



# Web & Mobile Solution

Dossier de conception technique  
Version 1.0

Auteur  
Bernole Yohan  
*Analyste Programmeur*

# TABLE DES MATIÈRES

<b>1.</b>	<b>VERSIONS</b>	<b>3</b>
<b>2.</b>	<b>INTRODUCTION</b>	<b>4</b>
2.1.	Objet du document	
2.2.	Références	
<b>3.</b>	<b>ARCHITECTURE LOGICIELLE</b>	<b>5</b>
3.1.	Base de donnée	
3.1.1.	Modèle physique de données	
3.1.2.	Diagramme de classe	
3.2.	Hébergement de la solution	
3.3.	Service de paiement	
3.4.	Outil de géolocalisation	
<b>4.</b>	<b>ARCHITECTURE DES COMPOSANTS</b>	<b>7</b>
4.1.	Diagramme de composants	
4.2.	Explication textuelle des composants	
<b>5.</b>	<b>ARCHITECTURE DE DÉPLOIEMENT</b>	<b>8</b>
5.1.	Diagramme de déploiement	
5.2.	Explication textuelle des éléments	

## 1. VERSIONS

Auteur	Date	Description	Version
BERNOLE Yohan	28/10/2021	Création du document	1.0

## 2. INTRODUCTION

### 2.1. Domaine fonctionnel

L'entreprise OC Pizza demande 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

Il faut donc intégrer:

- La gestion du compte utilisateur
- La gestion des commandes, de leur réception à leur livraison en passant par leur préparation
- Le stock d'ingrédients restants pour savoir quelles pizzas sont encore réalisables
- Un aide mémoire aux pizzaiolos indiquant la recette de chaque pizza

### 2.2. Objet du document

Le document fonctionnel fait précédemment, nous permet de voir toutes les fonctionnalités qui doivent être présentes dans l'application. Ainsi le présent document constitue le dossier de conception technique de l'application OC Pizza. Il est destiné à l'attention des développeurs, mainteneurs et de l'équipe technique.

L'objectif de l'analyse technique étant de lister les contraintes spécifiques dont les développeurs vont devoir tenir compte pour coder l'application, ce document présentera les langages et les conventions de développement, l'architecture logicielle et de déploiement.

Les éléments du présent dossier découlent:

- de l'analyse des besoins de l'entreprise OC Pizza
- de la rédaction du dossier de conception fonctionnelle

