Le client peut consulter et commander sur le site ou l'application.
- Le pizzaiolo peut consulter les commandes et les recettes sur l'application (sur une tablette qui lui sera accessible).
- Le livreur aura un accès limité à l'application avec une géolocalisation sur un téléphone mobile.

## 7 - GLOSSAIRE

<b>Back End</b>	<p>Le back End (arrière-plan) en opposition au front End, fait référence à l'ensemble des éléments qui ne sont pas visibles directement sur le site web mais qui va permettre le bon fonctionnement d'un site internet.</p> <p>Le back end est exécuté côté serveur et utilise principalement Python, Java, PHP.</p>
<b>Front End</b>	<p>Le Front End ou (web frontal) est la partie d'une page internet ou d'une application qu'un utilisateur puisse voir et avec lesquelles il peut interagir directement.</p> <p>Le front end est exécuté côté client (PC, Smartphone, Tablette)</p> <p>Le front end utilise principalement HTML, CSS et JavaScript.</p>
<b>Impact Mapping</b>	<p>L'Impact Mapping va permettre de délimiter le périmètre fonctionnel de notre projet, en passant en revue les fonctionnalités dont les utilisateurs auront besoin.</p>
<b>PostgreSQL</b>	<p>PostgreSQL est un système de gestion de base de données relationnelles (SGBDR)</p> <p>Une base de données permet d'enregistrer des données de façon organisée et hiérarchisée. C'est donc une collection d'informations organisées afin d'être facilement consultables, gérables et mises à jour.</p>
<b>Workflow</b>	<p>Le workflow ou « flux de travaux » est la représentation sous forme de flux des opérations à réaliser pour accomplir l'ensemble des tâches.</p>
<b>Back Office</b>	<p>Le back office regroupe un ensemble de tâches administratives ou logistiques liées à la vente (mise à jour des stocks, livraison, moyens de paiement, etc.)</p> <p>C'est la partie du site visible par le personnel de la pizzeria.</p>
<b>Javascript</b>	<p>Le JavaScript désigne un langage de développement informatique, et plus précisément un langage de script orienté objet. Il est employé pour rendre des pages.</p>
<b>PostgreSQL</b>	<p>PostgreSQL est un système de gestion de base de données relationnelles (SGBDR)</p> <p>Une base de données permet d'enregistrer des données de façon organisée et hiérarchisée. C'est donc une collection d'informations organisées afin d'être facilement consultables, gérables et mises à jour.</p>
<b>Personas</b>	<p>Un Persona est une personne fictive dotée d'attributs et de caractéristiques sociales et qui représente un groupe cible. On personnifie nos cibles avec des besoins et un nom afin de donner une image sur un utilisateur type.</p>



Python	Python est le langage de programmation open source, multiplateformes. C'est un langage de programmation interactif et orienté objet.
--------	---

Cas d'utilisation	Le cas d'utilisation correspond à un ensemble d'actions réalisées par le système en interaction avec les acteurs en vue d'une finalité, il décrit ce qu'un utilisateur doit pouvoir faire grâce au logiciel à développer.
CSS	Le CSS est l'acronyme de « Cascading Style Sheets » ce qui signifie « feuille de style en cascade ». Le CSS correspond à un langage informatique permettant de mettre en forme des pages web.
Framework	Un Framework est un cadre ou un socle dont l'objectif est de simplifier et d'uniformiser le travail des développeurs. Il établit les fondations d'un logiciel ou son squelette applicatif.
Front Office	Le front office est l'ensemble des actions, fonctions ou tâches liées à la vente se faisant en contact avec le client. C'est la partie du site visible par le client.
HTML	Le HTML pour HyperText Markup Language, désigne un type de langage informatique. Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure.