



# Groupe OC Pizza

## OC Pizza app

Dossier de conception technique

Version 1.0

### **Auteur**

Edouard Plantevin  
*Analyste Programmeur*

## TABLE DES MATIÈRES

1.Versions	3
2.Introduction	4
2.1.Objet du document	4
2.2.Références	4
4.Architecture Technique	5
4.1.Application Web	5
4.2.Base de données	5
4.2.1.SGBD Utilisé	5
4.2.2.Modèle physique de données	5
4.Architecture de Déploiement	6
4.1.Déploiement de l'application	6
4.2.Serveur de déploiement	6
5.Architecture logicielle	7
5.1.Principes généraux	7
5.1.1.Les couches	7
5.1.2.Les modules	8
5.1.3.Structure des sources	9
6.Points particuliers	10
6.1.Ressources	10
6.1.2.Ressources graphique	10
6.1.2.Données	10
6.2.Environnement de développement	10
6.3.Procédure de packaging / livraison	10
7.Glossaire	11

# 1. VERSIONS

Auteur	Date	Description	Version
EP	07/01/2020	Création du document	1.0

## 2.INTRODUCTION

### 2.1.Objet du document

Le présent document constitue le dossier de conception technique de l'application OC Pizza app  
l'Objectif du document est de présenter les outils, les technologies, et les méthodes mises en oeuvre pour réaliser l'application

Les éléments du présents dossiers découlent :

- De l'entretien réaliser avec le dirigeant de la société OC Pizza
- De l'analyse des besoins et de la rédaction du dossier de conception fonctionnelle.

### 2.2.Références

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

1. **DCF - 1.0** : Dossier de conception fonctionnel
2. **DCF - 1.0** : Dossier d'exploitation

## 4.ARCHITECTURE TECHNIQUE

### 4.1.Application Web

Le développement de l'application sera fait en langage dart (version 2.7.0) avec l'utilisation du framework flutter (version 1.9.1)

### 4.2.Base de données

#### ***4.2.1.SGBD Utilisé***

Le système de gestion de base de données utilisé sera PostgreSQL. Ce choix se justifie en raison de la compatibilité de ce SGBD avec la plate-forme de déploiement

#### ***4.2.2. Modèle physique de données***

Le modèle physique de donnée présenté ci-dessous se base sur le diagramme de classe détaillée dans le Dossier de conception fonctionnelle.

