

OC PIZZA

Documentez votre système de gestion de pizzeria

Dossier d'exploitation

Version 1.0

Auteur

Mehdi BICHARI

Développeur

TABLE DES MATIÈRES

1 - Versions.....	4
2 - Introduction.....	5
2.1 - Objet du document.....	5
2.2 - Références.....	5
3 - Pré-requis.....	6
3.1 - Système.....	6
3.1.1 - Serveur de Base de données.....	6
3.1.1.1 - Caractéristiques techniques.....	6
3.1.2 - Serveur Web.....	6
3.1.2.1 - Caractéristiques techniques.....	6
3.1.3 - Serveur d'application.....	6
3.1.3.1 - Caractéristiques techniques.....	6
3.2 - Bases de données.....	7
3.3 - Web-services.....	7
4 - Procédure de déploiement.....	8
4.1 - Déploiement du serveur de base de données.....	8
4.1.1 - Environnement.....	8
4.1.2 - Artefacts.....	8
4.1.3 - Vérifications.....	9
4.2 - Déploiement du Serveur d'Application.....	10
4.2.1 - Artefacts.....	10
4.2.2 - Environnement de l'application web.....	10
4.2.3 - Fichiers de configurations.....	10
4.2.3.1 - ocpizza-gunicorn.conf.....	10
4.2.3.2 - production.py.....	11
4.2.4 - Changements de variables.....	11
4.2.5 - Vérifications.....	12
4.3 - Déploiement du serveur Web.....	13
4.3.1 - Artefacts.....	13
4.3.2 - Configuration.....	13
Vous pouvez rajouter des noms de domaines pour votre site directement dans ce fichier de configuration.....	13
4.3.3 - Vérifications.....	13
5 - Procédure de démarrage / arrêt.....	14
5.1 - Serveur web.....	14
5.1.1 - Démarrage.....	14
5.1.2 - Arrêt.....	14
5.2 - Serveur d'application.....	14
5.2.1 - Démarrage.....	14
5.2.2 - Arrêt.....	15
5.3 - Serveur de Base de données.....	15
5.3.1 - Démarrage.....	15
5.3.2 - Arrêt.....	15
6 - Procédure de mise à jour.....	16
7 - Supervision/Monitoring.....	17
7.1 - Supervision de l'application web.....	17
8 - Procédure de sauvegarde et restauration.....	18
8.1 - Mise en place de la stratégie de sauvegarde.....	18
8.2 - Restauration de sauvegardes.....	18
8.2.1 - Remontée de la base de production.....	18
8.2.2 - Remontée de toutes les bases.....	18

1 - VERSIONS

Auteur	Date	Description	Version
Mehdi	26/10/2020	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 présente les informations dont l'équipe d'exploitation a besoin pour pouvoir assurer une exploitation en règle du système.

2.2 - Références

Pour de plus amples informations, se référer :

1. **DCT - OC Pizza** : Dossier de conception technique de l'application
2. **DCF - OC Pizza** : Dossier de spécification fonctionnelle de l'application

3 - PRÉ-REQUIS

3.1 - Système

3.1.1 - Serveur de Base de données

Serveur de base de données hébergeant la base données de l'application.

3.1.1.1 - Caractéristiques techniques

Linux CentOS 7

Nécessite l'installation de MySQL 8.0.19

Pré-requis matériel :

- **RAM** : 16Go
- **Stockage** : 500Go extensible en RAID 10
- **Processeur** : 3,5 Ghz 12 coeurs

3.1.2 - Serveur Web

Serveur physique ou virtuel responsable de recevoir les requêtes externes.

3.1.2.1 - Caractéristiques techniques

Linux CentOS 7

Nécessite NGINX 1.17.0

Pré-requis matériel :

- **RAM** : 8Go
- **Stockage** : 100Go
- **Processeur** : 2,7 Ghz 8 coeurs

3.1.3 - Serveur d'application

Serveur hébergeant l'application Python(Django).