### 2.3. Références

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

- Projet 6 OC Pizza: Document du domaine fonctionnel

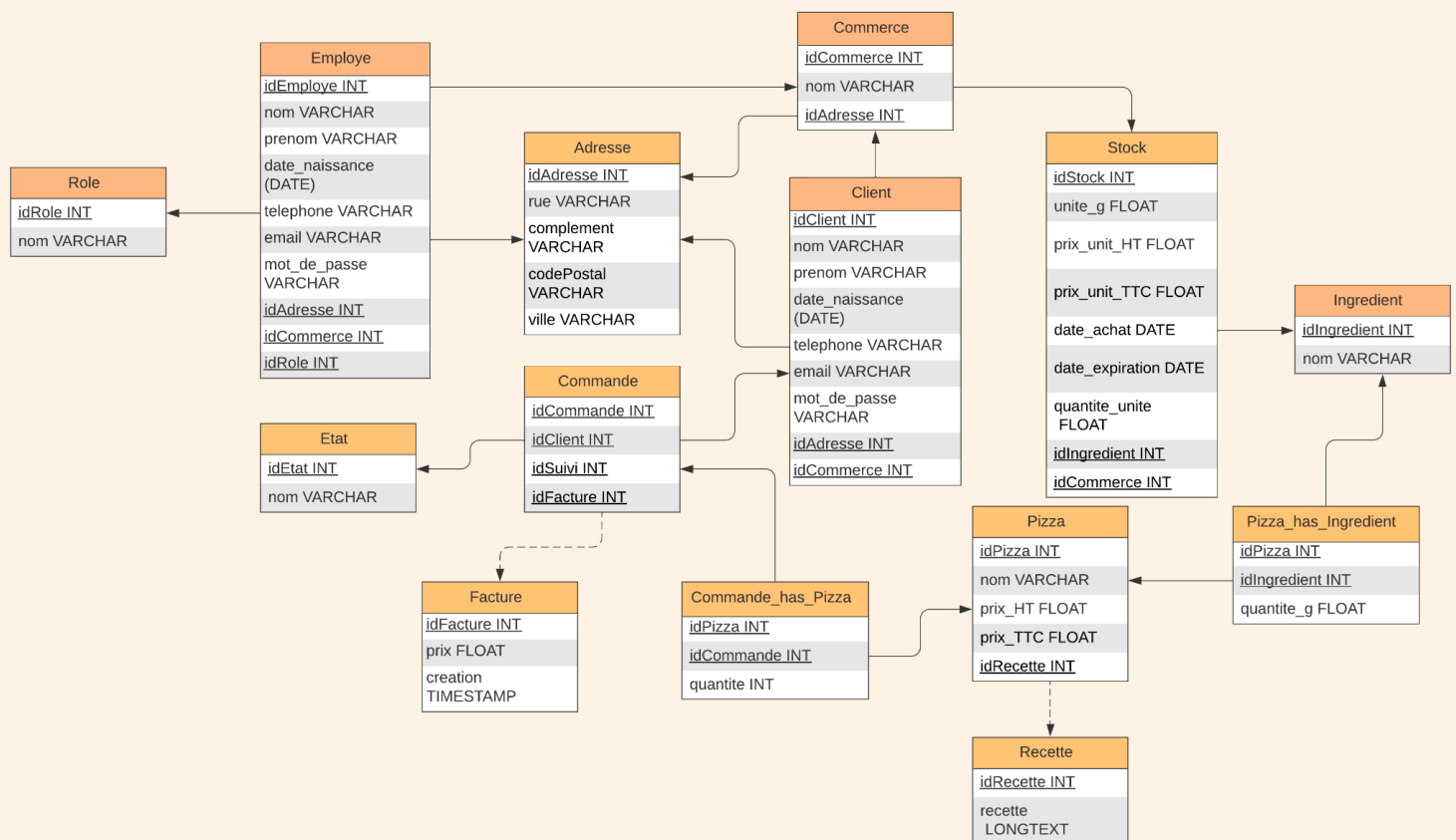
## 3. ARCHITECTURE LOGICIELLE

### 3.1. Base de données

Afin de créer la base de données, plusieurs étapes ont dû être mises en œuvre. De la création du modèle de données en passant par le diagramme de classe jusqu'à la base de données elle-même.

#### 3.1.1. Modèle Physique de données

Le modèle physique de données permet de voir les entités et les associations entre elles. C'est la première ébauche de notre base de données qui provient de l'analyse du domaine fonctionnel.



### 3.1.2. Diagramme de classe

Le diagramme de classe décrit clairement la structure du système en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets. Il reprend le modèle physique de données en donnant des types aux données correspondant aux types SQL et indiquant les cardinalités entre les associations.

*VOIR ANNEXE*

## 3.2. Hébergement de la solution

L'application sera développée en React côté client sur un serveur NodeJS. Il sera hébergé grâce à la solution IONOS pour l'hébergement Web qui comprend 100 Go d'espace, un nom de domaine et la possibilité de créer plusieurs bases de données MySQL avec le logiciel PhpMyAdmin.

<https://www.ionos.fr/hebergement/hebergement-web>

## 3.3. Service de paiement

Le service de paiement sera déployé grâce à la solution Stripe qui permet d'accepter les paiements par carte bancaire.

<https://stripe.com/fr/payments>

Pour déployer la solution Stripe avec React nous pouvons utiliser la documentation correspondante.

