|  |  |
| --- | --- |
| **OC Pizza**  **Système de gestion de pizzeria**  **Dossier de conception technique**  Version 2.0 | |
| **Auteur**  Lai Wah TSE  *Analyste-Programmeur* |
|  | |

Table des matières

1 -Versions 3

2 -Introduction 4

2.1 -Objet du document 4

2.2 -Références 4

3 -Architecture Technique 5

3.1 -Composants généraux 5

3.1.1 -Package A 5

3.1.1.1 -Composant X 5

3.1.1.2 -Composant Y 5

3.1.2 -Package B 5

3.1.2.1 -Composant Z 5

3.2 -Application Web 5

3.2.1 -Composants X 5

3.2.2 -Composants Y et Z 5

3.3 -Application XXX... 5

4 -Architecture de Déploiement 6

4.1 -Serveur de Base de données 6

4.2 -Serveur XXX 6

5 -Architecture logicielle 7

5.1 -Principes généraux 7

5.1.1 -Les couches 7

5.1.2 -Les modules 7

5.1.3 -Structure des sources 7

5.2 -Application Web 8

5.3 -Application Xxx 8

6 -Points particuliers 9

6.1 -Gestion des logs 9

6.2 -Fichiers de configuration 9

6.2.1 -Application web 9

6.2.1.1 -Datasources 9

6.2.1.2 -Fichier xxx.yyy 9

6.2.2 -Application Xxx 9

6.3 -Ressources 9

6.4 -Environnement de développement 9

6.5 -Procédure de packaging / livraison 9

6.6 -XXX 9

7 -Glossaire 10

# Versions

|  |  |  |  |
| --- | --- | --- | --- |
| Auteur | Date | Description | Version |
| Lai Wah TSE | 21/12/2022 | Création du document | 1.0 |
| Lai Wah TSE | 05/01/2023 | Finalisation du Document | 2.0 |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Objet du document

* Le présent document constitue le dossier de conception technique de l'application OC Pizza
* Objectif du document : Présenter les solutions techniques pour répondre aux besoins que les co-gérants de la société OC Pizza nous ont exprimé.
* Les éléments du présent dossier découlent :
* ∙ De l’entretien téléphonique entre alexandra, responsable de la société et Lola co-fondatrice de la société OC Pizza
* ∙ Des demandes que les co-gérants nous ont adressées

