

# OC PIZZA

## Système informatique OC PIZZA

Dossier de conception technique

Version 1.0

**Auteur**

A.VARVOUX

*Analyste-programmeur*

# TABLE DES MATIÈRES

<b>1 - Versions.....</b>	<b>3</b>
<b>2 - Introduction.....</b>	<b>4</b>
2.1 - Objet du document.....	4
2.2 - Références.....	4
<b>3 - Architecture Technique.....</b>	<b>5</b>
3.1 - Composants généraux.....	5
3.1.1 - <i>Package Vente</i> .....	5
3.1.1.1 - Composant Moteur de recherche.....	5
3.1.1.2 - Composant Panier.....	5
3.1.1.3 - Composant Authentification.....	5
3.1.2 - <i>Package Stock Pizzeria</i> .....	5
3.1.2.1 - Composant Stock.....	5
3.1.3 - <i>Package Facturation</i> .....	5
3.1.3.1 - Composant Commande.....	5
3.1.3.1 - Composant Client.....	5
3.1.4 - <i>Package Base de données</i> .....	5
3.1.5 - <i>Package Système bancaire</i> .....	5
3.2 - Application Web.....	6
3.3 - Application Base de données.....	7
<b>4 - Architecture de Déploiement.....</b>	<b>8</b>
4.1 - Serveur de Base de données.....	8
4.2 - Serveur d'application.....	8
<b>5 - Architecture logicielle.....</b>	<b>9</b>
5.1 - Principes généraux.....	9
5.1.1 - <i>Les couches</i> .....	9
5.2 - Application Web.....	9
5.3 - Application mobile.....	9
<b>6 - Points particuliers.....</b>	<b>10</b>
6.1 - Gestion des logs.....	10
6.2 - Fichiers de configuration.....	10
6.2.1 - <i>Application web</i> .....	10
6.3 - Ressources.....	10
6.4 - Environnement de développement.....	10
6.5 - Procédure de packaging / livraison.....	10

# 1 - VERSIONS

Auteur	Date	Description	Version
A.VARVOUX	19/03/2021	Création du document	1.0

## 2 - INTRODUCTION

### 2.1 - Objet du document

Le présent document constitue le dossier de conception technique du système informatique de gestion et de vente en ligne de l'entreprise OC Pizza.

Objectif du document : définir la démarche de conception de l'application OC PIZZA.

### 2.2 - Références

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

1. **Projet OCPIZZA – Dossier de conception fonctionnelle** : Dossier de conception fonctionnelle de l'application
2. **Projet OCPIZZA – Dossier d'exploitation** : Dossier d'exploitation de l'application
3. **Projet OCPIZZA – PV** : Procès-verbal de livraison

# 3 - ARCHITECTURE TECHNIQUE

## 3.1 - Composants généraux

### 3.1.1 - Package Vente

Le package Vente représente tout ce qui se rapporte à la vente de produits des pizzerias OC Pizza.

#### 3.1.1.1 - Composant Moteur de recherche

Ce composant gère toutes les recherches dans le catalogue de produits.

#### 3.1.1.2 - Composant Panier

Ce composant gère le panier, à partir de la création de celui-ci, jusqu'à la validation.

#### 3.1.1.3 - Composant Authentification

Ce composant gère l'authentification. Il y a une vérification des identifiants de connexion et des mots de passe des utilisateurs en interrogeant la base de données.

### 3.1.2 - Package Stock Pizzeria

Le package Stock Pizzeria représente le stock théorique et réel de la pizzeria.

#### 3.1.2.1 - Composant Stock

Ce composant gère tous les stocks disponibles.

### 3.1.3 - Package Facturation

Ce package représente la finalisation de la commande, c'est-à-dire la facturation.

#### 3.1.3.1 - Composant Commande

Ce composant gère toute la partie de prise de commande, de sa validation jusqu'à sa livraison.

#### 3.1.3.2 - Composant Client

Ce composant représente les clients d'OC Pizza, qu'ils passent commande avec leur compte dédié, un compte invité, ou bien s'ils ont passé commande via téléphone, grâce au personnel d'accueil du restaurant.

### 3.1.4 - Package Base de données

Ce composant représente la base de données PostgreSQL de l'application, «db\_ocpizza2».

