

OC PIZZA

CREATION DE SITE INTERNET

Dossier de spécification technique

Version 002

Auteur
Delphine Miel

TABLE DES MATIERES

1 - Versions	3
2 - Glossaire	4
3 - Introduction	
3.1 - Objet du document	5
3.1.1 -Contexte	5
4 -Domaine fonctionnel	
4.1 Référentiel : Diagramme de classes	6
4.2 Règles de gestion : Associations	7
5 Architecture technique	
5.1 -Application web: Diagramme de composants	9
5.1.1 - Gestion des menus et des stocks	9
5.1.2 - Gestion de la commande	10
5.1.3 - Gestion des acteurs principaux	10
6 -Architecture de déploiement	
6.1 Diagramme de déploiement	11
6.2 Explications	11

1 - VERSIONS

Auteur	Date	Description	Version
Delphine Miel	20/11/2020	Dossier de conception fonctionnelle	001
Delphine Miel	26/02/2021	Dossier de spécification technique	002

2- GLOSSAIRE

Diagramme de classes	Le diagramme de classes est un schéma utilisé pour présenter les <u>classes</u> et les <u>interfaces</u> des systèmes ainsi que leurs relations.
Diagramme de composants	Un diagramme de composants UML vous permet d'obtenir une vue d'ensemble de votre système logiciel
Diagramme de déploiement	Le diagramme de déploiement décrit le déploiement physique des informations générées par le logiciel sur des composants matériels.
Artefact	On appelle artefact l'information qui est générée par le logiciel.
Composant	Un élément logiciel
Interface	Une relation contractuelle. Les objets qui réalisent l'interface doivent remplir une sorte d'obligation.

3- INTRODUCTION

3.1 Objet du document

Le présent document constitue le dossier de spécification technique de **l'application OC PIZZA**. Il a pour objectif de décrire le domaine fonctionnel ainsi que l'architecture technique et de déploiement de l'application.

3.1.1 Contexte

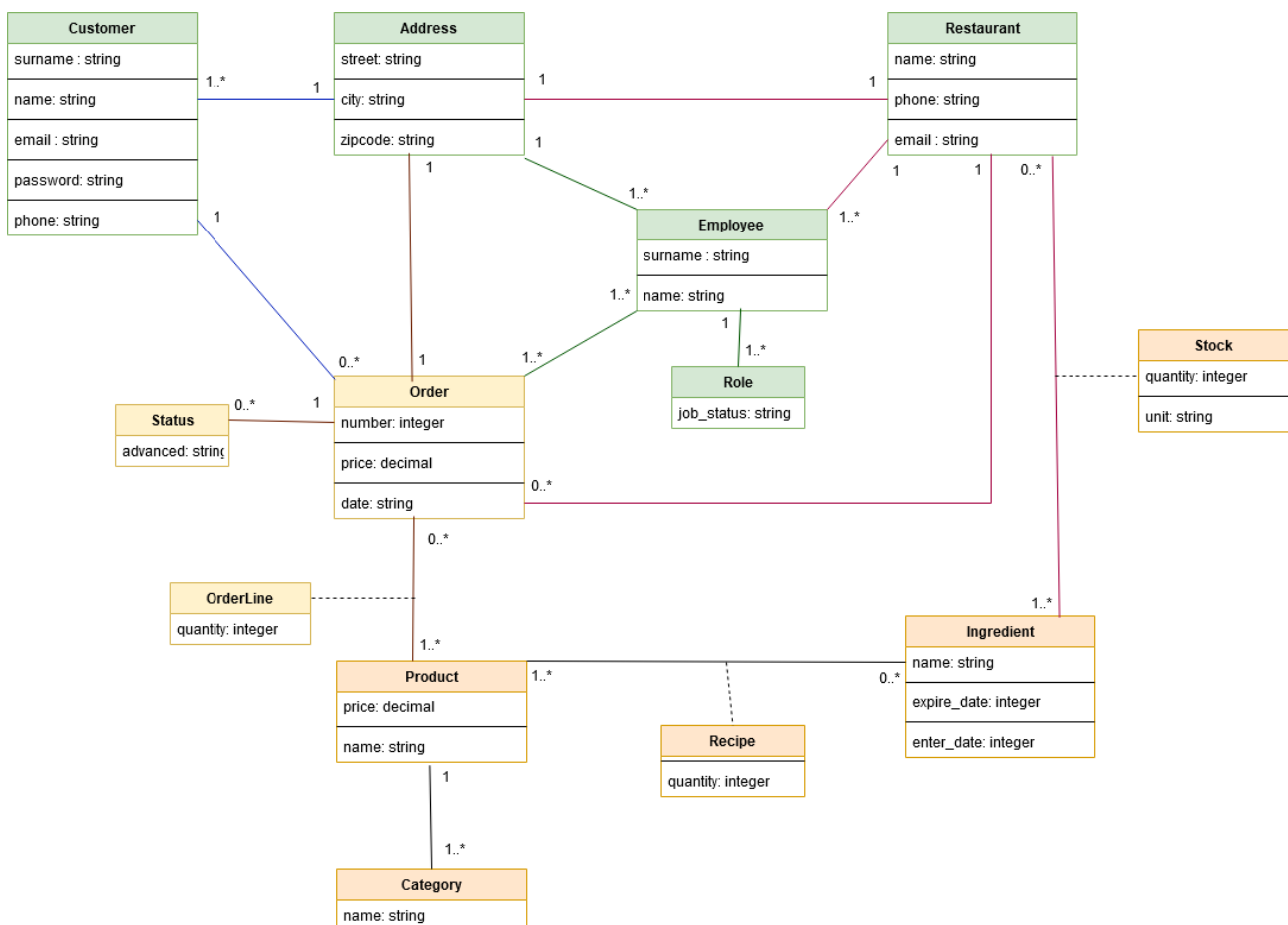
« **OC Pizza** » est un jeune groupe de pizzerias en plein essor et spécialisé dans les pizzas livrées ou à emporter. Il compte **déjà 5 points de vente** et prévoit d'en ouvrir au moins 3 de plus d'ici la fin de l'année.

