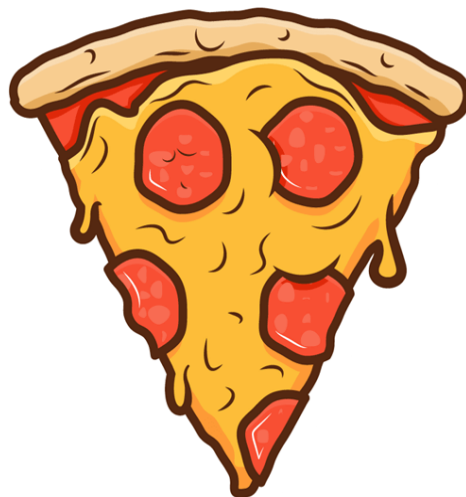


OC Pizza

Dossier de conception technique

Projet 9 - Documentez votre système de gestion de pizzeria



Auteur

RUFFIER Quentin

TABLE DES MATIÈRES

Versions	4
Introduction	4
Objet du document	5
Références	5
Le domaine fonctionnel	5
Diagramme de classe	6
Les classes	7
Les classes Account, Customer, Staff	7
La classe Command	7
La classe Pizza	8
La classe Ingredient	8
La classe Recipe	8
La classe Stock	8
La classe Restaurant	9
Architecture technique	9
Modèle physique de données	10
Diagramme de composant	11
Diagramme de déploiement	12
Le déploiement	13
Architecture logicielle	13
Principes généraux	13
Django	13
Structure des sources	14
Points particuliers	15
Fichiers de configuration	15
Ressources	15
Environnement de développement	15
Procédure de packaging / livraison	15
Glossaire	16

1 - VERSIONS

Auteur	Date	Description	Version
Quentin Ruffier	30/07/2021	Création du document	1.0

2 - INTRODUCTION

2.1 -Objet du document

Ce document représente le dossier de conception technique de l'application OC Pizza.
Il fait suite au document de conception fonctionnelle.

Il a pour but d'analyser la demande du client OC Pizza afin de :

- Modéliser les objets du domaine fonctionnel ;
- Identifier les différents éléments composant le système à mettre en place et leurs interactions ;
- Décrire le déploiement des différents composants envisagés ;
- Élaborer le schéma de la base de données.

Les éléments présents dans le dossier permettront de comprendre la nature, l'organisation et les relations entre les composants de la solution informatique précédemment décrite dans le dossier de conception fonctionnelle.

2.2 -Références

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

1. **OC Pizza - Dossier de conception fonctionnelle**
2. **OC Pizza - Dossier d'exploitation**
3. **OC Pizza - PV de livraison**

3 - LE DOMAINE FONCTIONNEL

3.1 -Diagramme de classe

Le diagramme ci-dessous montre les différentes classes de notre système et leurs associations. Les clés primaire et étrangère ne sont pas représentées sur celui-ci.