3.1.3.1 - Caractéristiques techniques

Linux CentOS 7

Nécessite Python 3.7

Pré-requis matériel :

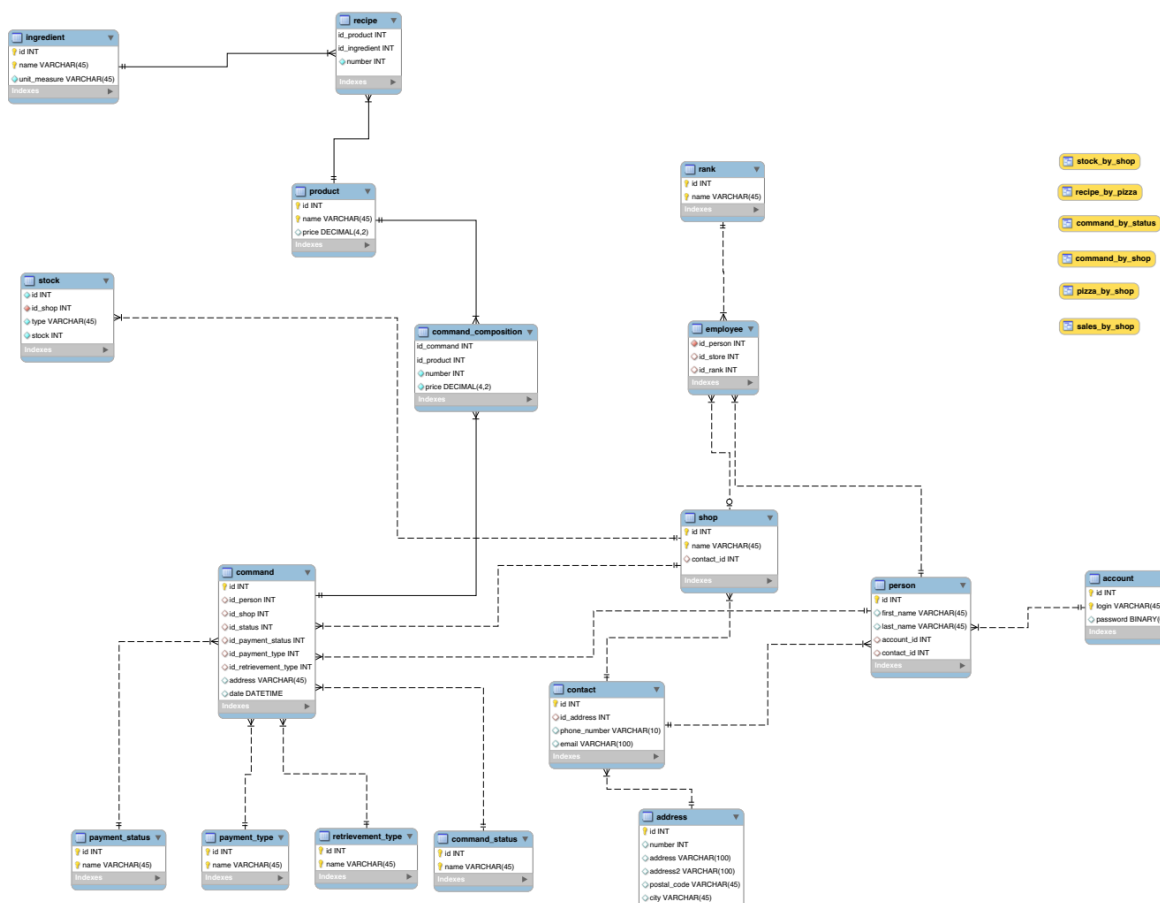
- **RAM** : 8Go
- **Stockage** : 100Go
- **Processeur** : 2,7 Ghz 8 coeurs

3.2 - Bases de données

Les bases de données et schémas suivants doivent être accessibles et à jour :

