

# OC PIZZA

## Nouveau Système Informatique d'OC Pizza

Dossier d'exploitation

Version 1.0

**Auteur**  
Emmanuel Plumas  
Développeur Java Junior

## TABLE DES MATIERES

<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 - Pre-requis.....</b>	<b>5</b>
3.1 - Système.....	5
3.1.1 - Caractéristiques techniques.....	5
3.2 - Serveur applicatif de Base de données.....	5
3.3 - Serveur applicatif Web.....	5
<b>4 - procédure de déploiement .....</b>	<b>6</b>
4.1 - Déploiement de l'Application Web.....	6
4.1.1 - Artefacts.....	6
4.1.2 - Environnement de l'application Web .....	6
4.1.3 - Répertoire de configuration applicatif .....	6
4.1.4 - Vérification .....	6
4.2 - Déploiement de la Base de données.....	7
4.2.1 - Artefacts.....	7
4.2.2 - Déploiement.....	7
4.2.3 - Environnement de la base de données .....	8
4.2.4 - Vérification .....	8
<b>5 - Procédure de démarrage/arrêt.....</b>	<b>9</b>
5.1 - Base de données .....	9
5.2 - Application Web .....	9
<b>6 - Procédure de mise à jour .....</b>	<b>10</b>
6.1 - Base de données .....	10
6.2 - Application Web .....	10
<b>7 - Supervision/Monitoring.....</b>	<b>11</b>
7.1 - Supervision de l'application Web .....	11
<b>8 - Procédure de sauvegarde et de restauration.....</b>	<b>12</b>
8.1 - Sauvegarde et restauration de la base de données .....	12
8.1.1 - Sauvegarde de la base de données :.....	12
8.1.2 - Restauration de la base de données :.....	12
8.2 - Sauvegarde et restauration de l'application web .....	12
8.2.1 - Sauvegarde de l'application web .....	12
8.2.2 - Restauration de l'application web .....	12

# 1 - VERSIONS

Auteur	Date	Description	Version
Emmanuel Plumas	24/09/2020	Création du document	1.0

## 2 - INTRODUCTION

### 2.1 - Objet du document

Le présent document constitue le dossier d'exploitation de l'application « OC Pizza ».  
Il a pour objectif de permettre aux équipes « delivery » de disposer de tous les éléments nécessaires à la mise en production de l'application au sein de la société « OC Pizza ».

### 2.2 - Références

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

1. **P8DOCPizza\_01\_fonctionnel** : Dossier de conception fonctionnelle de l'application
2. **P8DOCPizza\_02\_technique** : Dossier de conception technique de l'application

## 3 - PRE-REQUIS

### 3.1 - Système

La solution OCPizza sera déployé sur un serveur unique hébergé dans le cloud.

Au regard de la nature et du volume des requêtes, ainsi que de leur fréquence, un unique serveur physique hébergera le serveur de base de données ainsi que le serveur web.

Cette solution :

- garantie le coût le plus bas pour une architecture simple. (Moins de 30€HT/mois).
- évite d'avoir à exposer une adresse internet via serveur DNS. (La solution d'hébergement cloud nous permet d'associer l'hébergement avec l'adresse internet).
- permet à OCPizza de ne pas avoir à effectuer la maintenance du serveur, qui incombe alors à l'hébergeur.
- garantie à OCPizza la gestion matériel (obsolescence, remplacement,...) du serveur par l'hébergeur.

#### 3.1.1 - Caractéristiques techniques

La configuration minimale est la suivante :

- Serveur Monoprocasseur 8 cœurs type Intel Xeon E-2233 à 3,6 GHz
- Stockage : 160 Go SSD
- RAM : 8

A titre d'exemple, l'offre [OVH VPS Elite](#) remplit ces conditions.

### 3.2 - Serveur applicatif de Base de données

Le serveur de Base de données sera un serveur PostgreSQL.  
Version :11.9

### 3.3 - Serveur applicatif Web

Le serveur web sera un serveur Apache TOMCAT, packagé au sein de l'artefact.

## 4 - PROCÉDURE DE DÉPLOIEMENT

### 4.1 - Déploiement de l'Application Web

#### 4.1.1 - Artefacts

L'artefact est le fichier :

`p8JavaOcPizza-0.0.1-SNAPSHOT.jar`

Le nom de cet artefact sera amené à évoluer. Il s'agit ici d'une version de développement (SNAPSHOT). La livraison au client pour la mise en production se fera avec une version RELEASE.