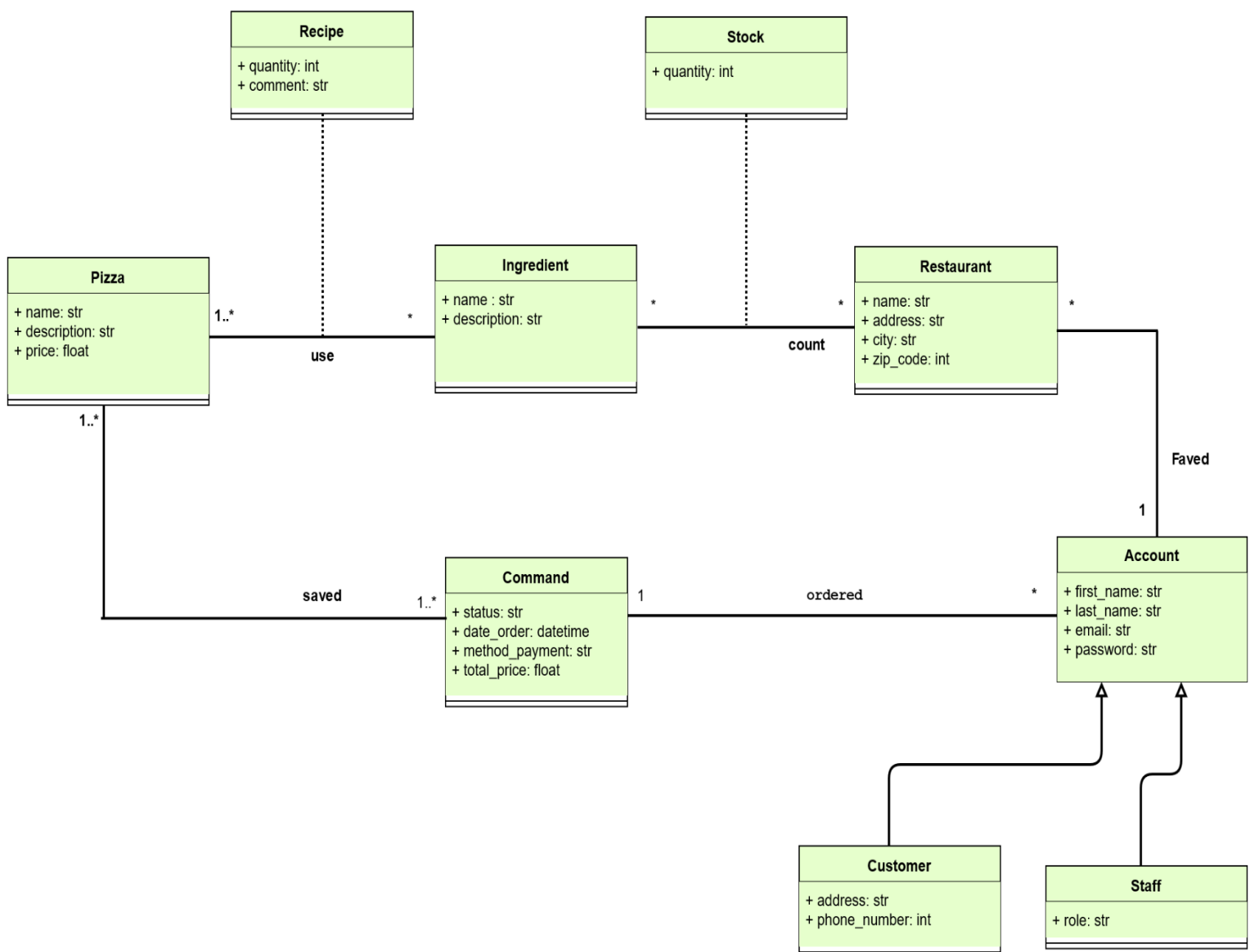


Figure 1 : Diagramme de classe

3.2 - Les classes

3.2.1 - Les classes Account, Customer, Staff

Cette classe mère représente le système d'authentification des différents utilisateurs du système.

Deux classes filles héritent de ses attributs.

Attributs	Description
first_name	Prénom
last_name	Nom
email	Courrier électronique
password	Mot de passe
role	Rôle au seins de OC Pizza
address	Adresse du client
phone_number	Numéro de téléphone

3.2.2 - La classe Command

Le système de prise de commande permet l'affichage en temps réel du statut. Permet de retrouver un historique avec la date.

Attributs	Description
status	Statut de la commande
date_order	Date et heure de la commande
method_payment	Moyen de paiement utilisé
total	Le prix total de la commande

3.2.3 - La classe Pizza

Cette classe représente les Pizzas à commander

Attributs	Description
name	Nom de la pizza
description	Description de sa composition
price	Prix

3.2.4 - La classe Ingredient

Cette classe représente les ingrédients utiles à la préparation des Pizzas.

Attributs	Description
name	Nom
description	Description

3.2.5 - La classe Recipe

Cette classe permet de retrouver les recettes des différentes pizzas pour la fonctionnalité "Mémo".

Attributs	Description
quantity	Quantité d'ingrédients
comment	Commentaire pour la préparation

3.2.6 - La classe Stock

Permet de suivre le stock d'ingrédients encore disponible par point de vente.

Attributs	Description
-----------	-------------

quantity	Quantité d'ingrédients
----------	------------------------

3.2.7 - La classe Restaurant

Cette classe permet de retrouver les points de vente de OC Pizza.

Attributs	Description
name	Nom du point de vente
address	Adresse
city	Ville
zip_code	Code postal



4 - ARCHITECTURE TECHNIQUE

4.1 -Modèle physique de données

Inspiré du diagramme de classes, le modèle physique de données (MPD) présente les tables utilisées pour stocker les données du système informatique. Il s'agit de la représentation de la base de données (BDD).