- ocpizza : version 1.0

Pour rappel :



3.3 - Web-services

Les web services suivants doivent être accessibles et à jour :

- **Payments API**: API de paiement Paypal - version 2.0
- **Maps Javascript API** : API de localisation Google Maps - version 3.43
- **sentry-sdk** : API d'envoi de logs sur Sentry - version 0.19.1

4 - PROCÉDURE DE DÉPLOIEMENT

4.1 - Déploiement du serveur de base de données

4.1.1 - Environnement

Afin de pouvoir remonter le schéma de la base de données fournie ainsi que les presets, vous devez :

- Ouvrez les ports 3306 et 33060 TCP entrant
- Installer mysql : <https://dev.mysql.com/doc/refman/8.0/en/installing.html>
- Vous connectez à mysql en root
 - mysql -u root -p (et renseigner le mot de passe fourni à l'installation)
 - Créer une nouvelle base de données nommée « ocpizza »
 - CREATE DATABASE `ocpizza`;
 - Créer un utilisateur avec votre **propre mot de passe** et lui donner les droits sur la base ocpizza
 - CREATE USER 'ocpizza-user'@'localhost' IDENTIFIED BY 'your-user-password';
 - GRANT ALL PRIVILEGES ON `ocpizza`.* TO 'ocpizza-user'@'localhost';
 - FLUSH PRIVILEGES;

LE MOT DE PASSE QUE VOUS DEFINISSEZ ICI VOUS APPARTIENT, NOTEZ LE ET GARDEZ LE PRECIEUSEMENT

4.1.2 - Artefacts

Les modèles de la base de données sont rassemblés dans le fichier **BDD.zip**, il contient deux fichiers sql :

- **ocpizza_dump.sql** : fichier contenant le modèle de la base de données.
- **base_data.sql** : fichier contenant les presets de la base de données.

Afin de remonter ces deux fichiers :

- Extraire l'archive **BDD.zip** dans le répertoire :

/etc/ocpizza/

- Positionner les droits d'exécution et de lecture sur les deux fichiers.
- Remonter le modèle de la base de données, vous aurez à renseigner le **mot de passe** de l'utilisateur que vous avez défini plus tôt :

- `mysql -u ocpizza-user -p ocpizza < /etc/ocpizza/ocpizza_dump.sql`
- Remonter les presets :
 - `mysql -u ocpizza-user -p ocpizza < /etc/ocpizza/base_data.sql`

4.1.3 - Vérifications

Afin de vérifier le bon fonctionnement de la remontée des données :

- Vérifiez la présence de la base de données dans mysql :
 - Connectez vous à mysql (`mysql -u root -p`)
 - `show databases ;`
 - **La base ocpizza doit apparaître.**
- Vérifiez la création de l'utilisateur
 - `mysql -u ocpizza-user -p`
 - Indiquez votre mot de passe
 - **Vous devez être connecté à mysql**
- Vérifiez la remontée de la base et des données
 - `mysql -u ocpizza-user -p`
 - Indiquez votre mot de passe
 - `USE ocpizza ;`
 - `show tables ;`
 - **Des tables doivent apparaître.**
 - `SELECT * FROM ocpizza.rank ;`
 - **Des données doivent apparaître.**

4.2 - Déploiement du Serveur d'Application

4.2.1 - Artefacts

L'application est disponible dans le fichier **ocpizza.zip**, il contient l'ensemble des fichiers nécessaires au bon fonctionnement de l'application.

Afin de préparer l'application :

- Installer python [Aide installation](#)
- Ouvrez les ports 3306 et 33060 TCP sortants dans votre firewall
- Extraire l'archive **ocpizza.zip** dans le répertoire :

/etc/ocpizza/

- Positionner les droits d'exécution et de lecture sur l'ensemble des fichiers

4.2.2 - Environnement de l'application web

Remplacez dans le fichier **/etc/ocpizza/settings/production.py** les champs suivants en remplaçant les différents champs par vos noms de domaines et l'adresse de ce serveur.

