



AB Solution

OC Pizza

Pizzapp

Dossier de conception technique

Version 1.1

Auteur

PANCARTE Marjorie
Analyste Développeuse



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 - Application Web.....	5
3.2 - Base de données	5
4 - Architecture de Déploiement	6
4.1 - Serveur de Base de données et déploiement	6
4.2 - Principes généraux.....	6
4.2.1 - <i>Application java</i>	6
4.2.2 - <i>Les couches</i>	7
5 - Points particuliers.....	8
5.1 - Fichiers de configuration	8
5.1.1 - <i>Application web</i>	8
5.2 - Ressources	8
5.3 - Environnement de développement	8
5.4 - Procédure de packaging / livraison.....	8
6 - Glossaire	9



1 - VERSIONS

Auteur	Date	Description	Version
PM	30/07/2019	Création du document	1.0



2 - INTRODUCTION

2.1 - Objet du document

Le présent document constitue le dossier de conception technique de l'application XXX...

Objectif du document...

Les éléments du présents dossiers découlent :

De l'entretien réalisé avec le dirigeant de la société OC PIZZA du 01/12/2017

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. **P8-DCF – 1.0** : Dossier de conception fonctionnelle de l'application
2. **P8 - DE – 1.1** : Dossier d'exploitation de l'application.



3 - ARCHITECTURE TECHNIQUE

3.1 - Application Web

Le développement de l'application web sera fait en java J2EE avec l'utilisation de framework Spring et thymeleaf.

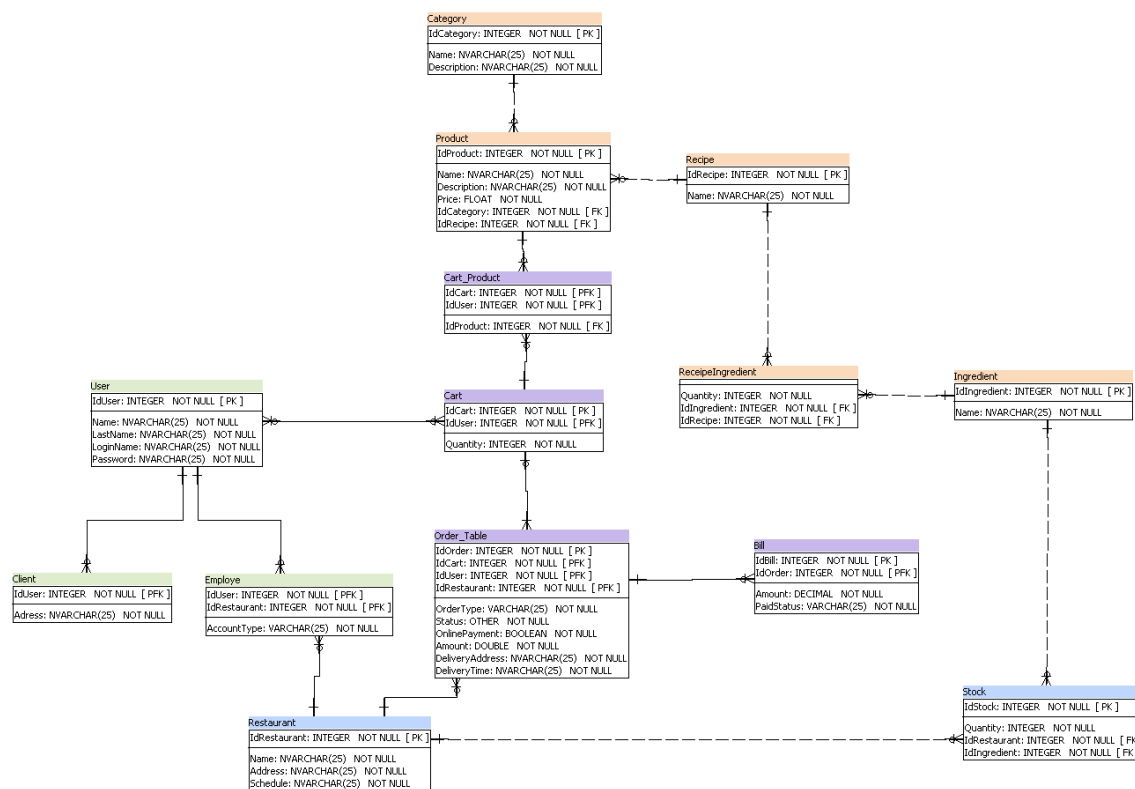
3.2 - Base de données

3.2.1 - SGBD Utilisé

Le Système de gestion de base de données utilisé sera MySQL. Ce choix se justifie en raison de sa simplicité et ses performances.