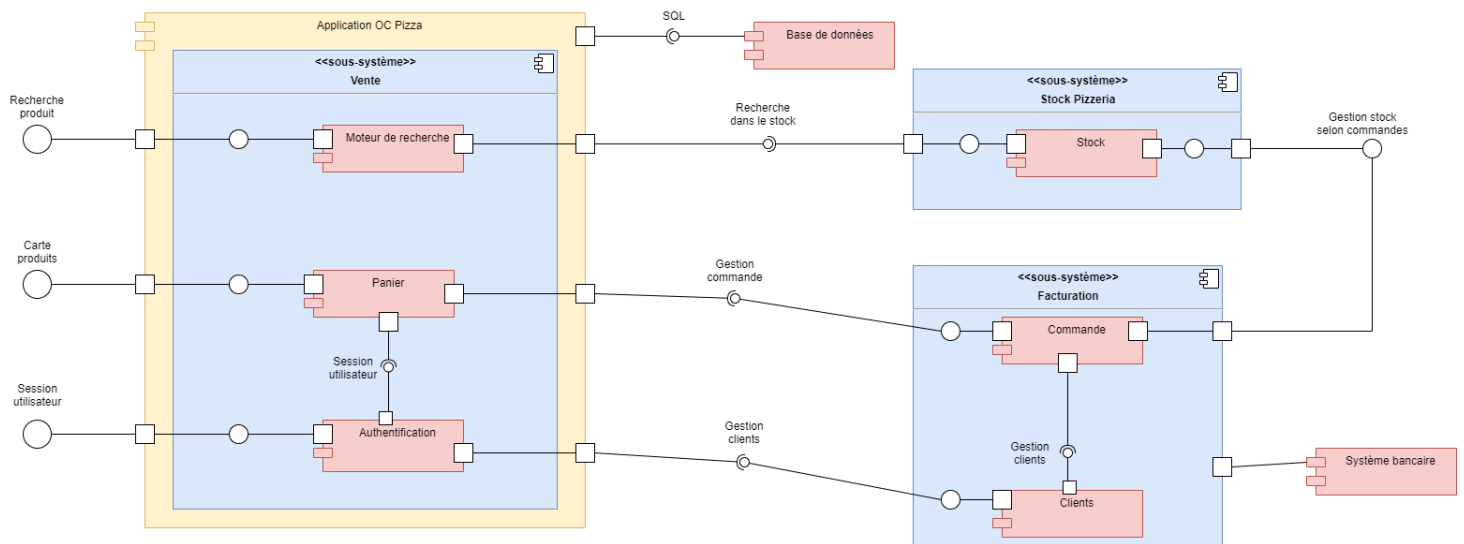
### 3.1.5 - Package Système bancaire

Ce composant représente le système de paiement des restaurants OC Pizza.

## 3.2 - Application Web

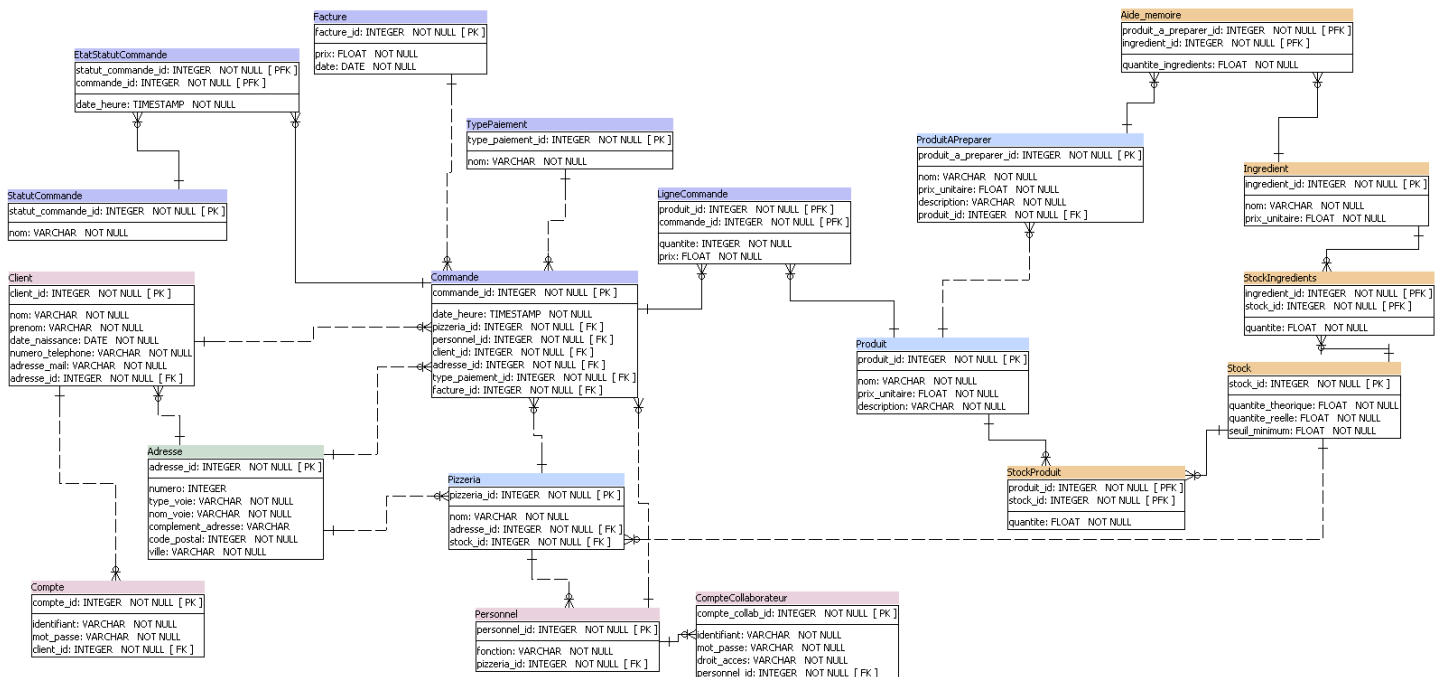
La pile logicielle est la suivante :