- `ALLOWED_HOST = ['ADRESSE_IP', 'mon-domaine.com', 'www.mon-domaine.com']`

Vous avez possibilité d'en rajouter de cette façon :

- `ALLOWED_HOST = ['ADRESSE_IP', 'mon-domaine.com', 'www.mon-domaine.com', 'nouveau-domaine.com']`

Afin de mettre en place l'environnement de l'application web, lancer le script suivant en indiquant le **mot de passe** de l'utilisateur que vous avez créé sur mysql, **l'adresse IP** du serveur SQL et **l'utilisateur** qui sera chargé de lancer l'application en production :

- `cd /etc/ocpizza/`
- `chmod +x deploy-apply.sh change-variable.sh change-user.sh`
- `sudo ./deploy-apply.sh user-launching-app-password your-sql-ip user-launching-app`

4.2.3 - Fichiers de configurations

4.2.3.1 - *ocpizza-gunicorn.conf*

Ce fichier est situé dans le répertoire **/etc/supervisor/conf.d**.

Il regroupe l'ensemble des informations nécessaires au lancement de l'application et à son maintien par supervisor.

Vous y trouverez :

- `command` : commande utilisée pour lancer l'application
- `user` : l'utilisateur lançant l'application
- `directory` : le répertoire contenant l'application
- `autostart` : l'application se démarre au lancement du serveur
- `autorestart` : en cas de crash, l'application se relance d'elle même
- `environment` : les différentes variables d'environnement nécessaires au fonctionnement de l'application
- `stderr_events_enabled` : activation des évènements d'erreurs
- `redirect_stderr` : redirection des erreurs
- `stdout_logfile` : fichier de log
- `stderr_logfile` : fichier de log d'erreurs

4.2.3.2 - *production.py*

Ce fichier se trouve dans le répertoire **/etc/ocpizza/settings**.

Il regroupe les paramètres django de votre application, les paramètres renseignés ici sont nécessaires au bon fonctionnement de votre application. La seule option paramétrable est le champ « `ALLOWED_HOST` » comme décrit plus haut.

4.2.4 - *Changements de variables*

Les variables d'environnements sont modifiables, pour cela effectuez les commandes suivantes :

- Positionnez vous dans le répertoire des scripts de l'application
 - `cd /etc/ocpizza/scripts`
- Lancez la commande avec les modifications nécessaires
 - `sudo ./change-variable.sh variable-name variable-value`
- Vous aurez besoin de redémarrer supervisor, ce qui impliquera **une inaccessibilité de votre site pendant quelques instants**
 - `sudo supervisorctl update`

Les variables pouvant être modifiées sont grâce à ce fichier sont :

- `HOST` : Adresse IP du serveur SQL
- `SQL_PASSWORD` : mot de passe de la base SQL

La liste ci-dessus peut être soumise à modification.

Pour modifier l'utilisateur lançant l'application :

- Positionnez vous dans le répertoire des scripts de l'application
 - `cd /etc/ocpizza/scripts`
- Lancez la commande avec les modifications nécessaires
 - `sudo ./change-user.sh new-user`
- Vous aurez besoin de redémarrer supervisor, ce qui impliquera **une inaccessibilité de votre site pendant quelques instants**
 - `sudo supervisorctl update`

4.2.5 - Vérifications

Afin de vérifier le bon déploiement de l'application, vous pouvez accéder à l'application en local en tapant dans un navigateur

- <http://127.0.0.1:8000> ou http://application_server_ip:8000

Les images n'apparaîtront pas, il s'agit d'un comportement normal, celles-ci seront hébergées par le serveur web.