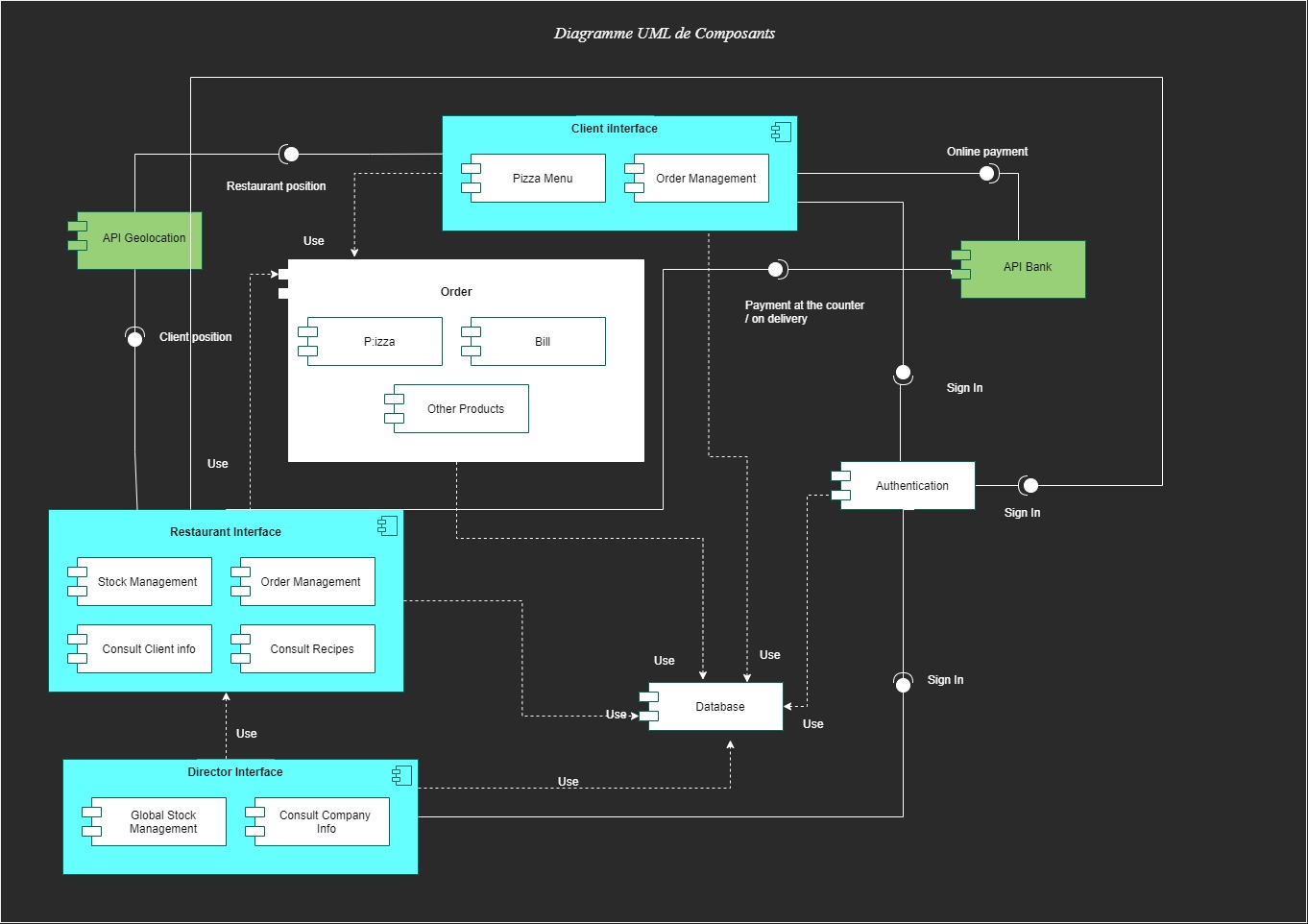
## Références

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

1. **DCF** – 1.0 : Dossier de conception fonctionnelle de l'application
2. **DE** – 1.0 : Dossier d’exploitation de l’application…

# Architecture Technique

## Composants généraux



**Description du composant - Client interface**

Concentre les composants nécessaires à l’utilisateur, le menu avec les différentes pizzas et produits et la gestion de commande (création, panier, information clients…).

**Dépendance avec les autres composants – Client interface -> Composant Order**

L’interface client utilisera le composant Commande pour effectuer une commande qui contiendra les pizzas, ainsi que les possibles autres produits qu’il aura choisi, vérifier l’état de celle-ci ainsi que récupérer la facture.

## Composant : Restaurant interface

**Description du composant - Restaurant interface**

Concentre les composants nécessaires aux différents points de vente, la gestion du stock d’ingrédients par le responsable, la gestion des commandes (update de l’état des commandes, réception des commandes …), consulter les infos clients notamment pour le livreur et consulter les recettes pour les pizzaïolos.

**Dépendance avec les autres composants – Restaurant interface -> Composant Order**

L’interface restaurant aura accès via le composant Commande aux pizzas et produits commandés. La mise à jour de l’état de la commande se fera également via cet échange. Le livreur pourra récupérer les informations clients via le composant Commande.

## Composant : Director interface

**Description du composant - Director interface**

Concentre les composants nécessaires aux Cogérants, la gestion du stock global et consulter les infos générales de la société (chiffre d’affaires, bénéfices, frais, liste des employés …).

**Dépendance avec les autres composants – Director interface -> Restaurant interface**

L’Interface Cogérant pourra récupérer les infos de l’Interface Point de vente (chiffres, employés …).

## Composant : Composant Order

**Description du composant - Composant Order**

Ce composant correspond aux commandes passées par le Client, on y trouve trois autres composants imbriqués, le composant Pizza qui correspond au pizzas commandées ainsi que le composant Other Products qui correspond aux autres produits commandés (boisson, dessert …), le troisième composant est le composant Bill qui contient les informations de paiements.

## Composant : Composant Database

**Description du composant - Composant Database**

Ce composant contient la base de données du système entier, c’est ici que toutes les données sont enregistrées.

**Dépendance avec les autres composants – Composant Database -> Client/Restaurant/Director Interface, Composant Order, Authentication**

La base de données va servir à ces composants pour stocker les informations de leurs composants (Informations du client, informations concernant les commandes et les factures, les ids de connexion des employés et des clients, les informations sur les points de vente, les stocks, les informations de la société…).

## Composant : Composant Authentication

**Description du composant - Composant Authentication**

C’est grâce à ce composant que les clients, les employés et les cogérants pourront se connecter au système et accéder à leurs différentes fonctionnalités/services.

**Dépendance avec les autres composants – Composant Authentication -> Client / Restaurant / Director Interface**

Ce composant est nécessaire pour s’authentifier en tant que client, employé (Responsable, Pizzaïolo et Livreur) ou comme Cogérant. Suite à cette authentification les différents rôles accèderont à leurs fonctionnalités.

## Composant : API Bank

**Description du composant - API Bank**

C’est via cette API que les clients pourront payer leurs commandes soit en ligne avec des services de paiement en ligne (Paypal, Google Pay, carte bancaire …) soit aux guichets via carte bancaire.