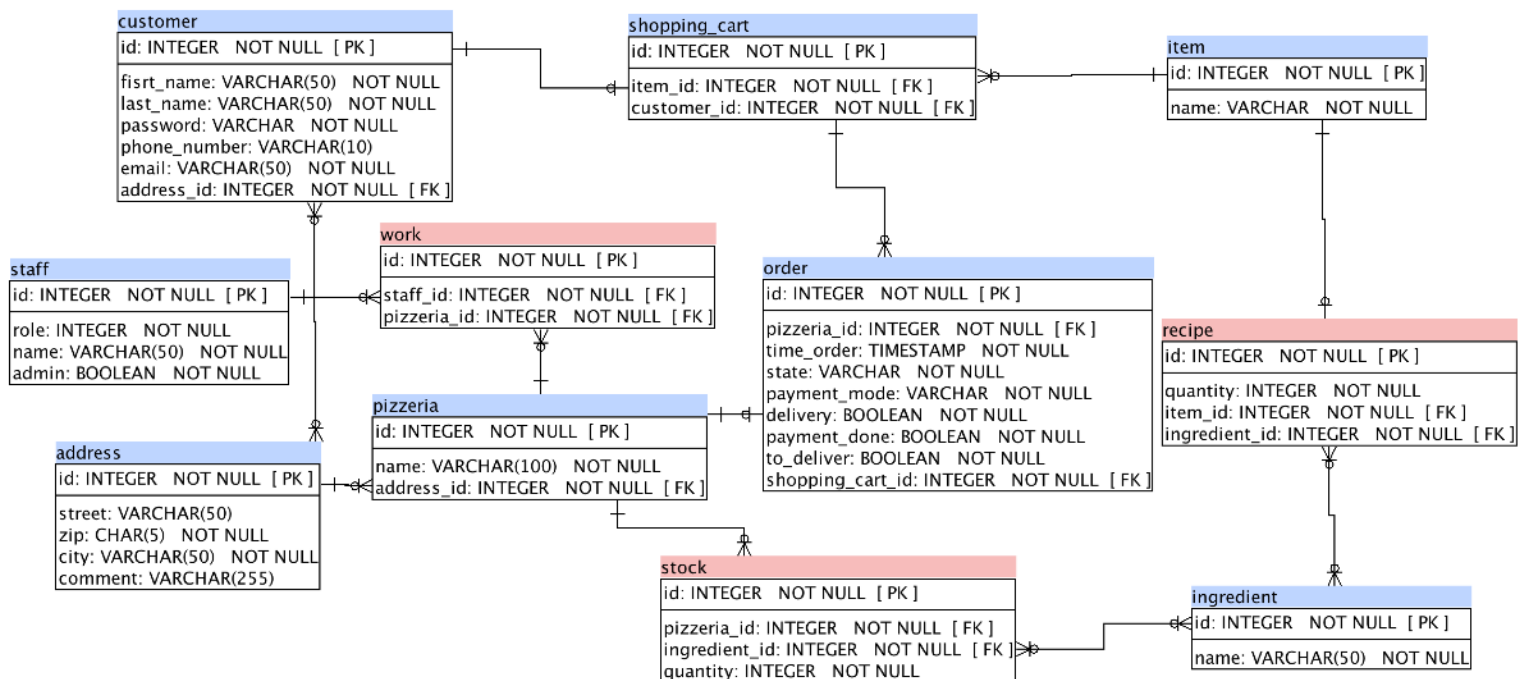
## 4.ARCHITECTURE DE DÉPLOIEMENT

### 4.1.Déploiement de l'application

La solution envisagée prend la forme d'une application web à partir de laquelle les utilisateurs pourront effectuer les différentes actions prévues. Que ce soit la commande par le client, la consultation de la recette par le préparateur de la commande ou encore la consultation des factures les responsables. Il est cependant envisagé de développer en complément une application mobile, destinée au clients et au livreurs pour leurs permettre d'avoir accès à la liste des commandes à livrer ou à la validation d'une livraison ainsi qu'au information nécessaire à la livraison ( adresse et numéro de téléphone du clients ), les livreurs étant itinérants, ce type d'interface leur facilitera la tâche.

### 4.2.Serveur de déploiement

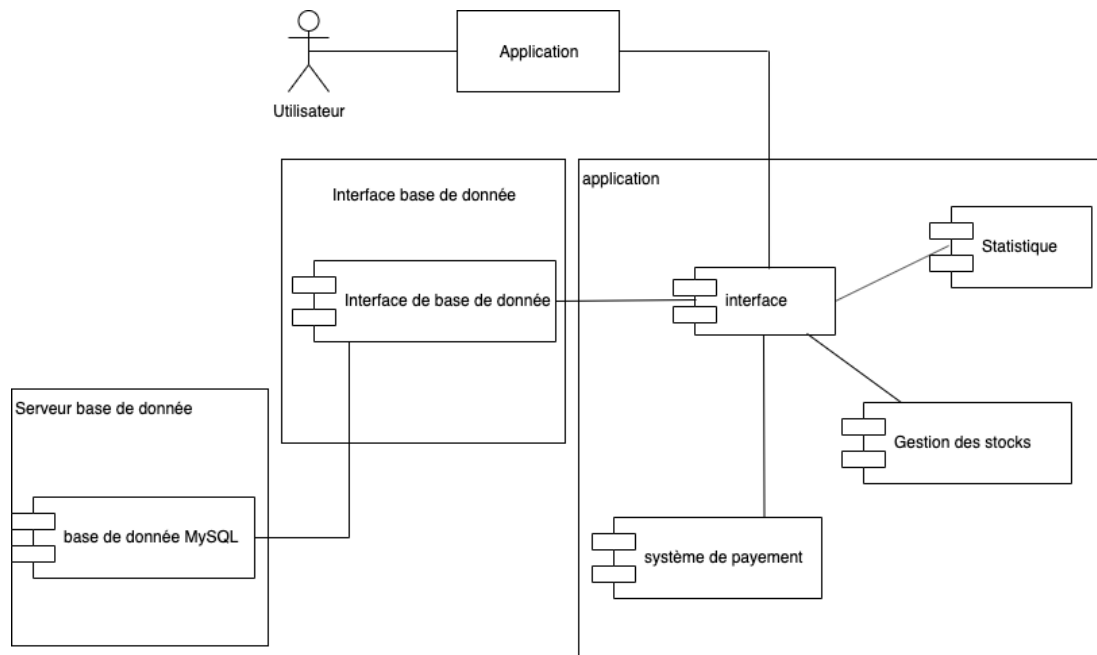
L'application OC Pizza App sera déployée sur les serveurs de Firebase, un service de cloud-computing de type Platform as a Service (PaaS).