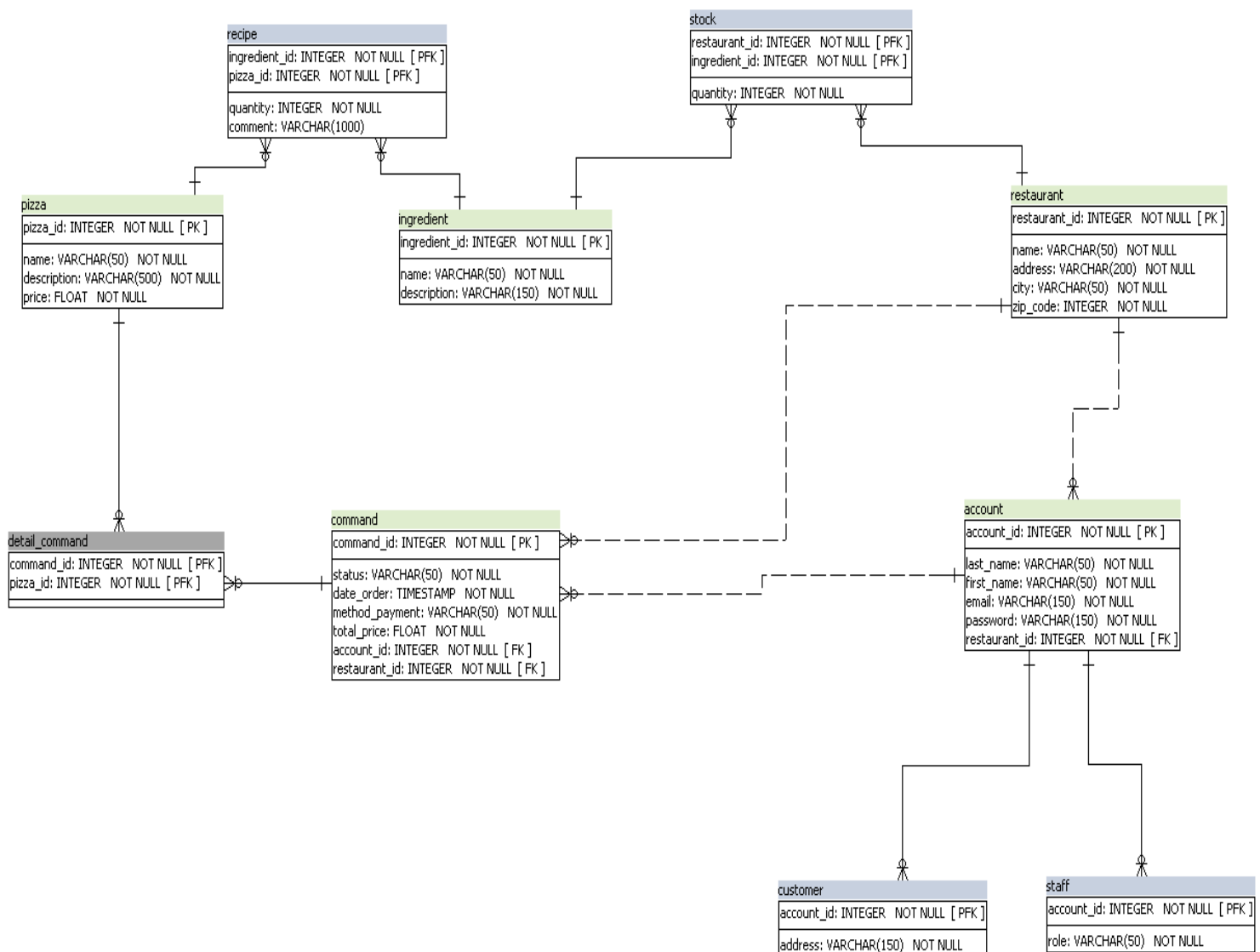


Figure 2 : MPD

4.2 - Diagramme de composant

Le diagramme de composants ci-dessous décrit l'organisation du système du point de vue des éléments logiciels. Il permet de mettre en évidence les dépendances entre les différents composants.