**Dépendance avec les autres composants – API Bank -> Restaurant/Client Interface Cette Api**

Il est requise pour le paiement en carte bancaire via le guichet ou à la livraison et le paiement par carte bancaire ou service de paiement en ligne (Paypal, Google Pay…) lors de la commande en ligne.

## Composant : API Geolocation

**Description du composant - API Geolocation**

Grâce à cette API le client pourra accéder au restaurant le plus proche de sa position et les employés pourront voir la position du client, nécessaire pour le livreur.

**Dépendance avec les autres composants – API Geolocation -> Restaurant/Client Interface**

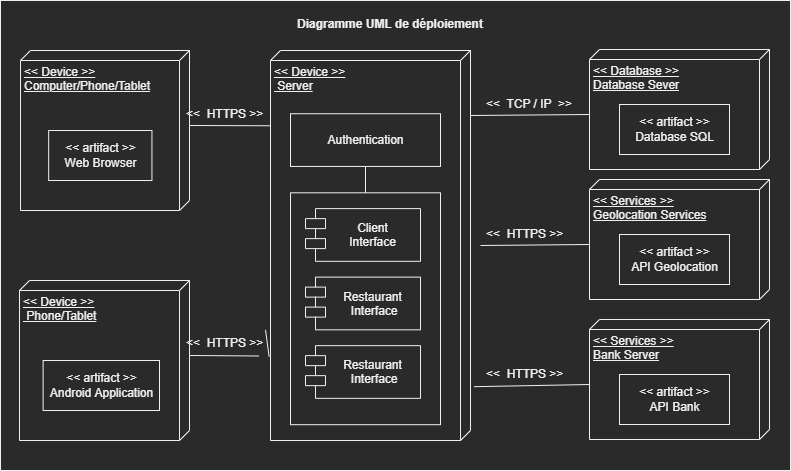
Cette Api est nécessaire pour visualiser la position de l’utilisateur par les employés et la position du restaurant le plus proche pour le client.

## Application Web

La pile logicielle est la suivante :

* ∙ OS : Linux
* ∙ Serveur Web : Apache
* ∙ Base de données : MySQL
* ∙ Interpréteur de script : PHP7

# Architecture de Déploiement



## Descriptions des noeuds

Les différents noeuds possèdent en haut à gauche une annotation sous la forme « stéréotype » on dit que ces nœuds sont stéréotypés, les stéréotypes présents dans notre digramme sont :

* Device qui représente généralement les périphériques mais également des systèmes informatiques comme ici des ordinateurs, tablettes, téléphones …
* Database qui représente les bases de données
* Service qui rend service à un autre noeud (exemple Banks Server rends service au noeud Server)

Les noeuds communiquent entre eux grâce à différents protocoles, ici HTTPS (HyperText Transfer Protocol Secure) qui permet une communication entre un client et un serveur contenant des données de manière sécurisé grâce à un certificat d’authentification émis par une autorité tierce par exemple Google, c’est le protocole par défaut pour les pages web avec le http. L’autre protocole utilisé ici est le TCP/IP qui est le protocole par défaut pour des transports de données.

### <<Device>>Computer/Phone/Tablet

Appareils accédant au service par le biais du site internet via un navigateur Web (Firefox, Chrome …). Échange des données sous le protocole HTTPS avec le serveur.

### <<Device>>Phone/Tablet

Appareils accédant au service par le biais de l’Application mobile. Échange des données sous le protocole HTTPS avec le serveur.

### <<Device>>Server

Le cœur du système, c’est le système informatique d’OC Pizza, après l’authentification l’utilisateur récupéra l’interface qui lui est dédié. Il pourra par exemple être stocké via un hébergeur comme OVH ou un serveur physique dans les locaux d’OC Pizza.

### <<Database>>Database Server

La base de données du système où seront enregistrées toutes les données de celui-ci. Cette base de données sera stockée sur un service externe (serveur). Échange des données sous le protocole TCP/IP avec le serveur.