Cet artefact est récupérable sur le repository github du projet à l'adresse suivante :

<https://github.com/eplms/p8JavaOcPizza>

L'artefact doit être chargé sur le serveur dédié.

#### 4.1.2 - Environnement de l'application Web

La **version 8 du JRE** (Java Runtime Environment), téléchargeable à l'adresse suivante, devra être installée sur le serveur web :

<https://www.oracle.com/java/technologies/javase-jre8-downloads.html>

#### 4.1.3 - Répertoire de configuration applicatif

Il n'y a pas de répertoire de configuration pour cette application car toute la configuration est embarquée dans le jar et n'est pas prévue pour être modifiée.

#### 4.1.4 - Vérification

La vérification du déploiement de l'application se fait en se connectant à l'URL de la page d'accueil de l'application :

`http://urldusiteOcPizza`

NB : La vérification du déploiement ne pourra se faire qu'une fois l'application ET la base de données déployés.

## 4.2 - Déploiement de la Base de données

### 4.2.1 - Artefacts

L'artefact à récupérer est le script de création de base de données, qui permettra la création de la structure de la base de données.

```
Script_ p8_Java_Oc_Pizza.sql
```

Cet artefact est récupérable sur le repository github spécifique au script, à l'adresse suivante :

```
https://github.com/eplms/p8JavaOcPizzaScriptSql.git
```

### 4.2.2 - Déploiement

Les commandes diffèrent selon que la variable d'environnement Pgdata a été renseignée lors de l'installation de PostgreSQL.

Partant du principe que le serveur de base de données est installé, le déploiement se fait en suivant les étapes suivantes :

#### a- Lancement du serveur de base de données

Si la variable d'environnement Pgdata a été renseignée lors de l'installation, le lancement du serveur se fait grâce à la commande

```
postgres
```

Sinon, le lancement du serveur se fait grâce à la commande :

```
postgres -d /usr/local/pgsql/data
```

#### b- Création de la base de données dans le serveur de base de données

La création de la base de données se fait grâce à la commande :

```
createdb db_p8_ocpizza
```

ou, la commande suivante, si la variable d'environnement Pgdata n'a pas été renseignée à l'installation

```
/usr/local/pgsql/bin/createdb db_p8_ocpizza
```

Il est important de respecter le nom de la base de données (ici db\_p8\_ocpizza). Celui-ci est configurée avec cet orthographe dans l'application.

#### c- Lancement du script de création de base de données à partir de l'artefact

Il convient d'accéder à la base de données créée, à l'aide de la commande :

```
psql db_P8_ocpizza
```

Le lancement du script de création de la structure de la base de données se fait ensuite grâce à la commande :

```
psql -f Script_p8_Java_Oc_Pizza.sql
```

**Autre Méthode de déploiement possible :**

Le déploiement de la base de données peut également se faire par l'intermédiaire de **Pgadmin4**, application proposant une interface visuelle. Pgadmin4 est téléchargeable à l'adresse suivante :

<https://www.pgadmin.org/download/>

Le lancement du serveur de base de données se fera comme dans la méthode précédente. La suite du processus (création de la base de données et lancement du script de création de base de données) se fera à l'aide de Pgadmin4.

### 4.2.3 - Environnement de la base de données

Le serveur de base de données PostgreSQL 11, téléchargeable à l'adresse suivante, devra être installé sur le serveur.

<https://www.postgresql.org/download/>

### 4.2.4 - Vérification

La vérification du déploiement de la base de données, se fait en suivant les étapes suivantes :

a- [Connexion à la base de données](#)

```
psql db_P8_ocpizza
```

b- [Interrogation visant à lister les tables présentes dans la base de données](#)

```
\d
```

La vérification du déploiement de la base de données peut également se faire grâce à l'application **pgadmin4**, téléchargeable à l'adresse suivante :

<https://www.pgadmin.org/download/>

Enfin, la vérification du déploiement de la base de données ET de l'application se fait en se connectant à l'URL de la page d'accueil de l'application :

<http://urldusiteOcPizza>

NB : La vérification du déploiement global ne pourra se faire qu'une fois l'application ET la base de données déployés.



