

OC PIZZA

Application web et gestion OC PIZZA

Dossier d'exploitation

Version 1.0

Auteur

Tony Gonzalez
Analyste programmeur

TABLE DES MATIERES

1 -Versions.....	3
2 -Introduction	4
2.1 -Objet du document	4
2.2 -Références	4
3 -Prérequis.....	5
3.1 -Serveur.....	5
3.2 -Bases de données.....	5
3.3 -Webservices.....	5
4 -Procédure de déploiement	6
4.1 -Configuration du serveur.....	6
4.1.1 -Pare-feu.....	6
4.1.2 -Connexion au serveur.....	9
4.2 -Déploiement de l'application web	7
4.2.1 -Environnement de l'application.....	7
4.2.2 -Variables d'environnement.....	7
4.2.3 -Déploiement	8
4.3 -Déploiement de la base de données	8
4.3.1 -Configuration.....	8
4.3.2 -Migrations.....	9
4.3.3 -Création d'un super utilisateur.....	9
4.3.4 -Import des données.....	9
5 -Procédure de démarrage / arrêt	10
5.1 -Fonctionnement.....	10
5.2 -Démarrage de l'application	11
5.3 -Arrêt de l'application.....	11
5.4 -Vérifications	11
6 -Procédure de mise à jour	12
6.1 -Base de données.....	12
6.2 -Serveur.....	12
7 -Supervision/Monitoring	13
7.1 -Supervision de l'application.....	13
7.2 -Supervision du serveur.....	13
8 -Procédure de sauvegarde et restauration	14
9 -Glossaire	15

1 - VERSIONS

Auteur	Date	Description	Version
Tony Gonzalez	01/10/2020	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.

L'objectif de ce document est de fournir, à l'équipe technique d'OC Pizza, les informations relatives au déploiement de l'application sur un serveur (production), ainsi qu'à sa maintenance.

2.2 - Références

Pour de plus amples informations, se référer :

1. **DCT – Version 1.1** : Dossier de conception technique de l'application.
2. **DCF – Version 1.1** : Dossier de conception fonctionnelle de l'application.

3 - PREREQUIS

3.1 - Serveur

L'application web OC Pizza est hébergée sur un serveur de Digital Ocean.

Le nom de domaine est de l'application : OCPizza.fr

L'adresse complète étant : <https://OCPizza.fr>

3.2 - Bases de données

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

- **MySQL** : version 8.0

3.3 - Webservices

Les web services suivants doivent être accessibles et à jour :

- **PayPal** : pour le paiement en ligne.
- **Sentry** : pour le monitoring des logs de l'application.
- **GitHub** : service de versioning et de sauvegarde du projet.
- **Travis CI** : service utilisé pour tester l'application.
- **Digital Ocean** : serveur web.

4 - PROCEDURE DE DEPLOIEMENT

4.1 - Configuration du serveur

Le Droplet créé se nomme OCPizza.

Le système d'exploitation du serveur est Ubuntu version 20.04 et son Data center est situé à Amsterdam. Ce dernier, disponible avec Digital Ocean, étant le plus proche de France. Son adresse IP est : 161.34.152.135

La documentation de Digital Ocean se trouve à cette adresse :

<https://www.digitalocean.com/docs/>

4.1.1 – Pare-feu

Le pare-feu doit être activé et autorise les connexions SSH (port 22), les requêtes http (port 80) et https (port 443).

4.1.2 – Connexion au serveur

Pour se connecter au serveur en ligne de commande (connexion SSH) :

Ssh ocpizza@161.34.152.135

Le mot de passe est Oc-PizZa-2020.

4.2 - Déploiement de l'Application Web

4.2.1 - Environnement de l'application

L'application est composée des dossiers et fichiers suivants :

- oc_pizza : contient les fichiers de configuration de l'application.
- static : contient les fichiers relatifs à la partie front-end (CCS, JavaScript, images).
- template : contient les fichiers HTML.
- authentication : dossier applicatif relatif à la création de comptes et connexion de tous les utilisateurs.
- personnel : dossier applicatif pour les opérations relatives aux employés.
- commandes : dossier applicatif du système de commandes de pizzas.
- Procfile : fichier de configuration utilisé par Gunicorn pour démarrer l'application.
- requirements.txt : fichier indiquant les librairies nécessaires au bon fonctionnement de l'application.
- README.md : documentation concernant le fonctionnement de l'application.

