

Dossier d'exploitation

Auteur	Date	Description	Version
Benjamin C.	07/05/2019	Création du document	V1

Objet du document

Le présent document constitue le dossier d'exploitation de l'application oc-pizza.com.

Il a pour objectif de définir le système nécessaire à son fonctionnement et actuellement mis en place, les instructions de déploiement, les instructions de maintenance et de surveillance.

Système

Hébergement

L'application est hébergé sur Amazon Web Service sur une instance EC2 Ubuntu Server 18.04 LTS x64

IP Public : 3.16.90.115

DNS : www.oc-pizza.com

Application

Les éléments pour faire fonctionner l'application:

- Python >= 3.6
- Django >= 2.0
- Virtualenv == 15.0.0

Les packages utilisés sont identifiés dans le fichier *requirements.txt* à la racine du dépôt.

Serveur de base de données

Le SGDB utilisé est Postgres == 10.7 hébergé sur AWS

Serveur web

- Nginx >= 1.16.0
- Gunicorn == 19.9.0
- Supervisor >= 4.0.2

Procédure de déploiement

Le code est hébergé sur Github dans un dépôt privé.

Une vérification des tests est effectué grâce à Travis CI qui déploie automatiquement les mises à jour grâce à Code Deploy de AWS.

Le déploiement de l'application mis à jour est assuré par un script (détails plus bas).

Avant le premier déploiement

Ces manipulations sont réalisées avec une connexion ssh sur l'instance AWS. il faut récupérer le fichier "votre_clé_de_connection.pem" sur le tableau de bord AWS.

```
ssh -i "votre_clé_de_connection.pem" ubuntu@ec2-3-16-90-115.us-east-2.compute.amazonaws.com
```

Ces manipulations sont à titre indicatif en cas de changement d'hébergement, elles ont déjà été réalisées par nos soins.

Mise à jour du système et installation des outils

```
sudo apt-get update && sudo apt-get upgrade
sudo apt-get install python3-pip python3-dev libpq-dev postgresql
postgresql-contrib nginx supervisor virtualenv
```

Mise en place de l'environnement

```
git clone https://github.com/oc-pizza/oc-pizza.git
cd oc-pizza
virtualenv env -p python3
source env/bin/activate
chmod +x ~/oc-pizza/scripts/*
```

Liste des variables d'environnement à créer

Nom	Description	Valeur
ENV	Variable définissant l'environnement de production	'PRODUCTION'
SECRET_KEY	Variable de chiffrement nécessaire à Django	Variable aléatoire secrète sur 32 digits
NAME_DB	Nom de la base de donnée	'OCPIZZA'
USER_DB	Nom de l'utilisateur Postgres	'OCPIZUSER'
PASSWORD_DB	mot de passe de l'utilisateur Postgres	variable aléatoire sur 12 digits mini conseillé
EMAIL_HOST_PASSWORD	clé pour Sendgrid (Service d'email)	clé disponible sur le compte Sendgrid

Nom	Description	Valeur
NEW_RELIC_CONFIG_FILE	Emplacement du fichier de configuration de NewRelic	'/home/ubuntu/oc-pizza/newrelic.ini'

Pour créer une variable:

```
export ENV=PRODUCTION
```

Renseigner les mêmes valeurs de variables dans le fichier de configuration de Supervisor. Veillez à avoir les variables renseignées sur le dépôt pour ne pas devoir refaire la manipulation à chaque *pull*

```
nano ~/oc-pizza/conf/supervisor/oc-pizza.conf
```

une copie de ce fichier sera faite dans ~/conf/ lors du premier déploiement.

Création de la base de données

Remplacez '*****' par le mot de passe renseigné dans la variable d'environnement *PASSWORD_DB*

```
sudo -u postgres psql
CREATE DATABASE OCPIZZA;
CREATE USER OCPIZUSER WITH PASSWORD '*****';
ALTER ROLE OCPIZUSER SET client_encoding TO 'utf8';
ALTER ROLE OCPIZUSER SET default_transaction_isolation TO 'read
committed';
ALTER ROLE OCPIZUSER SET timezone TO 'Europe/Paris';
GRANT ALL PRIVILEGES ON DATABASE OCPIZZA TO OCPIZUSER;
```

Lancez le script de création des profils et de peuplage de la base avec les pizzas et les boutiques déjà renseignées. (Cette manipulation ne sera plus jamais réalisée et sera faite à partir d'une sauvegarde)

```
~/oc-pizza/scripts/inject-db.sh ~/oc-pizza/conf/init-db.sql
```

Lancer le script de déploiement

Faire un push sur le dépôt sur la branche master afin de lancer le processus de déploiement automatique ou vous pouvez exécuter le script manuellement.

```
~/oc-pizza/scripts/deploy.sh
```

Les étapes réalisées par le script de déploiement

- Activation de l'environnement virtuel
- Extraction de l'archive contenant l'application (archive du dépôt) ou *git pull* en cas de lancement manuelle du script
- Installation des packages du fichier *requirements.txt*
- Vérification de la présence de toutes les variables d'environnement
- Test de connexion à la base de données (vérification de l'accès et présence de la base)
- Collecte des fichiers statics
- Vérification de la tâche de sauvegarde de la base de données (ou création si non présent dans le *crontab*)
- Lancement d'une sauvegarde de la base de donnée
- Migration de la base de donnée
- Copie des fichiers de configuration de *~/oc-pizza/conf* vers *~/conf/* si ils n'existent pas.
- Création de liens symboliques pour les fichiers de configuration de Nginx et Supervisor, mise à jour et redémarrage des services.

Configuration

La configuration de l'application se trouve dans les fichiers

```
~/oc-pizza/core/settings/__init__.py
~/oc-pizza/core/settings/production.py
```

Les fichiers static se trouvent dans ce dossier

```
~/oc-pizza/staticfiles/
```

Le fichier de configuration de nginx

```
~/conf/nginx/oc-pizza.conf
```

Le fichier de configuration de Supervisor

```
~/conf/supervisor/oc-pizza.conf
```

Procédures de maintenance

Une sauvegarde de la base de donnée à lieu tous les jours à 3h dans un fichier nommé OCPDB_YYYYMMDDHHHH.sql dans un dossier *~/backup_db/*

Un historique de 14 jours est sauvegardé.

la tache cron

```
0 3 * * * ~/oc-pizza/scripts/cronjob.sh
```

Procédure de démarrage/arrêt

Pour arrêter l'application

```
sudo supervisorctl stop oc-pizza-gunicorn
```

Pour démarrer l'application

```
sudo supervisorctl start oc-pizza-gunicorn
```

Procédure de manipulation de la base de données

Faire une sauvegarde manuelle

```
~/oc-pizza/scripts/backup-db.sh
```

Injecter une sauvegarder dans la base. Remplacez `YYYYMMDDHHHH` par la valeur appropriée.

ATTENTION CELA SUPPRIME LES DONNEES EN COURS, FAITES UNE SAUVEGARDE AVANT.

```
~/oc-pizza/scripts/inject-db.sh ~/backup_db/OCADB_YYYYMMDDHHHH.sql
```

Outils de surveillance

Surveillance du trafic

Le trafic de l'application est surveillé par Newrelic qui alertera si une activité intense met en défaut le serveur.

Vous pouvez consulter l'activité sur le site de [NewRelic](#). La configuration est déjà en place.

Surveillance de l'application

L'application est surveillée par [Sentry](#)