Vous pouvez vérifier le bon fonctionnement de supervisor avec la commande suivante :

- `sudo supervisorctl status`

4.3 - Déploiement du serveur Web

4.3.1 - Artefacts

Les batches de l'application sont construits sous la forme d'une archive ZIP contenant un script :

- **deploy-web.sh** : Fichier bash pour la mise en place des fichiers de configuration du serveur web

Afin de préparer le serveur Web :

- Ouvrez le port 443 TCP entrant dans votre firewall
- Extraire l'archive **web.zip** dans le répertoire :

/etc/ocpizza

- cd /etc/ocpizza
- Positionner les droits d'exécution sur le script SH.
- Lancer le script de configuration en spécifiant les **noms de domaine** de l'application (**limité à deux** pour le moment) et l'**adresse IP** du serveur d'application :
 - ./deploy-web.sh **domain-name-1 domain-name-2 application-server-ip**

4.3.2 - Configuration

Voici le fichier de configurant de nginx se trouve à l'emplacement suivant:

- /etc/nginx/sites-availables/**domain-name-1**

Vous pouvez rajouter des noms de domaines pour votre site directement dans ce fichier de configuration.

Pour cela, rajoutez-les à la suite sur les lignes commençant par « server name » avant le point-virgule.

4.3.3 - Vérifications

Vous pouvez vérifier le bon fonctionnement du serveur Web en lançant les commandes suivantes :

- Vérifier le fichier de configuration nginx :

- `sudo nginx -t`
- Vérifier le statut du service nginx
 - `sudo service nginx status`

5 - PROCÉDURE DE DÉMARRAGE / ARRÊT

Dans le cadre d'une extinction de tous les serveurs, respectez l'ordre suivant :

1. Serveur web
2. Serveur d'application
3. Serveur de Base de données

5.1 - Serveur web

5.1.1 - Démarrage

- 1) Démarrer le serveur
- 2) Vérifiez le statut du service nginx avec la commande
sudo service nginx status
- 3) S'il ne fonctionne pas
sudo service nginx start
ou
sudo service nginx restart
en fonction du retour du statut

5.1.2 - Arrêt

- 1) Arrêter le service nginx
sudo service nginx stop
- 2) Arrêter le serveur

5.2 - Serveur d'application

5.2.1 - Démarrage

- 1) Démarrer le serveur
- 2) Vérifiez le statut du service supervisor avec la commande
sudo supervisorctl status
- 3) S'il ne fonctionne pas
sudo supervisorctl start
ou
sudo supervisorctl restart

en fonction du statut initial

5.2.2 - Arrêt

- 1) Arrêter supervisor
sudo supervisorctl stop
- 2) Arrêter le serveur

5.3 - Serveur de Base de données

5.3.1 - Démarrage

- 1) Démarrer le serveur
- 2) Vérifiez le statut du service mysql avec la commande
sudo service mysql status
- 3) S'il ne fonctionne pas
sudo service mysql start
ou
sudo service mysql restart
en fonction du statut initial

5.3.2 - Arrêt

- 1) Arrêter le service mysql
sudo service mysql stop
- 2) Arrêter le serveur

6 - PROCÉDURE DE MISE À JOUR

Dans le cadre d'une mise à jour un **fichier zip** vous sera fourni.

IL EST RECOMMANDE DE METTRE L'APPLICATION HORS PRODUCTION ET D'EFFECTUER UNE SAUVEGARDE DE LA BASE DE DONNEES AVANT TOUTE MISE À JOUR.

Vous devrez agir sur le serveur d'application :

- Arrêter supervisor
 - `sudo supervisorctl stop domain-name-1`
- Décompresser l'archive dans **/etc/ocpizza** en écrasant les fichiers.
- `cd /etc/ocpizza`
- `chmod +x update_number.sh`
- `./update_number.sh`
- Un message vous invitera à relancer supervisor
 - `sudo supervisorctl start domain-name-1`

7 - SUPERVISION/MONITORING

7.1 - Supervision de l'application web

Afin de tester que l'application web est toujours fonctionnelle, vous pouvez tenter d'accéder au nom de domaine lié à votre application.