4.2.2 - Variables d'environnement

L'application en production doit être exécutée avec les variables d'environnements suivantes définies au démarrage. La configuration de l'application se trouve dans le dossier settings lui-même à l'intérieur du dossier oc_pizza.

Les fichiers de configurations sont les suivants :

- __init__.py : configuration de développement
- production.py : configuration de production

A noter que le fichier production.py reprend en grande partie la configuration du fichier __init__.py et seules quelques modifications sont effectuées.

Nom	Valeur	Description
ENV	PRODUCTION	Indique que l'environnement utilisé est celui de production
SECRET_KEY	4w7(#mdf++#1tmn44sq7n*bpw78+1cyq2stm81mxzodj=f-%p0	Cette clé est nécessaire au démarrage de l'application Django et doit absolument rester confidentielle.

INFO : il ne faut pas mettre de « / » à la fin de la valeur de la variable et ne pas utiliser d'espace dans le chemin.

4.2.3 - Déploiement

Avant de commencer le déploiement, nous installerons python 3 et les librairies nécessaires au bon fonctionnement de l'application :

```
pip install python3 python3-pip mysql-connector-python
```

Le déploiement de l'application se fait en ligne de commande via GitHub de la manière suivante :

```
git clone https://github.com/OCPizza/oc_pizza.git
```

L'installation des dépendances du fichier requirements.txt se fait à l'aide de la commande suivante :

```
pip install -r oc_pizza/requirements.txt
```

4.3 - Déploiement de la base de données

4.3.1 - Configuration

Le système de gestion de base de données (SGBD) utilisé es MySQL version 8.0

Configuration de la base de données

La configuration de la base de données se trouve dans le fichier de configuration production.py et peut être modifiée à tout moment.

Sa configuration est la suivante :

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.mysql',  
        'NAME': 'ocpizza',  
        'USER': 'ocpizza',  
        'PASSWORD': ' Oc-PizZa-2020',  
        'HOST': '',  
        'PORT': '5432',  
    }  
}
```


4.3.2 - Migrations

Toutes modifications de la structure de la base de données entraînent la nécessité d'effectuer par la suite des migrations.

Les migrations se font à l'aide de la commande suivante.

```
python3 manage.py makemigrations
```

Il faut ensuite entériner les changements avec la commande :

```
python3 manage.py migrate
```

4.3.3 - Création d'un super utilisateur

Un super utilisateur est déjà créé. Les informations sont les suivantes :

- username : oc_pizza
- password : Oc-PizZa-2020
- e-mail : oc_pizza@gmail.com

Si vous avez besoin de créer un nouveau super utilisateur, procédez comme suit :

```
python3 manage.py createsuperuser
```

Par la suite, Django vous demandera de choisir un nom d'utilisateur et un mot de passe. Il faudra aussi indiquer une adresse mail.

Le super utilisateur ainsi que tous les utilisateurs de l'application se trouve dans la table MySQL « user ».

4.3.4 - Import des données

Il est possible, en cas de besoin, d'importer les données existantes de l'environnement de développement vers celui de production.

Pour cela, il suffit de réaliser un instantané (dump) de la base de données de la manière suivante :

```
python3 manage.py dumpdata oc_pizza > oc_pizza/dumps/oc_pizza.json
```

Pour importer l'instantané, utilisez la commande suivante :

```
python3 manage.py loaddata oc_pizza/dumps/oc_pizza.json
```

5 - PROCEDURE DE DEMARRAGE / ARRET

5.1 - Fonctionnement

Pour fonctionner, le serveur utilisé est Nginx.

Pour l'installer nous tapons la commande suivante :

```
sudo apt-get install nginx
```

La configuration de Nginx est située dans `etc/nginx/sites-available/oc_pizza`

Si vous changez la configuration de Nginx, il faudra ensuite entériner les changements de la manière suivante : `sudo service nginx reload`

5.2 - Démarrage de l'application

Pour démarrer l'application en développement, il faut utiliser la commande :

```
python3 manage.py oc_pizza.
```

En production, l'application est lancée par Supervisor.

Supervisor est un système qui permet de lancer des services et les redémarre s'ils échouent.

Son installation se fait avec cette commande : `sudo apt-get install supervisor`

Vous pourrez retrouver sa configuration en tapant cette commande :