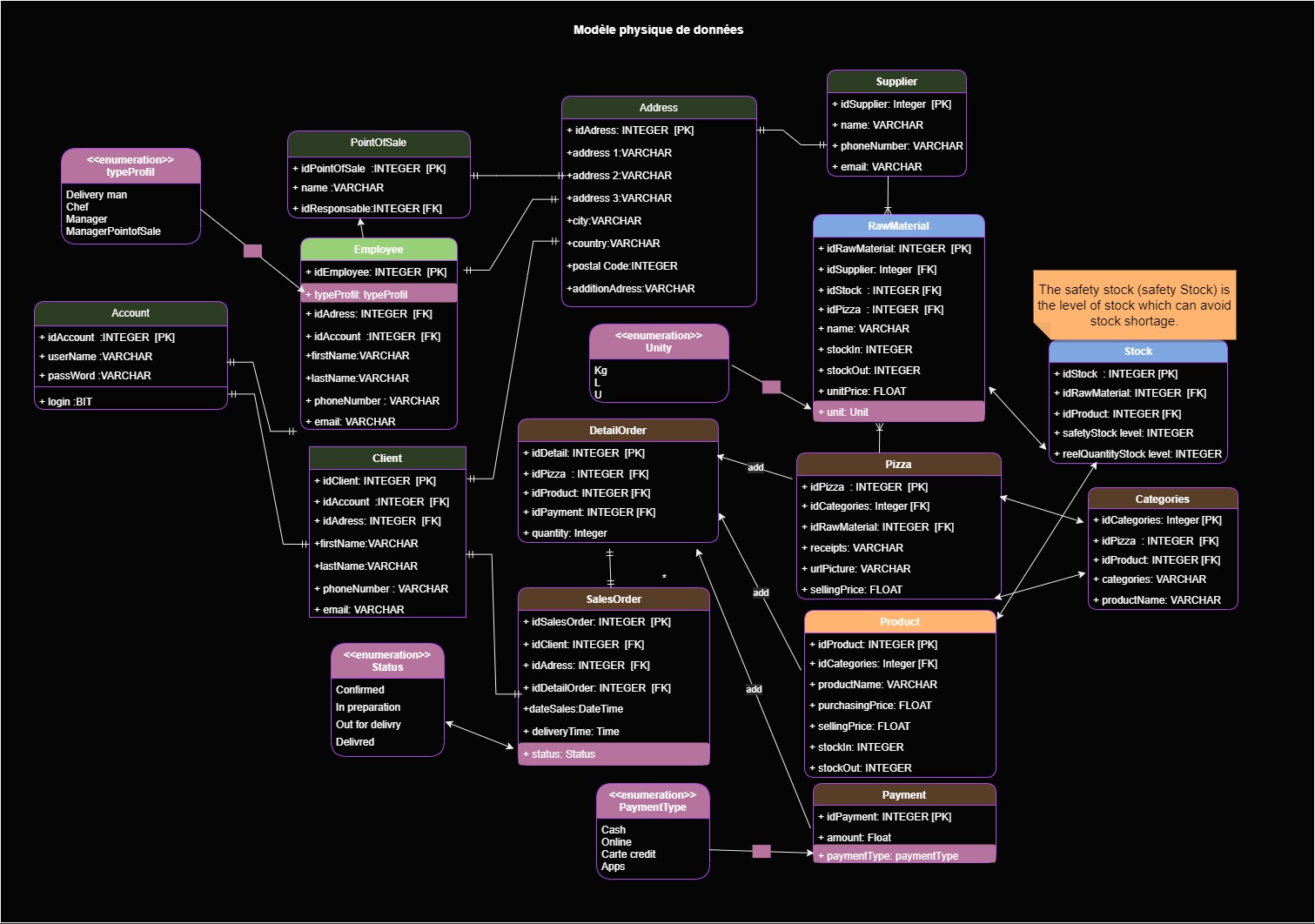
### <<Service>>Geolocation Service

Le service de géolocalisation utilisera un service externe comme google maps. Les échanges se feront sous le protocole HTTPS.

### <<Service>Bank Server

Le service bancaire sera déployé via une Api externe. Les échanges des données sont sous le protocole HTTPS avec le serveur.

## Serveur de Base de données



Dans la méthode Merise, le modèle physique des données (MPD) consiste à implanter une base de données dans un Système de gestion de base de données. Le langage utilisé pour ce type d'opération est le SQL. Un système de gestion de base de données (abr. SGBD) est un [logiciel système](https://fr.wikipedia.org/wiki/Logiciel_syst%C3%A8me) servant à stocker, à manipuler ou gérer, et à partager des données dans une [base de données](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es), en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations.

Par exemple, utilisez une Primary key (PK) et une Foreign key (FK) qui permettent le référencement croisé des données associées entre les tables, et des contraintes de Foreign key, qui aident à maintenir la cohérence des données associées.

La base de données du système est l’endroit où seront enregistrées toutes les données de celui-ci, elle fonctionnera sous MySQL 8.0 qui est un service de base de données gratuite et Open Source. Cette base de données sera stockée sur un service externe (serveur OVH à hébergement mutualisé). Échange des données sous le protocole TCP/IP avec le serveur.

## Serveur de l’Application

Serveur contenant notre application stockée sur un serveur hébergé chez OVH sous hébergement mutualisé et proposant une certification SSL. Notre serveur sera programmé dans le langage PHP7 est un langage généralisé et Open Source spécialement conçu pour les applications web.