## 5.ARCHITECTURE LOGICIELLE

### 5.1.Principes généraux

Le



développement de l'application OC Pizza App se présente sous le forme d'un projet Flutter

L'application sera réaliser suivant le pattern MVC (Model View Controller)

#### 5.1.1.Les couches

L'architecture applicative est la suivante :

- une couche **Vue (view)**
- une couche **Modèle (Model)**
- Une couche **Contrôleur (Controller)**

### 5.1.2. Les modules

Couche	Rôle
Vue	Regroupe les notions de Views et Templates de l'application. Elle représente la partie servant d'interface entre l'utilisateur et l'application
Modèle	Regroupe les classes utilisées par l'application pour manipuler les données
Contrôleur	Regroupe la logique concernant les actions effectuer par l'utilisateur



### 5.1.3. Structure des sources

La structuration des répertoires du projet suit la logique suivante :

```

racine
├── <Model>
│   ├── customer
│   ├── shopping_cart
│   ├── item
│   ├── staff
│   ├── order
│   ├── pizzeria
│   ├── address
│   └── ingredient
├── <Controller>
│   └── src
│       ├── customer
│       │   ├── loginCustomerController
│       │   ├── homeController
│       │   └── menuController
│       └── staff
│           ├── loginStaffController
│           ├── detailPizzaController
│           ├── detailCustomerController
│           ├── detailOrderController
│           └── dataController
└── <View>
    ├── lib
    └── pages
        ├── loginCustommerPage
        ├── homePage
        ├── menuPage
        ├── loginStaffPage
        ├── detailPizzaPage
        ├── detailCustomerPage
        ├── detailOrderPage
        └── dataPage
    
```

...

## 6.POINTS PARTICULIERS

### 6.1.Ressources

#### *6.1.2.Ressources graphique*

Les ressources graphiques permettant de réaliser le design de l'application sont fournies par OC Pizza

#### *6.1.2.Données*

Les données de base (catalogue de pizza, ingrédients, magasin) implémentés dans la base de données de l'application sont également de la responsabilité de la société OC Pizza

### 6.2.Environnement de développement

Le développement de l'application ne requiert pas l'utilisation d'un IDE en particulier. Elle sera développée en utilisant un éditeur de texte au choix des développeurs. (Andoird Studio, Visual Studio sont très bien adapté pour ce project)

### 6.3.Procédure de packaging / livraison

L'application fera l'objet d'un déploiement sur les plateformes Apple Store et Play Store au moment de la livraison finale

Il sera également remis à cette occasion un dossier d'exploitation permettant la continuité de l'utilisation de l'application.

## 7.GLOSSAIRE

<b>Flutter</b>	Framework permettant la réalisation d'application web et mobile en Dart
<b>Dart</b>	Dart est un langage de programmation développé par Google
<b>IDE</b>	Environnement de développement intégré
<b>SGBD</b>	Système de gestion de base de données