## 5 - PROCÉDURE DE DÉMARRAGE/ARRÊT

### 5.1 - Base de données

#### **Pour démarrer la base de données :**

```
pg_ctl start
```

(Dans le cas où la variable d'environnement est renseignée à l'installation)

```
pg_ctl -D /usr/local/pgsql/data start
```

(Dans le cas contraire)

#### **Pour arrêter la base de données**

```
pg_ctl stop
```

(Dans le cas où la variable d'environnement est renseignée à l'installation)

```
pg_ctl -D /usr/local/pgsql/data stop
```

(Dans le cas contraire)

### 5.2 - Application Web

#### **Pour démarrer l'application Web :**

La commande pour lancer l'application est :

```
java -jar p8JavaOcPizza-0.0.1-SNAPSHOT.jar
```

#### **Pour arrêter l'application Web :**

Il suffit de fermer la console.

## 6 - PROCEDURE DE MISE A JOUR

### 6.1 - Base de données

La mise à jour de la base de données se fait en exécutant le script correspondant à la nouvelle version de la base de données, dans la base de données existantes.

Il s'agira donc de saisir dans la base de données la commande de lancement de script suivante :

```
psql -f scriptNouvelleBase.sql
```

### 6.2 - Application Web

La mise à jour de l'application web se fait en :

- Supprimant le fichier jar existant
- Téléchargeant le fichier jar correspondant à la version mise à jour
- Lançant l'application correspondant à ce **nouveau fichier** jar en saisissant, la commande

```
java -jar p8JavaOcPizza-0.0.2-SNAPSHOT.jar
```

## 7 - SUPERVISION/MONITORING

### 7.1 - Supervision de l'application Web

Un système de monitoring est mis à disposition. Pour ce faire, un module de Spring, Spring Boot Actuator, est intégré à l'application. Celui-ci permet d'exposer plusieurs endpoints permettant de surveiller l'application. Parmi les endpoints les plus courants, nous retrouvons :

- « /health »
- « /metrics »
- « /sessions »

#### **Statut de l'application : /health**

Le statut de l'application est consultable via le endpoint « /health ».

Pour connaître le statut (up ou down) de l'application, il suffit donc d'appeler l'URL suivante :

<http://urldusiteOcPizza/actuator/health>

#### **Paramètres de l'application : /metrics**

Le endpoint « /metrics » renseigne sur les paramètres de l'application : mémoire, données sur Tomcat,...

Pour accéder à ces renseignements, il suffit d'appeler l'URL suivante :

<http://urldusiteOcPizza/actuator/metrics>

#### **Sessions : /sessions**

Le endpoint « /session » permet de connaître les sessions http utilisées.

Pour connaître ces sessions, il suffit donc d'appeler l'URL suivante :

<http://urldusiteOcPizza/actuator/sessions>

#### **Autres endPoints :**

La liste des endpoints directement disponibles ainsi que l'ensemble des informations relatives au monitoring sont consultables sur la documentation officielle de Spring Boot actuator à l'adresse suivante :

<https://docs.spring.io/spring-boot/docs/current/reference/html/production-ready-features.html#production-ready-endpoints>

Enfin, il est également possible de créer des endpoints en fonction de besoins particuliers.

## 8 - PROCEDURE DE SAUVEGARDE ET DE RESTAURATION

### 8.1 - Sauvegarde et restauration de la base de données

#### 8.1.1 - Sauvegarde de la base de données :

La sauvegarde de la base de données se fait à l'aide de la commande :

```
pg_dump db_p8_ocpizza > nomDuFichierDeSauvegarde.sql
```

Le fichier de sauvegarde contient alors un script SQL.

#### 8.1.2 - Restauration de la base de données :

La restauration de la base de données en utilisant un script SQL précédemment sauvegardé se fait à l'aide de la commande :

```
psql -f nomDuFichierDeSauvegarde.sql
```

### 8.2 - Sauvegarde et restauration de l'application web

#### 8.2.1 - Sauvegarde de l'application web

La sauvegarde de l'application web se fait en sauvegarder le fichier jar de l'application :

```
p8JavaOcPizza-0.0.1-SNAPSHOT.jar
```

#### 8.2.2 - Restauration de l'application web

La restauration de l'application web se fait en relançant l'application à partir du **nouveau fichier** jar selon l'instruction ci-dessous.

Il est nécessaire de saisir la commande :

```
java -jar p8JavaOcPizza-0.0.2-SNAPSHOT.jar
```