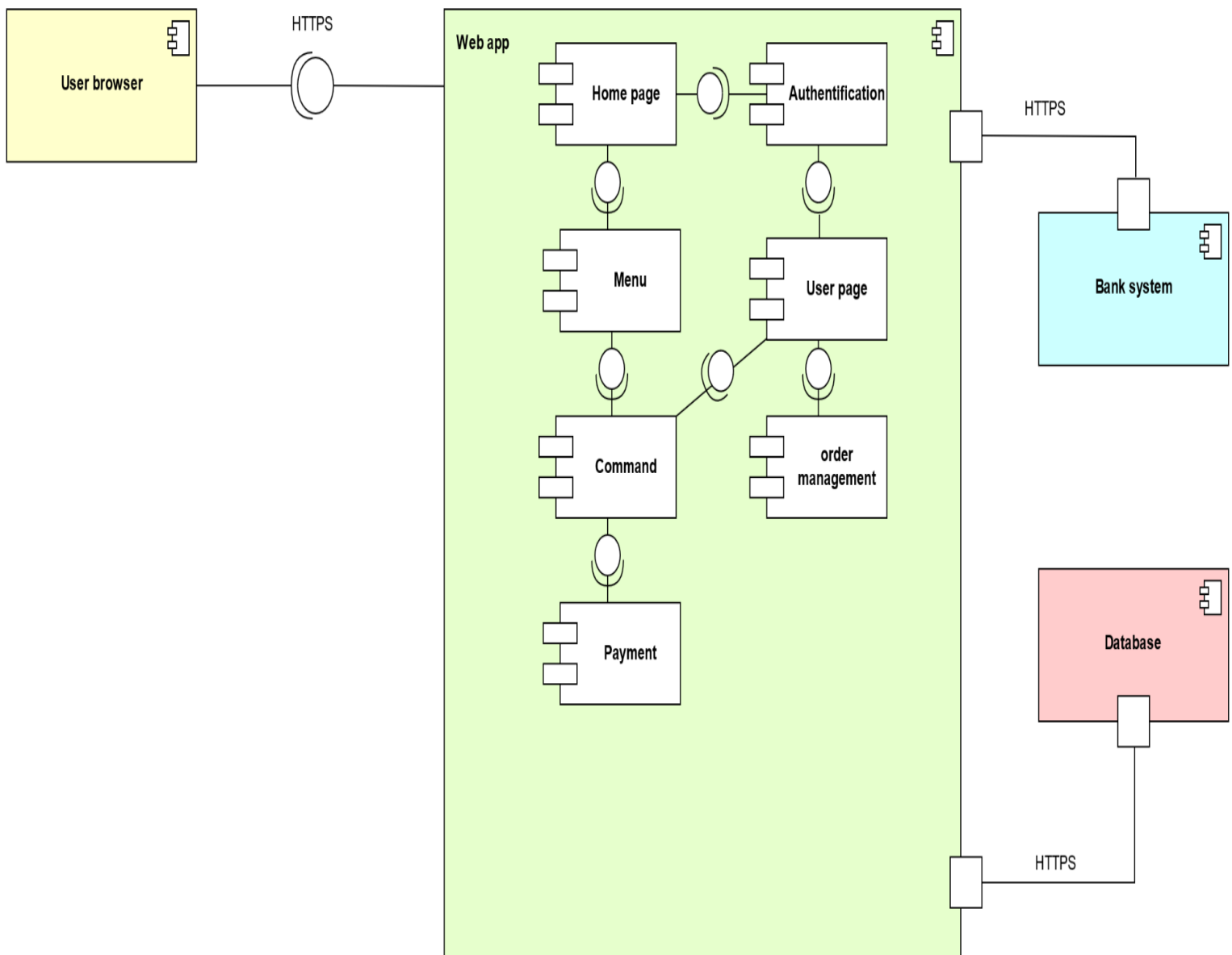


Figure 3 : Diagramme de composant

4.3 - Diagramme de déploiement

Le diagramme de déploiement ci-dessous représente l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que leurs relations entre eux.