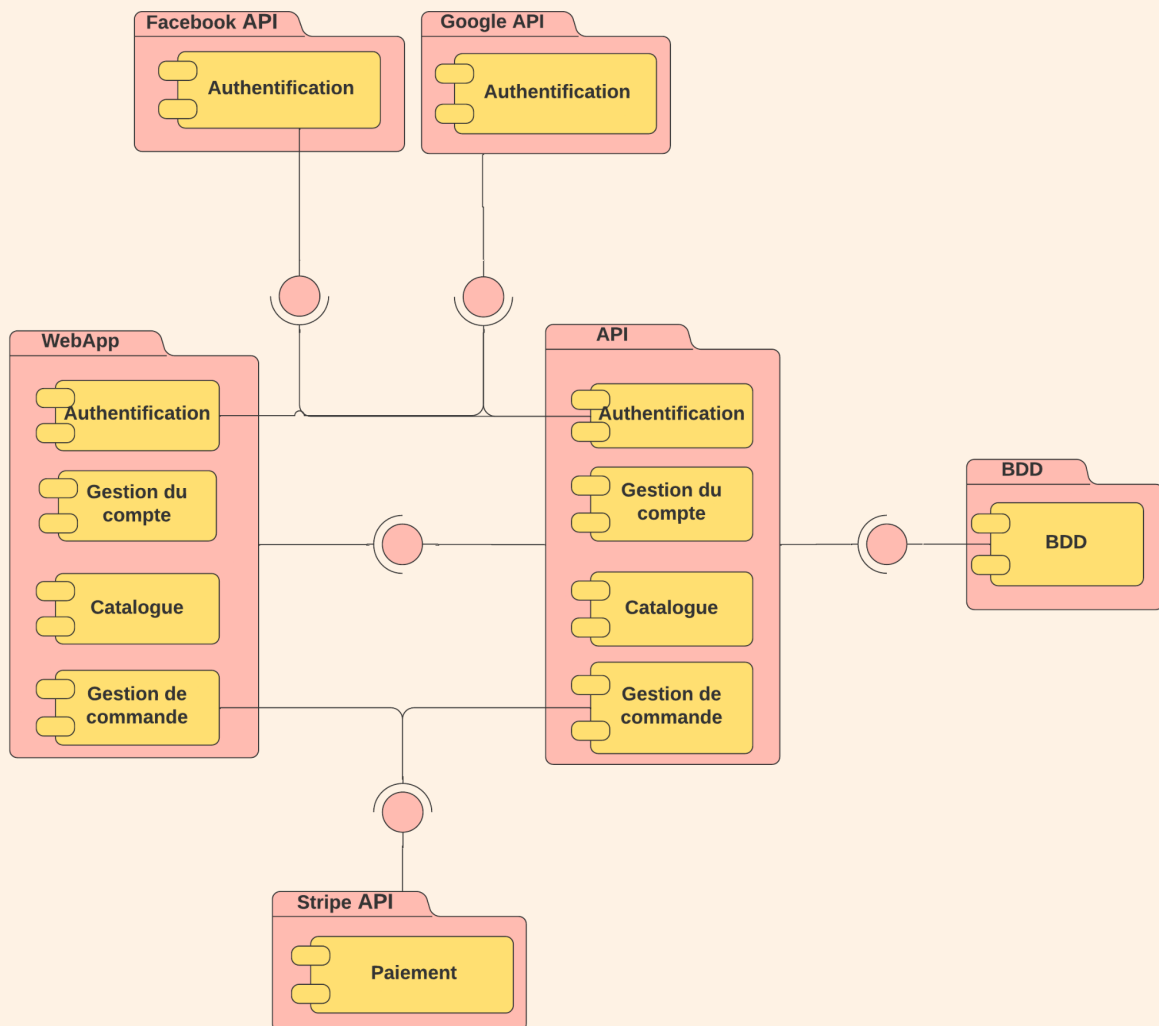
<https://stripe.com/docs/stripe-js/react>

## 3.4. Outil de géolocalisation

En ce qui concerne la géolocalisation des livreurs, pour le consommateur, la solution retenue pour indiquer la position est l'utilisation de Google Map API. L'utilisateur pourra se rendre sur son espace et suivre sa commande.

## 4. Architecture des composants

### 4.1. Diagramme de composants



### 4.2. Explication textuelle des composants

L'utilisateur interagit avec les composants de la web application. L'API de l'application permet d'utiliser les fonctions pour obtenir et/ou modifier les composants dans ma base de données.

Le catalogue représente l'ensemble des produits, la gestion du compte représente l'utilisateur ainsi que la gestion de commande de celui-ci.

