

OC PIZZA

APPLICATION PIZZAPP

DOSSIER D'EXPLOITATION

version 1.0

Villatte lorelei

TABLE DES MATIERES**I. VERSIONS**

Tableau	3
---------	---

II. INTRODUCTION

2.1 OBJET DU DOCUMENT	3
2.2 REFERENCES	3

III. PRE-REQUIS

3.1 SYSTEME	5
3.2 BASE DE DONNEES	5
3.3 WEB SERVICES	5

IV. PROCEDURE DE DEPLOIEMENT

4.1 DEPLOIEMENT DE L'APPLICATION WEB	6
4.1.1 Artefacts	6
4.1.2 Variables d'environnement	6
4.1.3 Configuration	7
4.1.4 Data sources	7
4.1.5 Déploiement	7 - 8
4.1.6 Déploiement de la base de données	8

V. PROCEDURE DE DEMARRAGE / ARRET

5.1 APPLICATION WEB	9
---------------------	---

VI. PROCEDURE DE MISE A JOUR

6.1 USAGE	9 - 10
-----------	--------

VII. SUPERVISION / MONITORING

7.1 SUPERVISION DE L'APPLICATION WEB	10
--------------------------------------	----

VIII. PROCEDURE DE SAUVEGARDE ET RESTAURATION

8.1 USAGE	11
-----------	----

IX. GLOSSAIRE

Tableau	12
---------	----

I. VERSIONS

AUTEUR	DATE	DESCRIPTION	VERSION
Villatte Lorelei	14/02/2019	Création du dossier d'exploitation	1.0

II. INTRODUCTION

2.1 OBJET DU DOCUMENT

Le présent document constitue le dossier d'exploitation de l'application **PIZZAPP**.

Ce document a pour vocation de détailler les différentes informations techniques de l'application et leurs utilisations.

2.2 REFERENCES

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

P9 - DCF- 1.0 : Dossier de conception fonctionnelle de l'application

P9 - DCT - 1.0 : Dossier de conception technique de l'application

III. PRE-REQUIS

3.1 SYSTEME

L'application **PIZZAPP** est hébergé sur la plateforme d'Heroku lié au nom de domaine www.ocpizza.fr.

3.2 BASE DE DONNEES

Utilisation de PostgreSQL version 11, hébergé sur **Heroku**.

3.3 WEB SERVICES

Les web services suivants doivent être accessibles et à jour :

Google Maps API : afin de pouvoir utilisé ce service il est impératif d'utiliser la clé suivante **AlzaSyVguc-DHx8T4vKUqJJxRxHQQNYAxTFD4Qc**

Son service étant payant son règlement doit être effectué à temps sous peine de dysfonctionnement de l'application.

Stripe API : afin de pouvoir utilisé ce service il est impératif d'utiliser la clé suivante **sk_34Rs_4eC39HqLyjWDarjtT1zdp7dc**

Son service étant payant son règlement doit être effectué à temps sous peine de dysfonctionnement de l'application.

IV. PROCEDURE DE DEPLOIEMENT

4.1 DEPLOIEMENT DE L'APPLICATION WEB

4.1.1 Artefacts

L'application **OCPIZZA** contient les répertoires suivants :

EMPLACEMENT	ROLE
./docs	Documentation de référence des paquets.
./procfile	Instruction démarrage application.
./requirements.txt	Dépendances de développement.
./pizzap/	Fichiers de configuration de l'application Django
./production/	Fichiers liés au package Production
./static/	Fichiers .css et .js
./templates/	Fichiers HTML de l'application Django
./ventes/	Fichiers liés au package Ventes

Procfile

Cette ligne contient la commande à exécuter par Heroku pour tout lancement de l'application

```
web: gunicorn pizzapp.wsgi
```

4.1.2 Variables d'environnement

NOM	VALEUR	OBLIGATOIRE	DESCRIPTION
ENV	PRODUCTION	oui	Environnement de développement
SECRET_KEY	--a0; F;rE[?/? w^zcumh{9	oui	Clé secrète de production Django

4.1.3 Configuration

les différents fichiers de configuration :

./procfile

fichier contenant la commande utilisée par Heroku pour le déploiement de l'application.

./requirements.txt

fichier contenant la listes et versions des librairies à installer sur le serveur Heroku.

