

Projet 9 – Parcours Développeur d'application – Python (V1)

Documentez votre système de gestion de pizzeria
OC PIZZA

OPENCLASSROOMS

Dossier d'exploitation

Auteur

Lamia EL RALIMI

Analyste programmeuse

TABLE DES MATIERES

1 - Versions	3
2 - Introduction	4
2.1 - Objet du document	4
2.2 - Références.....	4
3 - Pré-requis	5
3.1 - Système	5
3.1.1 - <i>Serveur de Base de données</i>	6
3.1.1.1 - Caractéristiques techniques	6
3.1.2 - <i>Serveur Web</i>	6
3.1.2.1 - Caractéristiques techniques	6
3.1.3 - <i>Serveur d'application</i>	7
3.1.3.1 - Caractéristiques techniques	7
3.2 - Web-service	7
4 - Procédure de déploiement.....	8
4.1 - Environnement de l'application	8
4.1.1 - <i>Configuration de l'application</i>	9
4.1.2 - <i>Initialisation de la base de données</i>	10
5 - Procédure de démarrage / arrêt.....	11
5.1 - Serveur d'application	11
5.2 - Serveur Web	11
5.3 - Base de données	12
6 - Procédure de mise à jour.....	13
6.1 - Serveur d'application	13
6.2 - Base de données	13
7 - Supervision/Monitoring.....	14
7.1 - Supervision du serveur	14
7.2 - Supervision du serveur	14
8 - Procédure de sauvegarde et restauration	16
9 - Glossaire	17

1 - VERSIONS

Auteur	Date	Description	Version
Lamia EL RALIMI	02/05/2021	Création du document	1.0

2 - INTRODUCTION

2.1 - Objet du document

Le présent document constitue le dossier d'exploitation de l'application pour le système de gestion de l'ensemble des pizzerias du groupe OC Pizza.

L'objectif de ce document est de définir les différentes caractéristiques du système nécessaires à l'exécution de l'application ainsi qu'à sa maintenance (mise à jour, supervision, sauvegarde et restauration).

Par ailleurs, ce document permet de détailler les différentes étapes de déploiement de l'application, la procédure de mise à jour, de sauvegarde et de restauration.

2.2 - Références

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

- Dossier de conception fonctionnelle ;
- Dossier de conception technique ;
- PV de livraison ;

3 - PRÉ-REQUIS

3.1 - Système

L'ensemble de l'application web sera hébergé sur un serveur de type IAAS (Infrastructure As A Service) fournit par [Digital Ocean](#) ayant les caractéristiques techniques suivantes :

Caractéristiques	Option souscrite
Système d'exploitation	Linux / Ubuntu version 20.04
Mémoire	8 GB
Disque SSD	25 GB
Processeur(s)	2

Sur ce serveur, un compte utilisateur **admin_ocp** sera créé. Il sera utilisé pour la mise en place et la maintenance de l'application.

Pour assurer le déploiement de l'application sur ce serveur, les paquets suivants devront être installés :

Paquets	Version minimale
Python	3.8
MySQL	8.0
Pipenv	2020.11.15
Supervisord	4.2.0
Cerbot	1.0.0

3.1.1 - Serveur de Base de données

Le serveur de base de données sera également hébergé via un plan souscrit chez Digital Ocean et le système de gestion de base de données relationnelle (SGBDR) sera MySQL.

Ce serveur sera configuré de la manière suivante :

- Une base de données nommée **ocpizza_db**.
- L'utilisateur **admin_ocp** aura accès à tous les privilèges sur cette base de données.

Selon les besoins futurs, il sera possible de faire évoluer ce plan.

3.1.1.1 - Caractéristiques techniques

Le serveur de base de données présentera les caractéristiques suivantes :

SGBD	MySQL
Version minimale	8.0

3.1.2 - Serveur Web

Le serveur Web sera un serveur **Nginx**.

Le pare-feu sera configuré pour autoriser Nginx en connexions entrantes et sortantes. **Cerbot** sera utilisé afin de générer des certificats **SSL** conformes et assurer la bonne configuration du serveur **NGINX**.

