



OC Pizza

Projet 9

Dossier d'exploitation

Version 1.0

Auteur

Emmanuel Nocquet
Développeur web

TABLE DES MATIÈRES

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 - Serveur de Base de données.....	5
3.1.1.1 - Caractéristiques techniques.....	5
3.1.2 - Serveur Web.....	5
3.2 - Bases de données.....	5
3.3 - Web-services.....	5
4 - Procédure de déploiement.....	6
4.1 - Déploiement de l'application Web.....	6
4.1.1 - Variables d'environnement.....	6
4.1.2 - Configuration.....	6
4.1.2.1 - Fichier procfile.....	6
4.1.2.2 - Fichier settings.py.....	6
4.1.2.3 - Fichier requirements.txt.....	6
4.1.3 - Déploiement.....	7
4.1.3.1 - Pré-requis.....	7
4.1.3.2 - Connexion au compte "Heroku".....	7
4.1.3.3 - Création de l'application.....	7
4.1.3.4 - Création des variables d'environnement.....	7
4.1.3.5 - Création du lien entre l'application et le dépôt "Git".....	7
4.1.4 - Vérifications.....	8
4.2 - Déploiement de la base de données.....	8
4.2.1 - Schémas de la base de données.....	8
4.2.2 - Création d'un superutilisateur.....	8
4.2.3 - Import des données.....	9
4.2.4 - Vérifications.....	9
5 - Procédure de démarrage / arrêt.....	10
5.1 - Application web.....	10
6 - Procédure de mise à jour.....	11
7 - Supervision/Monitoring.....	12
7.1 - Supervision de l'application web.....	12
8 - Procédure de sauvegarde et restauration.....	13
9 - Glossaire.....	14

1 - VERSIONS

Auteur	Date	Description	Version
E.Nocquet	01/05/2020	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.

Ce document contient les informations dont l'équipe d'exploitation a besoin pour pouvoir assurer une exploitation en règle du système et pouvoir réagir de manière appropriée lorsqu'un problème surgit.

Il contient les instructions à suivre pour le déploiement et la maintenance de l'application OC Pizza.

2.2 - Références

Pour de plus amples informations, se référer :

1. **DCT - 001** : Dossier de conception technique de l'application
2. **DCF – 002** : Dossier de conception fonctionnelle de l'application

3 - PRÉ-REQUIS

3.1 - Système

3.1.1 - *Serveur de Base de données*

Serveur de base de données hébergeant les schémas de base de données tels que décrits dans le dossier de conception technique de l'application.

3.1.1.1 - *Caractéristiques techniques*

Le type de base de données utilisé est PostgreSQL en version 12.1.

3.1.2 - *Serveur Web*

Mis en place et entretenu par la plateforme « Heroku » sur laquelle l'application « OC Pizza » sera installée.

3.2 - Bases de données

La base de données est hébergée sur les serveurs de la plateforme "Heroku". Elle est mise en place automatiquement par Heroku lors de l'installation de l'application "OC Pizza".

3.3 - Web-services

Les web services suivants doivent être accessibles et à jour :

- un compte existant sur la plateforme "Heroku": les paiements doivent être à jour.
- Un système de paiement en ligne est requis: "Payline" est un service avec lequel nous avons l'habitude de travailler

4 - PROCÉDURE DE DÉPLOIEMENT

4.1 - Déploiement de l'application Web

4.1.1 - Variables d'environnement

Voici les variables d'environnement nécessaires au fonctionnement de l'application OC Pizza :

Nom	Obligatoire	Description
ENV	Oui	Sa valeur doit être "PRODUCTION". Elle permet d'indiquer à l'application qu'elle se trouve en environnement de production et non de développement
SECRET_KEY	Oui	Sa valeur est : 'e\$x\\x0cgcEtAs0'[d\\lhG*+m". Elle permet à l'application de fonctionner et ne s'applique qu'à l'environnement de production

Ces variables d'environnement sont définies au moment du déploiement.

4.1.2 - Configuration

Voici les différents fichiers de configuration :

- **procfile** : ce fichier est indispensable à Heroku pour pouvoir lancer l'application en tant que service web
- **settings.py** : fichier de configuration de l'application Django
- **requirements.txt** : fichier qui liste les ressources nécessaires au fonctionnement de l'application web

4.1.2.1 - Fichier procfile

Il contient des instructions qui vont être interprétée par Heroku. Son contenu est à minima le suivant:

```
web: gunicorn oc_pizza.wsgi
```

4.1.2.2 - Fichier settings.py

Ce fichier contient des éléments qui ne devraient pas être modifiés sous peine de dysfonctionnements de l'application.