Un des responsables du groupe a pris contact avec nous afin de **mettre en place un système informatique, déployé dans toutes ses pizzerias** et qui lui permettrait notamment :

- **Plus d'efficacité dans la gestion des commandes**, de leur réception à leur livraison en passant par leur préparation ;
- Un **suivi en temps réel des commandes** passées et en préparation ;
- Un **suivi en temps réel du 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'ils le souhaitent – sinon, ils paieront directement à la livraison
 - Modifier ou annuler leur commande 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**.

4 DOMAINE FONCTIONNEL

4.1 Référentiel : Diagramme de classes



TRADUCTION :

- ❖ Employee = Employé
- ❖ Customer = Client
- ❖ Address = Adresse
- ❖ Order = Commande
- ❖ Product = Produit

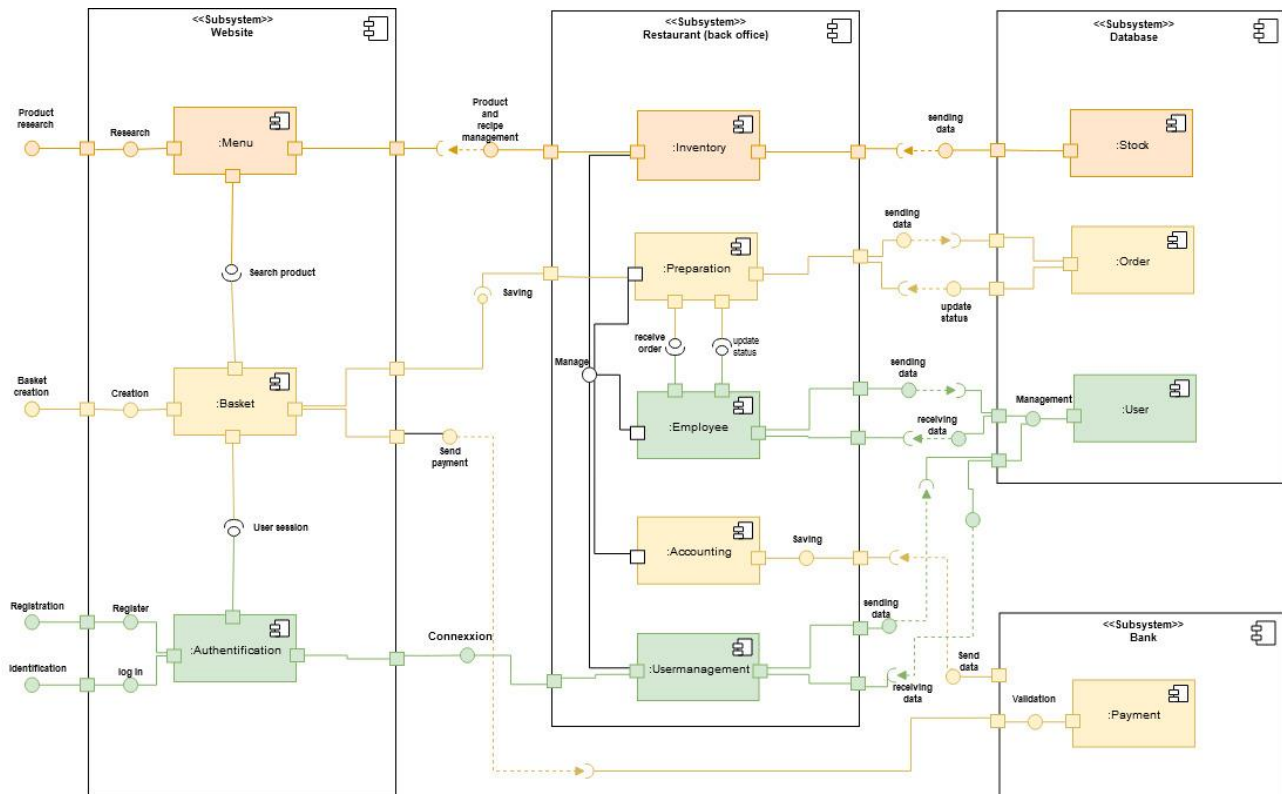
4.2 Règles de gestion : Multiplicités

- **Customer – Address :**
 - Un client ne peut avoir qu'une adresse.
 - Une adresse peut appartenir à un ou plusieurs clients.
- **Customer – Order :**
 - Un client peut avoir zéro ou plusieurs commandes
 - Une commande ne peut appartenir qu'à un seul client
- **Restaurant – address :**
 - Un restaurant peut avoir qu'une adresse
 - Une adresse ne peut appartenir qu'à un seul restaurant
- **Restaurant – Employee :**
 - Un restaurant peut avoir un ou plusieurs employés
 - Un employé ne peut appartenir qu'à un seul restaurant
- **Restaurant – Ingredient :**
 - Un restaurant peut stocker zéro ou plusieurs ingrédients
 - Un ingrédient peut être stocké par un ou plusieurs restaurant
- **Restaurant – Order :**
 - Un restaurant peut avoir zéro ou plusieurs commandes
 - Une commande ne peut appartenir qu'à un seul restaurant