3.1.2.1 - Caractéristiques techniques

Le serveur Web présentera les caractéristiques suivantes :

Type de serveur Web	Nginx
Version minimale	1.0

3.1.3 - Serveur d'application

Le serveur d'application utilisé sera **Gunicorn**. Il sera surveillé par l'outil **Supervisor** afin d'assurer le démarrage et le redémarrage automatique de l'application.

Un fichier de configuration de l'outil Supervisor sera créé dans le répertoire suivant :
`/etc/supervisor/conf.d/ocpizza-gunicorn.conf`.

3.1.3.1 - Caractéristiques techniques

Le serveur d'application aura les caractéristiques suivantes :

Type de serveur d'application	Gunicorn
Version minimale	20.0.4

3.2 - Web-service

Le Web service de [Stripe](#) sera utilisé pour les transactions financières.

4 - PROCÉDURE DE DÉPLOIEMENT

4.1 - Environnement de l'application

Répertoire de projet : **ocpizza**

Utilisateur : **admin_ocp**

Se connecter au serveur en SSH :

```
$ ssh admin_ocp@123.111.222.333
```

Se placer dans le répertoire du projet :

```
$ cd /home/ocpizza
```

Configurer git :

```
$ git config --global user.name "username"
```

```
$ git config --global user.email "email@email.com"
```

```
$ git remote add origin https://github.com/it-consulting/ocpizza.git
```

Cloner le projet :

```
$ git clone git@github.com:it-consulting/ocpizza.git .
```

Mettre en place l'environnement virtuel (*s'active automatiquement la première fois, pour l'activer dans le futur, la commande est la même*) :

```
$ pipenv shell
```

Installer les dépendances du projet :

```
$ pipenv install
```

Collecter les fichiers statiques :

```
$ python3 manage.py collectstatic
```


4.1.1 - Configuration de l'application

Les variables d'environnement de l'application sont définies dans un fichier `.env` à la racine du répertoire de projet.

Le fichier `.env` sera créé directement sur le serveur au moment du déploiement de l'application.

Voici la liste des variables d'environnement :

Nom	Obligatoire
DB_NAME	Oui
DB_USER	Oui
DB_PASSWORD	Oui
HOST	Oui
PORT	Oui
ENV	Oui
SECRET_KEY	Oui

De plus, la configuration des paramètres de l'application se trouve dans le dossier settings.

Voici les différents fichiers de configuration :

Chemin	Description
ocpizza/settings.py	Fichier de configuration de l'application.
ocpizza_log	Fichier de configuration des logs.
/etc/supervisor/conf.d/ocpizza-gunicorn	Fichier de configuration de Supervisor.

4.1.2 - Initialisation de la base de données

Vérifier les points suivants :

- Les paramètres de la base de donnée sont correctement renseignés dans le fichier `.env`.
- L'environnement virtuel est activé

Créer les tables de la base de données :

```
$ python3 manage.py migrate
```

La base de données sera pré-remplies à l'aide des informations transmises auparavant par OC Pizza sous forme de fichiers CSV. Ces fichiers CSV seront directement téléchargés sur le serveur dans le répertoire `/home/admin_ocp/db_seed.csv`.

La commande pour lancer l'initialisation :

```
$ python3 manage.py feed_db -d /home/admin_ocp/db_seed.csv
```

5 - PROCÉDURE DE DÉMARRAGE / ARRÊT

5.1 - Serveur d'application

Le démarrage et redémarrage du serveur d'application est géré automatiquement par **Supervisor**. Cependant, si cela s'avère nécessaire l'utilisateur peut exécuter les commandes suivantes à l'aide de Supervisor.

Démarrer Gunicorn :

```
$ sudo supervisorctl start ocpizza-gunicorn
```

Vérifier le statut du processus :

```
$ sudo supervisorctl status ocpizza-gunicorn
```

Arrêter Gunicorn:

```
$ sudo supervisorctl stop ocpizza-gunicorn
```

À la suite d'une modification du fichier de configuration, exécuter les commandes suivantes :

```
$ sudo supervisorctl reread
```

```
$ sudo supervisorctl update
```

5.2 - Serveur Web

Le cas échéant, voici comment agir sur le serveur Nginx.