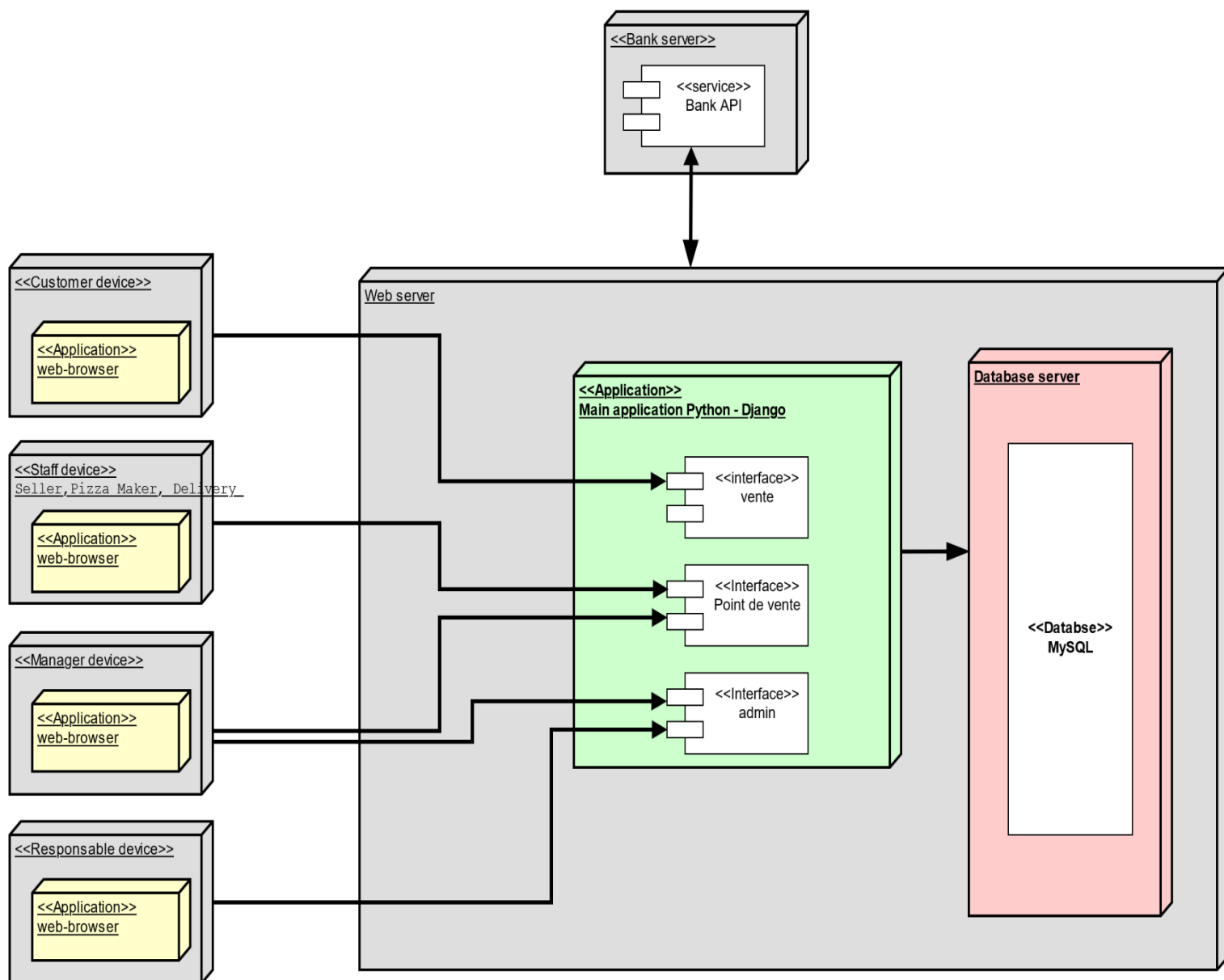


Figure 4 : Diagramme de déploiement

4.4 - Le déploiement

Le système OC Pizza sera déployé sur des serveurs OVH. Pourquoi OVH ? Ce service d'hébergement web est une entreprise française basée dans le nord de la France, présente sur le marché depuis 1999, sa fiabilité est excellente.

L'offre de services d'OVH ne s'arrête pas à l'hébergement, elle s'élargit avec toute une suite de services "Cloud" (stockage et sauvegarde).

De plus, OVH dispose d'un service de statistiques et logs installé et consultable facilement grâce à l'espace client. Des modules peuvent être installés sur le serveur afin d'avoir, par exemple, un monitoring pour surveiller partiellement votre serveur et son activité liée à des parties telles que le processeur, la mémoire vive (RAM), les partitions disques, etc. (Real Time Monitoring) .

Cette solution semble tout à fait appropriée pour le système de OC Pizza.

5 - ARCHITECTURE LOGICIELLE

5.1 -Principes généraux

Les sources et versions du projet sont gérées par Git, les dépendances et le packaging par pip. Il s'agit d'un projet développé avec Python et Django. Django propose un ORM pour communiquer avec la base de données permettant de s'abstraire de l'écriture SQL.

Django crée le projet sous la forme du design pattern « MVT » (Model, View, Template).