Vous pouvez également vérifier le fonctionnement des différents services :

- Serveur de base de données
 - `sudo service mysql status`
- Serveur d'application
 - `sudo service supervisor status`
 - `sudo supervisorctl status`
- Serveur Web
 - `sudo service nginx status`

7.2 - Monitoring

Le monitoring de l'application se réalise via Sentry, vous y avez accès en suivant l'URL suivante :

<https://sentry.io/>

Les accès vous ont été communiqués par mail, sur l'adresse de contact que vous nous avez fourni.

De plus, vous avez accès au monitoring via la plateforme à laquelle vous avez souscrite, je vous invite à vous rapprocher de leur service client pour plus d'informations.

8 - PROCÉDURE DE SAUVEGARDE ET RESTAURATION

8.1 - Mise en place de la stratégie de sauvegarde

Afin de mettre en place la stratégie de sauvegarde de la base de données :

- Placez-vous dans le répertoire **/etc/ocpizza** du serveur de base de donnée
 - `sudo cd /etc/ocpizza`
- Ajoutez les droits d'exécution sur les scripts de sauvegarde
 - `sudo chmod +x job-sql.sh backup.sh`
- Mettre tous les droits sur le répertoire de l'emplacement de sauvegarde de votre choix (**directory**) et sur le fichier de log **/etc/ocpizza/logs/backup.log**
 - `sudo chmod 777 directory /etc/ocpizza/logs/backup.log`
- Lancez le script de configuration en spécifiant **dans cet ordre** le nom de l'utilisateur mysql root (**user**), son mot de passe (**your-user-password**), et l'emplacement des sauvegarde (**directory**).
 - `./job-sql.sh user your-root-password directory`

AFIN DE PREVENIR TOUTE PERTE DE DONNEES, APPLIQUEZ LE PLAN SUIVANT DE RETENTION :

- Gardez l'ensemble des sauvegardes pendant au minimum une semaine
- Une fois la semaine passée, gardez une sauvegarde par jour pendant trois mois.
- Une fois les trois mois passés, gardez une sauvegarde par semaine pendant un an.

PENSER A REGULIEREMENT EXTERNALISER UNE PARTIE DE VOS SAUVEGARDES SUR UN DISQUE EXTERNE OU UN SERVEUR LOCALISE AUTRE PART

8.2 - Restauration de sauvegardes

AVANT DE REMONTER UNE SAUVEGARDE, VEILLEZ A METTRE VOTRE SITE HORS PRODUCTION EN ARRÊTANT SUPERVISOR SUR LE SERVEUR D'APPLICATION

```
sudo supervisorctl stop domain-name-1
```

8.2.1 - Remontée de la base de production

Afin de remonter la base de production, utilisez la commande suivante :

```
sudo mysqldump -u user -pyour-root-password ocpizza > backup-name.sql
```

8.2.2 - Remontée de toutes les bases

Afin de remonter toutes les bases de données (ainsi que les système) :

```
mysqldump --all-databases --single-transaction --quick --lock-tables=false > backup-  
name.sql -u user -pyour-root-password
```

9 - GLOSSAIRE

application-server-ip	Adresse IP du serveur d'application
backup-name	Nom de la sauvegarde à restaurer
directory	Répertoire de sauvegarde
domain-name-1	Nom de domaine principal
domain-name-2	Nom de domaine secondaire
new-user	Nouvel utilisateur lançant l'application
update_number	Nom du script de mise à jour avec un numéro incrémenté
user	Nom de l'utilisateur root sur mysql
user-launching-app	Nom de l'utilisateur lançant l'application
user-launching-app-password	Mot de passe de l'utilisateur lançant l'application
variable-name	Nom de la variable à changer
variable-value	Valeur de la variable à changer
your-root-password	Mot de passe de votre utilisateur root sur mysql
your-sql-ip	Adresse IP du serveur SQL
your-user-password	Mot de passe de votre utilisateur SQL