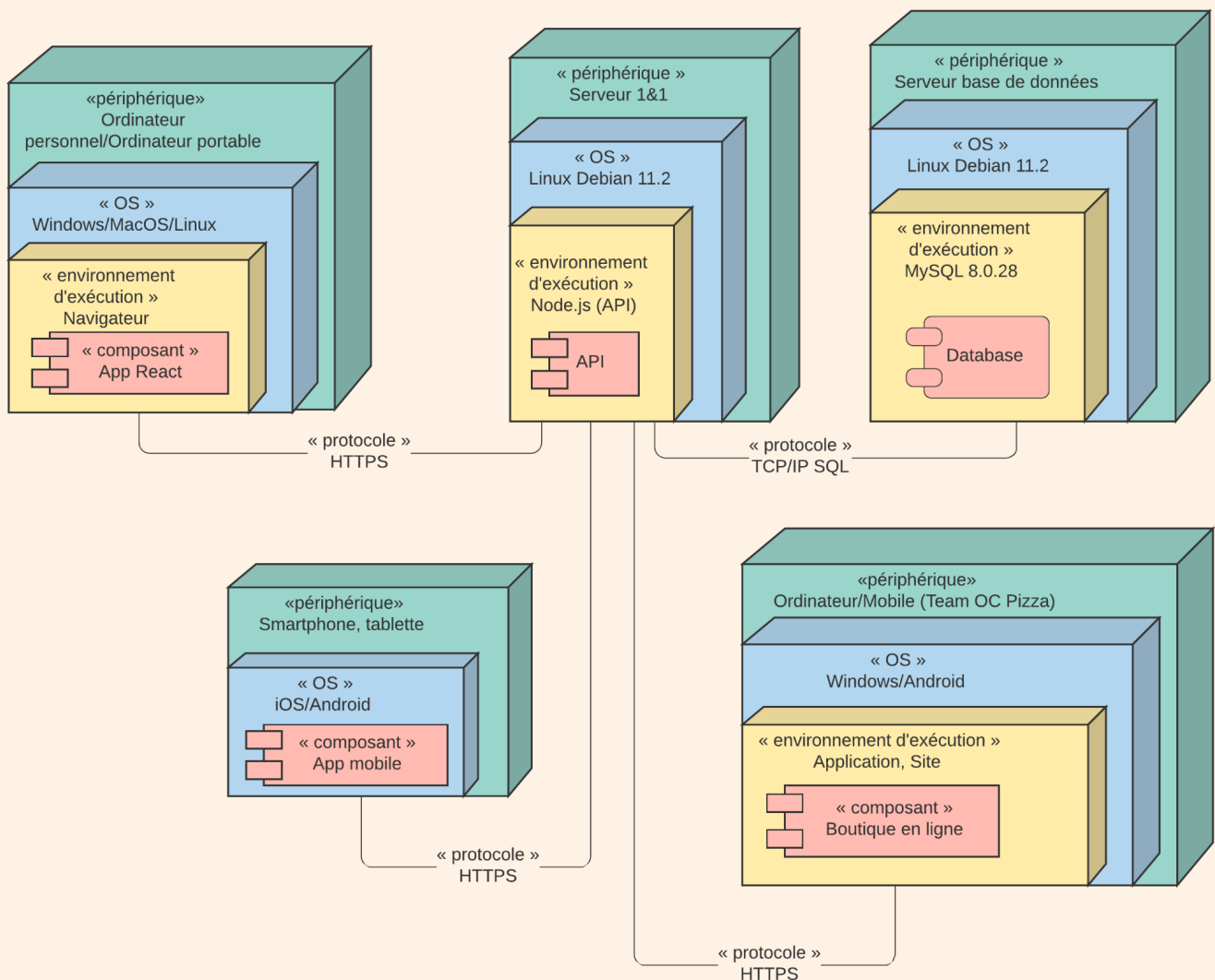
L'authentification peut se faire à partir de la connexion Facebook ou Google mis en place avec les API correspondantes.

Les API génèrent un jeton pour permettre de s'authentifier en relation avec l'application, et tout est stocké en lien avec l'API dans la base de données.

Le paiement est similaire, l'API Stripe permet de pouvoir utiliser le paiement par carte.

## 5. Architecture de déploiement

### 5.1. Diagramme de déploiement





## 5.2. Explication textuelle du déploiement

Le site web sera consultable sur ordinateur, tablette et mobile (iOS & Android) grâce à un navigateur web. L'application est réalisée en React et est hébergée avec la solution IONOS vue ci-dessus sur un serveur Linux Debian avec NodeJS installé sur le serveur.

La solution IONOS permet aussi d'héberger la base de données et de l'utiliser avec l'interface Phpmyadmin en lien avec le système de gestion de BDD MySQL.

Du côté de l'équipe OC Pizza, ils peuvent accéder à l'application avec un navigateur web sur ordinateur ou tablette pour recevoir les commandes, proposer le paiement, les aides mémoires..