- **Employee – Address :**
 - Un employé ne peut avoir qu'une adresse.
 - Une adresse peut appartenir à un ou plusieurs employés.
- **Employee – Order :**
 - Un employé peut gérer une à plusieurs commandes
 - Une commande peut être gérée par un ou plusieurs employés
- **Employee – Role :**
 - Un employé peut avoir qu'un seul rôle
 - Un rôle peut être occupé par un ou plusieurs employés

- **Oder – Address :**
 - Une commande ne peut avoir qu'une seule adresse
 - Une adresse appartenir qu'à une commande
- **Order – Status :**
 - Une commande ne peut avoir qu'un seul statut
 - Un statut peut appartenir à zéro ou plusieurs commandes
- **Order – Product :**
 - Une commande peut avoir un ou plusieurs produits
 - Un produit peut être attribué à zéro ou plusieurs commandes
- **Product – Category :**
 - Un produit ne peut avoir qu'une seule catégorie
 - Une catégorie peut être attribué à un ou plusieurs produits
- **Product – Ingredient :**
 - Un produit peut être composé d'un ou plusieurs ingrédients
 - Un ingrédient peut se retrouver dans zéro ou plusieurs produits

5 ARCHITECTURE TECHNIQUE

5.1 Application Web : Diagramme de composants



5.1.1 Gestion des Menus et des stocks

- Dans le **site internet**, le composant « **Menu** » propose une interface à l'utilisateur afin de faire une recherche de produit.
- Ce dernier récupère les informations via une interface de « *product and recipe management* » afin de disposer des informations produits fourni par le composant « **inventory** » se trouvant dans le **restaurant** (Back-Office).
- Toutes ces informations sont elles-mêmes récupérées dans le composant « **stock** » de la **base de données**.