- Application **Java EE 8** / HTML5 / CSS3 / Javascript
- Serveur d'application **Apache Tomcat 10.0.2**
- Base de données **PostgreSQL 13.2**



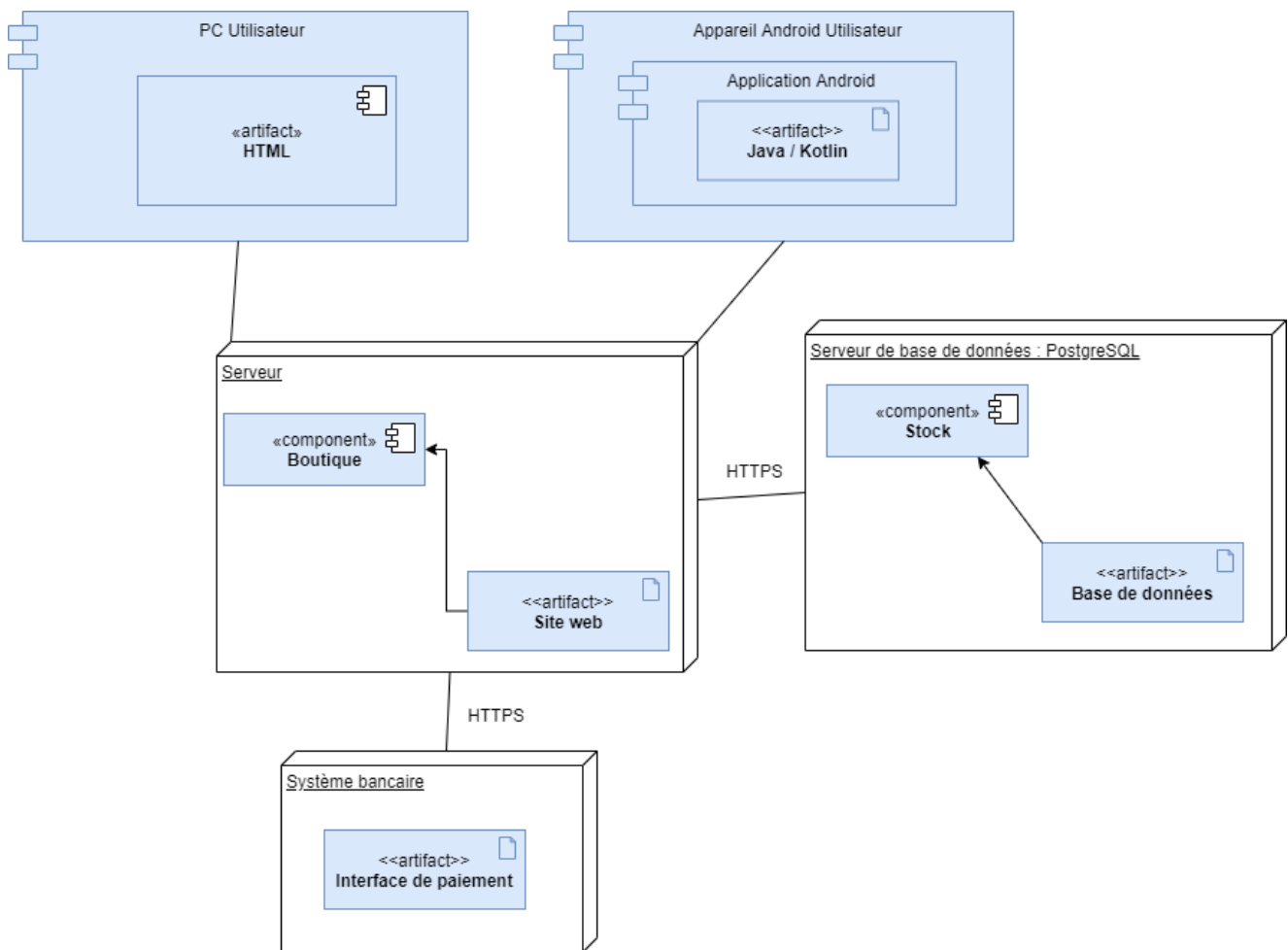
## 3.3 - Application Base de données

### 3.3.1 - Modèle physique de données (MPD)



Le modèle physique de données, le MPD, est élaboré à partir du diagramme de classes. Il permet la préparation de l'implémentation dans un système de gestion de base de données relationnelles, comme PostgreSQL. Passer à l'étape du MPD permet de construire la structure finale de la base de données avec les liens entre les éléments qui composent la base de données.

# 4 - ARCHITECTURE DE DÉPLOIEMENT



## 4.1 - Serveur de Base de données

Un système de gestion de base de données (SGBD) est un logiciel qui permet le stockage d'informations dans une base de données. Il permet de créer, lire, mettre à jour, et supprimer les données qui sont contenues dans la base de données.

Ces données seront stockées dans le SGBD relationnelle **PostgreSQL**, qui est une **base de données open source, populaire et robuste**.

Afin d'administrer cette base de données PostgreSQL, on utilisera *Amazon RDS for PostgreSQL*.

## 4.2 - Serveur d'application

Le serveur d'application choisi est Apache Tomcat, qui est un serveur HTTP.



# 5 - ARCHITECTURE LOGICIELLE

## 5.1 - Principes généraux

Les sources et versions du projet sont gérées par **Git**, les dépendances et le packaging par **Apache Maven**.

### 5.1.1 - Les couches

L'architecture applicative est la suivante :

- une couche **model**: structure des données, n'est pas affecté si il y a une modification de la logique ou de l'interface
- une couche **view**: présentation de l'interface graphique