4.1.2.3 - Fichier requirements.txt

Ce fichier indique à Heroku qu'il s'agit d'une application Python, liste les dépendances nécessaires à l'application pour son bon fonctionnement. Lors de l'installation de l'application, ces dépendances seront installées avant le démarrage de l'application.

4.1.3 - **Déploiement**

Plusieurs méthodes sont possibles pour effectuer un déploiement sur la plateforme Heroku: notre préférence va à l'utilisation de la ligne de commande via "Heroku CLI". Le déploiement nécessite de lier une application créée sur Heroku avec un dépôt "Git". En l'espèce, l'application sera liée au dépôt Git du projet "OC Pizza" contenant son code source.

4.1.3.1 - **Pré-requis**

Un compte devra avoir été créé au préalable sur la plateforme "Heroku". Le code source de l'application sera disponible sur un dépôt "Git".

4.1.3.2 - **Connexion au compte "Heroku"**

```
$ heroku login
```

Entrez les identifiant et mot de passe ayant servi à la création du compte.

4.1.3.3 - **Création de l'application**

```
$ heroku create oc_pizza
```

Nous venons de créer une application "oc_pizza" sur Heroku;

4.1.3.4 - **Création des variables d'environnement**

Création de la variable "ENV":

```
$ heroku config:set ENV='PRODUCTION'
```

Création de la variable "SECRET_KEY":

```
$ heroku config:set SECRET_KEY='.e$xl\\x0cgcEtAs0'[d\\lhG*+m'''
```

Vérification que l'environnement de production est correctement configuré en tapant:

```
$ heroku config
```

Cette commande affichera normalement les deux variables que vous venez de créer.

4.1.3.5 - **Création du lien entre l'application et le dépôt "Git"**

Cette commande lance l'installation de l'application sur Heroku:

```
$ git push heroku master
```

4.1.4 - **Vérifications**

La dernière commande entraîne une suite d'opérations automatisées. Une fois celles-ci terminées, le terminal devrait afficher les lignes suivantes:

```
$ oc_pizza deployed to Heroku
$ Verifying deploy... done
```

Le site est maintenant accessible via internet à l'adresse "https://oc_pizza.herokuapp.com/".

Il est toujours possible d'accéder aux informations de l'application déployée via la CLI Heroku, celle-ci met à disposition tout un ensemble de commandes, dont voici quelques unes:

Accéder aux applications hébergées sur la plateforme:

```
$ heroku apps
```

Consulter les informations sur nos applications:

```
$ heroku status
```

4.2 - **Déploiement de la base de données**

Comme indiqué dans les pré-requis, la base de données est mise en place automatiquement par Heroku lors du déploiement de l'application web "OC Pizza": il s'agit d'une base PostgreSQL.

Cette mise en place ne va nécessiter que quelques opérations avant que cette base de données ne soit parfaitement fonctionnelle.

A nouveau, lors de cette étape, nous utiliserons la ligne de commande via "Heroku CLI".

4.2.1 - **Schémas de la base de données**

Les modèles de données de l'application "OC Pizza" vont servir de schémas pour la création des différentes tables de la base de données.

Cette opération s'effectue avec la commande suivante:

```
$ heroku run python manage.py migrate
```

4.2.2 - **Création d'un superutilisateur**

A présent, il s'agit de créer un super utilisateur, qui aura des droits étendus dans l'administration de l'application:

```
$ heroku run python manage.py createsuperuser
```


4.2.3 - *Import des données*

La commande "loaddata" permet d'importer des données issues d'une précédente version de l'application dans la base de données tout juste créée.

La commande dumpdata peut être utilisée pour générer le contenu donné à loaddata:

```
$ manage.py dumpdata oc_pizza > oc_pizza.json
```

Nous pouvons maintenant charger cette sauvegarde dans notre nouvelle base de données avec la commande loaddata:

```
$ heroku run python manage.py loaddata oc_pizza.json
```

4.2.4 - *Vérifications*

Il est possible de vérifier le bon déploiement de l'application de plusieurs manières: la première et plus évidente est de se rendre via internet à l'url de l'application "https://oc_pizza.herokuapp.com/".