./pizzapp/pizzapp/settings.py

fichier de configuration de l'application Django

4.1.4 Data sources

La base de données doit pouvoir être utilisée en production. Il est donc nécessaire de l'exporter pour l'importer plus tard sur le serveur Heroku.

Création d'un fichier dump

```
$ ./manage.py dumpdata > pizzapp/pizzapp.json
```

4.1.5 Déploiement

Les étapes qui suivent nécessite une intallation d'**Heroku CLI** afin de pouvoir déployer l'application avec **Git** en ligne de commande.

Création de l'application

Se rendre à la racine du projet de l'application et entre la commande suivante :

```
$ heroku create pizzapp
```

Création variables d'environnement

```
$ heroku config:set SECRET_KEY='--a0;|F;rE[?/?w^zcumh{9}'  
$ heroku config:set ENV='PRODUCTION'
```

Afin de vérifier que tout a bien été enregistré, utiliser :

```
$ heroku config
```

Envoyer les fichiers de l'application sur le serveur

```
$ git push heroku master
```

4.1.6 Déploiement de la base de données

Lancement des migrations

Avant d'importer les données, configurer les tables de base de données:

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

Création d'un super utilisateur :

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

Import base de données :

```
$ heroku run manage.py loaddata pizzapp/pizzapp.json
```

4.2 VERIFICATIONS

Pour le déploiement de l'application, après avoir effectuer **\$ git push heroku master** le message suivant doit apparaître dans le terminal

```
remote:https://pizzapp.fr/ deployed to Heroku
remote: Verifying deploy... done.
To https://git.heroku.com/pizzapp.git
```

Se rendre à l'adresse www.pizzapp.fr. pour contrôler.

La vérification de la base de données peut se faire directement via le site Heroku Dashbord → pizzapp → heroku Postgres

V. PROCEDURE DE DEMARRAGE / ARRET

5.1 APPLICATION WEB

Deux possibilités :

Se rendre sur le site Heroku :
Dashbord → pizzap → Settings → Maintenance
mode On pour arrêt et off pour le démarrage

Depuis le terminal:
Se placer à la racine du projet

Arrêt :

```
$ heroku ps:scale web=0
```

Mise en marche:

```
$ heroku run
```

V.I PROCEDURE DE MISE A JOUR

6.1 USAGE

Activer le mode maintenance :

```
$ heroku maintenance:on
```

Désactiver le mode maintenance:

```
$ heroku maintenance:off
```

Pour vérifier l'état de maintenance actuel de l'application

```
$ heroku maintenance
```

Personnaliser la page de maintenance :

Définir les vars `ERROR_PAGE_URL` et `MAINTENANCE_PAGE_URL` config sur les URL accessibles au public des pages personnalisées:

```
$ heroku config:set MAINTENANCE_PAGE_URL="//s3.amazonaws.com/<your_bucket>/  
your_maintenance_page.html
```

Plus d'informations à l'adresse suivante :

<https://devcenter.heroku.com/articles/error-pages#customize-pages>

VII. SUPERVISIONS MONITORING

7.1 SUPERVISION DE L'APPLICATION WEB

La plateforme heroku fournit des outils de monitoring à la rubrique **Metrics** du projet

En plus des mesures brutes, Heroku fournit des notifications en ligne de conditions spécifiques qui pourraient indiquer des problèmes avec l'application. Des liens vers des articles pertinents du Centre de développement sont inclus pour fournir des recommandations sur la façon de résoudre le problème. Ces exemples incluent des alertes sur les erreurs de mémoire, les délais d'attente de demande et les temps de réponse lents.

Plus de détails à l'adresse suivante <https://devcenter.heroku.com/articles/metrics>

VI. PROCEDURE SAUVEGARDE RESTAURATION

6.1 USAGE

Créer une sauvegarde :

```
$ heroku pg:backups:capture --app pizzapp
```

Planifier une sauvegarde:

```
$ heroku pg:backups:schedule DATABASE_URL --at '02:00 Europe/Paris' --app pizzapp
```

Restaurer les sauvegardes

```
$ heroku pg:backups:restore DATABASE_URL --app pizzapp
```

Plus d'informations à l'adresse suivante :

<https://devcenter.heroku.com/articles/heroku-postgres-backups>

VIII. GLOSSAIRE