- une couche **controller**: logique concernant les actions effectuées par l'utilisateur

## 5.2 - Application Web

L'application web sera développée selon l'architecture MVC (Model – View – Controller).

## 5.3 - Application mobile

L'application mobile sera développée selon l'architecture MVVM (Model – View - ViewModel) grâce à l'utilisation des Android Architecture Components, qui est un ensemble de bibliothèques et de bonnes pratiques pour améliorer la qualité des applications.

## 6 - POINTS PARTICULIERS

### 6.1 - Gestion des logs

Les logs seront gérés en utilisant la librairie Log4j2.

### 6.2 - Fichiers de configuration

Le développement sera réalisé avec le framework Spring.

#### 6.2.1 - Application web

Un fichier de configuration « application.properties » sera placé à la racine du projet dans lequel nous aurons les propriétés du projet, comme la configuration pour l'accès à la base de données.

### 6.3 - Ressources

Un fichier contiendra tous les menus nécessaires au système, la précédente base de données, ainsi que toutes les informations relatives aux différents restaurants.

### 6.4 - Environnement de développement

Pour le développement web, l'environnement de développement sera sur Eclipse / Visual Studio Code.

Pour le développement Android, l'environnement de développement sera sur Android Studio.

Si, par la suite, il y a une demande de développement iOS, l'environnement de développement sera sur Xcode.

### 6.5 - Procédure de packaging / livraison

L'application web sera packagée dans un fichier .jar. L'ensemble des scripts de création de la base de données seront livrés dans un fichier .zip.

L'application mobile sera packagée dans un fichier .apk.