Une deuxième possibilité est d'utiliser la ligne de commande (Heroku CLI) pour ouvrir un shell avec la base de données avec la commande:

```
$ heroku psql
```

Néanmoins, cette méthode requiert de l'utilisateur certaines connaissances en base de données et en langage SQL.

Enfin, la dernière possibilité est de se connecter à la plateforme Heroku via internet et d'accéder au dashboard de l'application "OC Pizza". En suivant le chemin: 'Ressources', puis 'Add-ons' et enfin en cliquant sur la base de données 'heroku postgres', on accède aux informations utiles sur la base de données.

5 - PROCÉDURE DE DÉMARRAGE / ARRÊT

5.1 - Application web

Comme indiqué lors du déploiement de l'application sur la plateforme Heroku, l'application web est accessible dès son déploiement effectué.

Afin de démarrer ou arrêter l'application, depuis la ligne de commande 'Heroku CLI', effectuer les commandes suivantes:

Pour démarrer l'application:

```
$ heroku run
```

Pour arrêter l'application:

```
$ heroku ps: scale web=0 --app oc_pizza
```

6 - PROCÉDURE DE MISE À JOUR

La mise à jour de l'application est possible en passant le site en "maintenance".

Cela s'effectue directement depuis le dashboard Heroku via internet, ou au moyen des commandes suivantes avec la CLI Heroku.

Mise de l'application web en mode maintenance:

```
$ heroku maintenance: on
```

A l'issue de nos opérations sur l'application, celle-ci peut être réactivée avec la commande:

```
$ heroku maintenance: off
```

7 - SUPERVISION/MONITORING

7.1 - Supervision de l'application web

Le monitoring de l'application peut être effectué depuis le dashboard Heroku: dans la rubrique 'metrics' de l'application "OC_Pizza".

Parce que le monitoring dépend du type d'application web et du désir du client, il vous appartient de configurer ces éléments selon vos besoins.

Une documentation très complète sur le sujet est disponible à l'adresse <https://devcenter.heroku.com/articles/metrics> .

Néanmoins, un bon début dans la supervision de n'importe quel site web est de définir une politique de gestion des logs.

8 - PROCÉDURE DE SAUVEGARDE ET RESTAURATION

Effectuer des sauvegardes régulières de vos données est essentiel: dans le cadre d'un suivi normal d'une application web mais également en cas d'incident, que celui-ci soit dû à un dysfonctionnement technique, une opération de maintenance, ou même une action malveillante.

Il est possible de suivre le cheminement expliqué au *4.2 Déploiement de la base de données* -

4.2.3 Import des données pour effectuer une sauvegarde du contenu de la base de données et le restaurer.

Heroku propose également ce service via l'exécution des commandes suivantes utilisables grâce à la CLI Heroku:

Sauvegarde de la base de données:

```
$ heroku pg:backups:capture <nom_base_de_données> --app oc_pizza
```

Restauration de la base de données:

```
$ heroku pg:backups:restore < nom_backup> <nom_base_de_données> --app oc_pizza
```

9 - GLOSSAIRE

Dashboard	Littéralement 'tableau de bord', interface faisant ressortir les différentes caractéristiques et l'état d'un système. Le dashboard Heroku est accessible à l'adresse: https://dashboard.heroku.com/apps une fois connecté à Heroku.
URL	Adresse complète d'un site hébergé sur internet
SQL	SQL est un langage informatique normalisé servant à exploiter des bases de données relationnelles
Shell	Le shell est un programme qui reçoit des commandes informatiques données par un utilisateur à partir de son clavier pour les envoyer au système d'exploitation qui se chargera de les exécuter
Heroku CLI	Interface en ligne de commande du service Heroku. Une liste complète des commandes et fonctionnalités est disponible à l'adresse: https://devcenter.heroku.com/articles/heroku-cli-commands
Backup	En informatique, la sauvegarde est l'opération qui consiste à dupliquer et à mettre en sécurité les données contenues dans un système informatique
Logs	Historique des événements ou de journalisation. Désigne l'enregistrement séquentiel dans un fichier ou une base de données de tous les événements affectant un processus particulier.