Les modèles servent à construire la base de données, les vues à définir les requêtes http et leur parcours sur le site et les « templates » structurent les pages en HTML.

5.1.1 - *Python - Django*

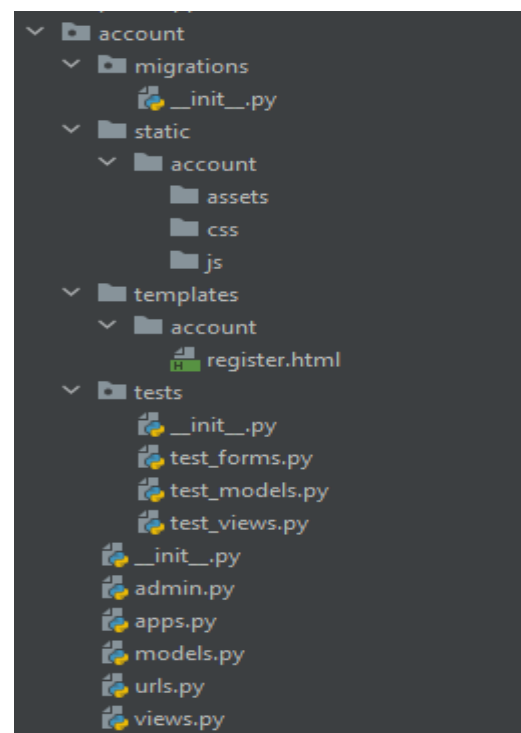
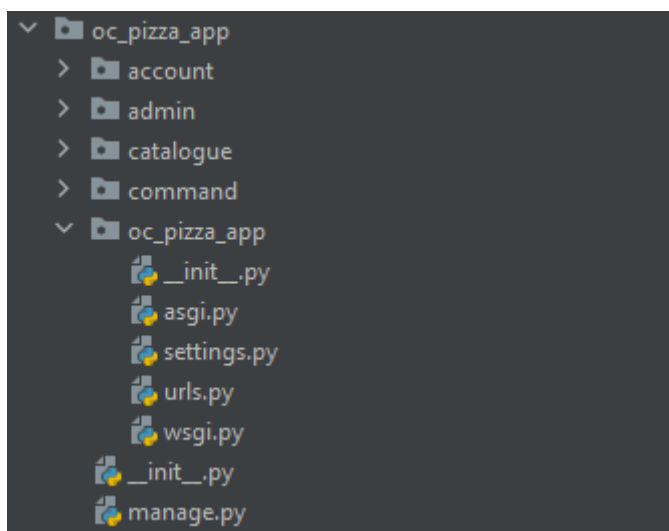
Dans Django, les modules sont appelés "Application", dans notre système nous aurons cinq applications au sein du Projet "**oc_pizza_app**" :

- **account** : Pour le système d'authentification utilisé par tous les acteurs.
- **admin** : La gestion des points de vente et des stocks.

- **command**: Pour le système de gestion de commande utilisé par les clients et les employés, et le système de mémo des recettes.
- **catalogue**: Pour la gestion des différentes interfaces, l'affichage des cartes, menus, produits, etc.
- **oc_pizza_app**: La partie principale du projet, rassemble tous les réglages

5.1.2 - Structure des sources

La structuration des répertoires du projet suit la logique suivante :



Les applications seront organisées sur le même principe que l'application "**account**".

5.2 -La base de données

Pour la base de données, nous utilisons MySQL, le SGBDR le plus populaire au monde.



5 - POINTS PARTICULIERS

5.1 -Ressources

5.1.1 - *Charte graphique*

Le groupe de pizzerias OC PIZZA doit nous faire parvenir un cahier des charges graphique (couleurs, polices, logos, symboles...) définissant l'identité visuelle (caractéristiques des différents éléments graphiques) de l'entreprise.

5.2 -Environnement de développement

L'utilisation d'un IDE est vivement recommandé. Mais libre aux développeurs d'utiliser la solution qui leur semble la plus adaptée.

5.3 -Procédure de packaging / livraison

L'application fera l'objet d'un déploiement sur le serveur OVH au moment de la livraison finale.

Le dossier **OC pizza - Dossier d'exploitation** , permet de lister toutes les informations dont le client a besoin pour assurer une bonne utilisation de l'application.



6 - GLOSSAIRE

BDD	Base de données
IDE	integrated development environment
MPD	Modèle physique de données