5.1.2 Gestion de la commande

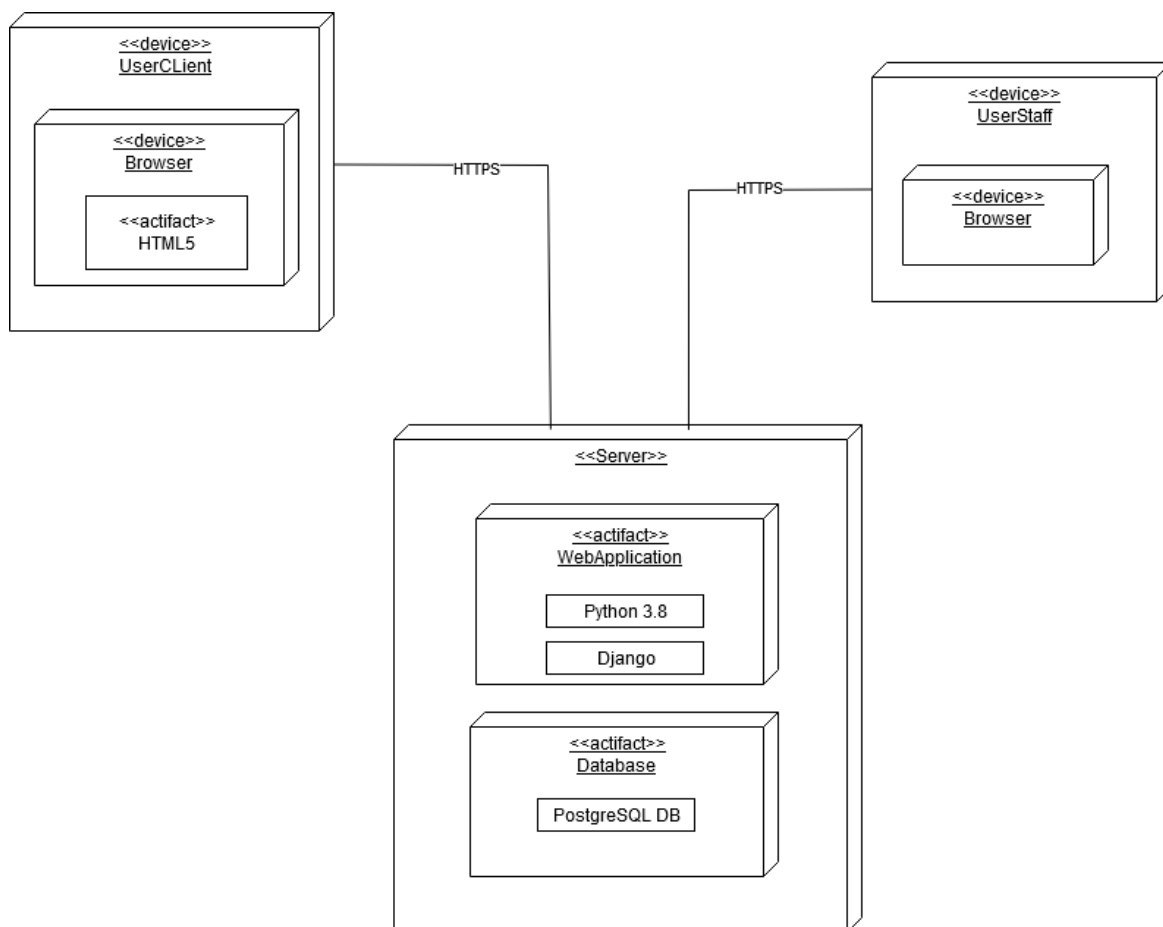
- Dans le **site internet**, le composant « **Basket** » va accomplir trois actions :
 - Il va proposer une interface à l'utilisateur afin de créer un panier.
 - Il va envoyer le paiement du panier au composant « **payment** » de la **Banque**.
 - La **banque** quant à elle, fournit l'information au composant « **accounting** » (finance) du **restaurant**.
 - Il va fournir les informations contenues dans le panier au composant « **preparation** » du **restaurant**.
 - Le **restaurant** va mettre a disposition des employés, les informations contenues dans le composant « **preparation** »
 - Le **restaurant** va communiquer avec le composant « **order** » de la **base de données** afin de soit sauvegarder la commande, soit de récupérer les informations comme le contenu ou l'avancement de celle-ci.

5.1.3 Gestion des acteurs principaux

- Dans le **site internet**, le composant « **authentification** » va proposer deux interfaces à l'utilisateur :
 - Une interface lui permettant de *s'inscrire* au site.
 - Une interface lui permettant de *s'identifier* (quand l'inscription a été faite au préalable).
- Le composant « **user management** » du **restaurant** va se connecter au **site internet** via une interface pour récupérer les informations clients et les gérer.
- Le composant « **user** » de la **base de données** va interagir avec deux composants du **restaurant** :
 - Le composant « **user management** » pour y envoyer des données ou en recevoir sur le sujet des clients (nom, adresse, email, mot de passe, téléphone).
 - Le composant « **employee** » (employé) pour y envoyer des données ou en recevoir sur le sujet des employés (nom, adresse, statut...).
- Le composant « **employee** » du restaurant va aussi faire une mise à jour des commandes en envoyant les informations au composant « **preparation** » du restaurant lui-même.

6 ARCHITECTURE DE DEPLOIEMENT

6.1 Diagramme de déploiement



6.2 Explications

- Le navigateur (browser) de « la machine utilisateur » et celui de « la machine du restaurant » communiquent avec l'application se trouvant sur un serveur web via un protocole HTTPS.
- Le serveur héberge :
 - L'application web construite avec python et Django
 - La base de données manipulée sur PostgreSQL