```
sudo vi /etc/supervisor/conf.d/oc_pizza-gunicorn.conf
```

Dans notre système, notre application est lancée par Unicorn, qui est lui-même lancé par Supervisor. En d'autres termes, si l'application s'arrête, Supervisor la redémarre automatiquement.

Dans le cas où vous auriez besoin de lancer Supervisor, à la suite de modifications de la configuration du serveur ou de l'application, il vous faudra utiliser les commandes suivantes :

```
sudo supervisorctl reread
```

puis :

```
sudo supervisorctl update
```

Il est aussi possible d'utiliser la commande suivante si aucun changement de configuration n'a été effectué :

```
sudo supervisorctl restart all
```

5.3 - Arrêt de l'application

En cas de maintenance du site ou autre, pour arrêter Supervisor la commande à utiliser est la suivante :

```
sudo supervisorctl stop all
```

5.4 - Vérifications

Pour vérifier le bon fonctionnement de Supervisor, vous pouvez voir son status :

```
sudo supervisorctl status
```

Vous devriez voir s'afficher dans la console un message comme celui-ci :

```
oc_pizza-gunicorn      RUNNING   pid 16309, uptime 0:00:09
```

Enfin, vous pouvez aussi vous rendre sur le site à l'adresse suivante :

```
https://OCPizza.fr
```

6 - PROCEDURE DE MISE A JOUR

6.1 - Base de données

Digital Ocean possède un service de mise à jour automatique. Les manipulations à effectuer sont expliquées dans la documentation de leur site :

<https://www.digitalocean.com/docs/databases/mysql/how-to/schedule-updates/>

6.2 - Serveur

Pour la mise à jour des programmes du serveur, tapez en ligne de commande :

```
sudo apt-get update
```

7 - SUPERVISION/MONITORING

7.1 - Supervision de l'application

Pour superviser l'application, nous utiliserons Sentry.

Sentry est un tableau de bord qui vous permet de visualiser ce qui se passe dans l'application Django.

La commande suivante servira à installer Sentry sur le serveur :

```
pip install --upgrade sentry-sdk
```

Vous pourrez retrouver toutes informations concernant l'application sur le site de sentry :

<https://sentry.io>

Nous indiquons aussi les informations relatives à la documentation :

<https://docs.sentry.io/platforms/python/guides/django/>

7.2 - Supervision du serveur

Pour surveiller l'activité du serveur, commencez par installer l'agent de Digital Ocean sur votre serveur avec la commande suivante :

```
curl -sSL https://agent.digitalocean.com/install.sh | sh
```

Puis allez dans l'interface d'administration de Digital Ocean. Dans votre droplet vous aurez accès aux informations suivantes :

CPU : composant qui exécute les instructions des programmes. Ici vous pouvez visualiser le pourcentage de puissance utilisé.

Memory : représente la mémoire vive utilisée.

Disk I/O (disk input/output) : quantité de données lues ou écrites sur le disque dur du serveur par seconde.

Disk usage : quantité de données stockées dans le disque dur du droplet.

Bandwidth (bande passante) : représente la quantité de données transférée entre le droplet et des ressources externes.

Top processes : affiche les processus les plus exigeants en termes de mémoire vive (Memory) et de processeur (CPU).

8 - PROCEDURE DE SAUVEGARDE ET RESTAURATION

Digital Ocean permet d'effectuer des sauvegardes et restaurations de notre Droplet.

Pour cela, dirigez-vous dans la section « Backups » puis cochez la case « Enable Backups » afin de les autoriser.

La documentation de Digital Ocean sur les sauvegardes et restaurations du Droplet se trouve à cette adresse :

<https://www.digitalocean.com/docs/images/backups/how-to/create-and-restore/>

Pour restaurer manuellement une sauvegarde de la base de données, vous trouverez également les informations nécessaire dans la documentation de Digital Ocean :

<https://www.digitalocean.com/docs/databases/mysql/how-to/restore-from-backups/>

9 - GLOSSAIRE

Droplet	Espace serveur
Data center	Un data center ou centre de données est un site physique regroupant des installations informatiques (serveurs, routeurs, commutateurs, disques dur)