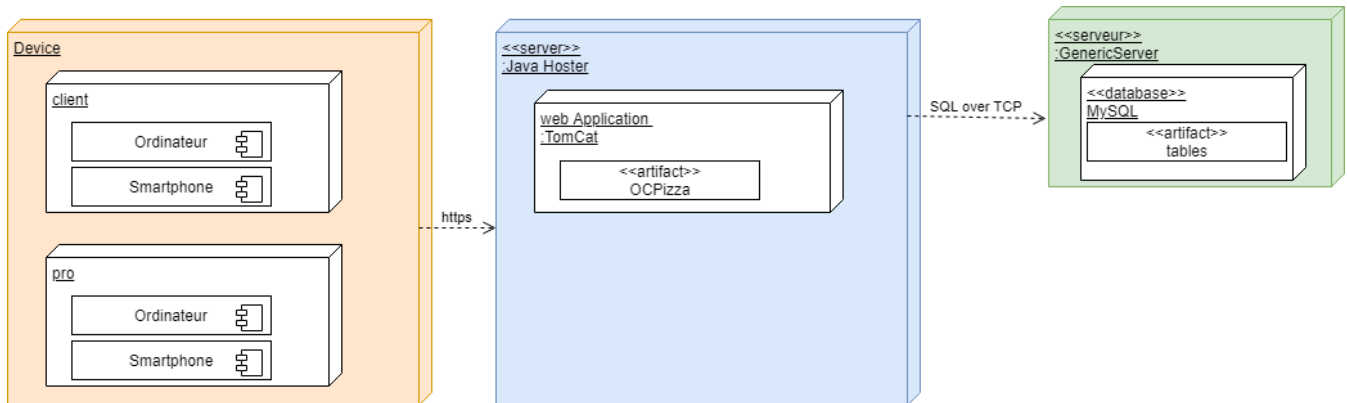
3.2.2 - Modèle physique de données

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



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 par le dirigeant. Il est cependant envisagé de développer en complément une application mobile, destinée principalement aux livreurs pour le permettre d'avoir accès à la liste des commandes à livrer ou à la validation d'une livraison. En effet, les livreurs étant itinérants, ce type d'interface leur facilitera la tâche.

4.1 - Serveur de Base de données et déploiement

L'application PIZZAPP sera déployée sur les serveurs de Heroku, un service de cloudcomputing de type Plateforme as a Service (PaaS).

Cette solution semble adaptée au projet et permettra de faciliter le déploiement et la gestion de application. Architecture logicielle

4.2 - Principes généraux

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

4.2.1 - Application java

Le développement de l'application PIZZAPP se présente sous la forme d'un « Projet » Java J2ee avec Spring comme Framework et thymeleaf l'intégration d'un module de paiement est mis en place avec la banque du propriétaire.



4.2.2 - Les couches

L'architecture Spring MVC est la suivante :

- une couche **configuration** :
- une couche **controler**
- une couche **repository**
- une couche **service**
- une couche **ressources** :

La structuration des répertoires du projet suit la logique suivante :

- les répertoires sources sont créés de façon à respecter la philosophie Maven (à savoir : « convention plutôt que configuration »)

```
racine
├── pom.xml
├── README.md
├── src
│   ├── main
│   │   ├── java
│   │   │   ├── configuration
│   │   │   ├── controler
│   │   │   ├── repository
│   │   │   ├── service
│   │   │   └── model
│   │   └── resources
│   │       ├── static
│   │       └── templates
```



5 - POINTS PARTICULIERS

5.1 - Fichiers de configuration

5.1.1 - Application web

...Le fichier suivant src\main\resources\application.properties devra être configuré avant le déploiement

```
spring.datasource.url=jdbc:mysql://localhost:5432/pizza
spring.datasource.username=root
spring.datasource.password=root
server.port=8080
```

```
spring.mail.host=smtp.gmail.com
spring.mail.port=587
spring.mail.username=exemple@gmail.com
spring.mail.password=password
```

...

5.2 - Ressources

- Ressources graphiques

Les ressources graphiques permettant de réaliser le design de l'application web sont fournies par OC PIZZA.

- Données

Les données de base (catalogue de pizzas, ingrédients, magasin) implémentées dans la base de données de l'application sont également de la responsabilité de la société OC PIZZA.

5.3 - Environnement de développement

Le développement de l'application ne requiert pas l'utilisation d'un IDE spécifique.

5.4 - Procédure de packaging / livraison

L'application fera l'objet d'un déploiement sur la plateforme heroku au moment de la livraison finale et un *dossier d'exploitation* permettant la continuité de l'utilisation de l'application. L'utilisation de cette plateforme pour cette app ne requiert pas des performances très élevées donc nous pouvons nous satisfaire de l'offre « hobby ».

L'<https://devcenter.heroku.com/categories/java-support>



AB Solution

6 - GLOSSAIRE