Démarrer le serveur web Nginx :

```
$ sudo service nginx start
```

Vérifier le statut du processus :

```
$ sudo service nginx status
```

Arrêter Nginx :

```
$ sudo service nginx stop
```

À la suite d'une modification du fichier de configuration, exécuter la commande suivante :

```
$ sudo service nginx reload
```

5.3 - Base de données

Démarrer la base de données MySQL :

```
$ sudo systemctl start mysql
```

Vérifier le statut du processus de la base de données :

```
$ sudo systemctl status mysql
```

Arrêter MySQL :

```
$ sudo systemctl stop mysql
```

À la suite d'une modification de la configuration de la base de données, exécuter la commande suivante :

```
$ sudo systemctl reload mysql
```

6 - PROCEDURE DE MISE A JOUR

Avant de lancer ces commandes, vérifier les points suivants :

- L'environnement virtuel est activé
- L'application n'est pas lancée

6.1 - Serveur d'application

Mise à jour des fichiers source de l'application depuis le dépôt GitHub :

```
$ git pull origin master
```

Mise à jour des fichiers statiques :

```
$ python3 manage.py collectstatic
```

Mise à jour des paquets du serveur d'application :

```
$ sudo apt-get update # voir les paquets pouvant être mis à jour
```

```
$ sudo apt-get upgrade # lancer la mise à jour de ces paquets
```

ATTENTION : La mise à jour des paquets peut altérer le bon fonctionnement de l'application.

6.2 - Base de données

Mise à jour de la base de données :

```
$ python3 manage.py migrate
```

7 - SUPERVISION/MONITORING

7.1 - Supervision du serveur

Les outils de supervision du serveur se font depuis l'espace client de Digital Ocean dans la partie **Monitoring** du dashboard :



Nous avons déjà paramétré des alertes pour les éléments suivants :

Eléments surveillés	Description
CPU (processeur)	Si l'intensité de son traitement dépasse les 70%.
Utilisation du disque	Si son utilisation dépasse 80%.
Utilisation de la mémoire	Si son utilisation dépasse 70%.

Ces alertes pourront être modifiées et d'autres alertes peuvent être ajoutées.

Les métriques en temps réel peuvent être consultées dans la partie **Graphs** des détails du droplet Digital Ocean ainsi que sur le serveur avec la commande suivante :

```
$ htop
```

7.2 - Supervision du serveur

La surveillance de l'application est assurée par [Sentry](#).

Sentry permet de trier et afficher les logs remontées par l'application.

Il fournit également des détails concernant le navigateur de l'utilisateur, l'URL et la requête concernée.



IT Consulting & Development

Des alertes ont été créées afin de signaler toute nouvelle erreur sur l'adresse e-mail suivante :

admin_ocpizza@ocpizza.com

8 - PROCEDURE DE SAUVEGARDE ET RESTAURATION

Une sauvegarde de la base de données sur le serveur dédié a été planifiée chaque jour, à 02h00 du matin heure de Paris.

Il est possible de lancer cette sauvegarde manuellement.

Lancer la sauvegarde :

```
$ mysqldump --databases ocpizza_db > dump.sql
```

Réaliser une restauration de la sauvegarde :

```
$ mysql < dump.sql
```


9 - GLOSSAIRE

Cerbot	Logiciel de certification par l'autorité de certification Let's Encrypt.
Digital Ocean	Fournisseur d'infrastructure cloud.
Django	Framework écrit dans le langage Python.
Gunicorn	Logiciel de serveur d'application Python.
MySQL	Système de gestion de base de données relationnelle (SGBDR).
Nginx	Logiciel de serveur web/HTTP va transférer les requêtes effectuées sur l'IP publique vers l'adresse IP privée.
Python	Langage de programmation.
Sentry	Une interface web pour trier et afficher les logs remontés par app. Fournit également des détails concernant le navigateur de l'utilisateur, l'URL, la requête... <i>(Possibilité de le relier à GitHub pour suivre l'avancée des issues.)</i>
Supervisor	Logiciel de gestion des processus système qui permet de contrôler un ensemble de processus vivant dans un environnement UNIX. Il lance des processus et les redémarrent s'ils échouent.