|  |
| --- |
| **OC PIZZA**  **Boutique en ligne**  **&**  **Gestionnaire centralisé**  Dossier d'exploitation  Version 1.0 |
| **Auteur**  Stephen AOGOLO  *Analyste-programmeur* |

Table des matières

1 -Versions 4

2 -Introduction 5

2.1 -Objet du document 5

2.2 -Références 5

3 -Pré-requis 6

3.1 -Système 6

3.1.1 -Serveur de Base de données 6

3.1.1.1 -Caractéristiques techniques 6

3.1.2 -Serveur Web 6

3.1.2.1 -Caractéristiques techniques 6

3.1.3 -Serveur de Batches 6

3.1.4 -Serveur de Fichiers 6

3.2 -Bases de données 6

3.3 -Web-services 6

3.4 -Autres Ressources 6

4 -Procédure de déploiement 7

4.1 -Déploiement des Batches 7

4.1.1 -Artefacts 7

4.1.2 -Variables d'environnement 7

4.1.3 -Configuration 7

4.1.3.1 -Fichier xxx.yyy 8

4.1.3.2 -Fichier zzz.ttt 8

4.1.3.3 -Fichier ... 8

4.1.4 -Ressources 8

4.1.5 -Vérifications 8

4.2 -Déploiement de l'Application Web 9

4.2.1 -Artefacts 9

4.2.2 -Environnement de l’application web 9

4.2.2.1 -Variables d’environnement 9

4.2.3 -Répertoire de configuration applicatif 9

4.2.3.1 -Fichier xxx.yyy 9

4.2.4 -DataSources 9

4.2.5 -Ressources 10

4.2.6 -Vérifications 10

5 -Procédure de démarrage / arrêt 11

5.1 -Base de données 11

5.2 -Batches 11

5.3 -Application web 11

6 -Procédure de mise à jour 12

6.1 -Base de données 12

6.2 -Batches 12

6.3 -Application web 12

7 -Supervision/Monitoring 13

7.1 -Supervision de l’application web 13

8 -Procédure de sauvegarde et restauration 14

9 -Glossaire 15

# Versions

|  |  |  |  |
| --- | --- | --- | --- |
| Auteur | Date | Description | Version |
| Stephen AOGOLO | 14/01/2021 | Création du document | 1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Objet du document

Ce dossier d’exploitation (***version 1.0***) fait suite au dossier de spécifications technique (***version 2.0***).

L’objectif de ce dossier d’exploitation se partage en deux parties.

Dans un premier temps, ce dossier reprend et décrit techniquement le contenu vital du système. Par la suite il renseigne sur les procédures nominales d’installation, d’utilisation et de maintenance du système. Ce document présente et décrit donc le déploiement du produit **OC PIZZA – Boutique en ligne & Gestionnaire centralisé**.

Les points suivants seront donc traités afin de garantir le meilleur niveau d’information quant aux manipulations post-production :

* La description technique des sous-systèmes
* La procédure nominale du déploiement
* La procédure nominale de mise en route et d’extinction
* La procédure nominale de mise à jour
* Les actions et méthodes nominales de supervision
* Les actions et méthodes nominales de sauvegarde
* Les actions et méthodes nominales de restauration

*Il est important de noter que la mention «****nominale****» fait référence à la manière standard et recommandé d’effectuer ces opérations. Dans une version future du système et donc un futur dossier d’exploitation, les procédures, les actions et les méthodes «****d’urgences****» seront disponibles afin d’intervenir dans des cas particuliers de dépannage du système.*

## Références

Pour de plus amples informations, merci de se référer aux documents suivants :

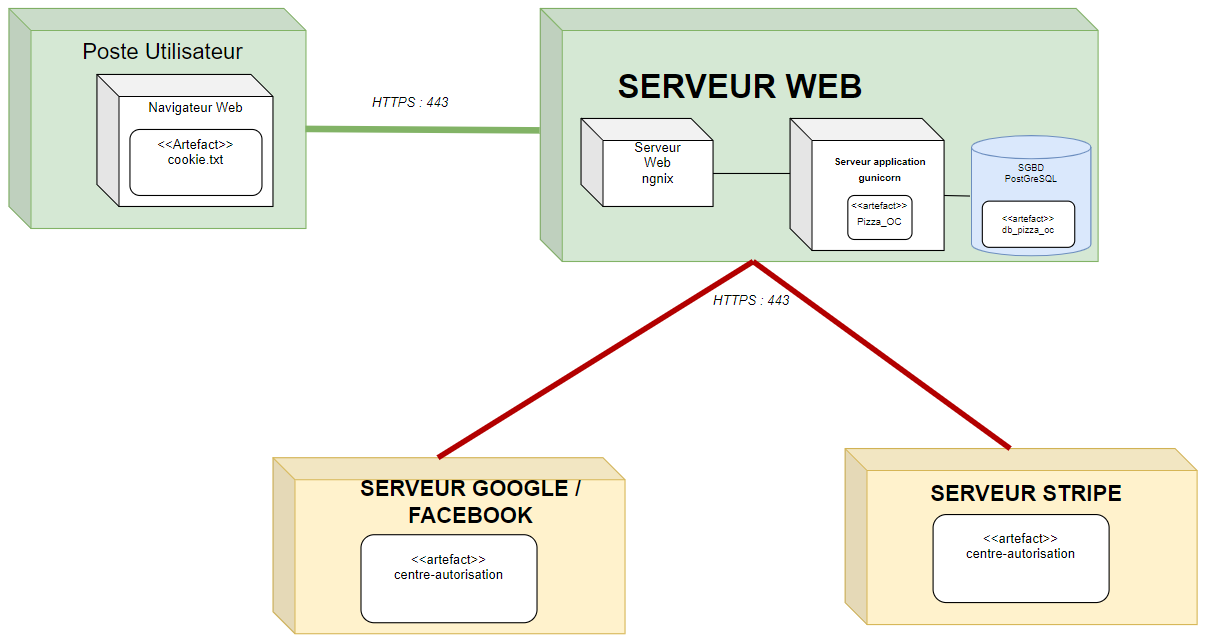
|  |  |
| --- | --- |
| REFERENCES | VERSIONS |
| Dossier de conception fonctionnelle | 1.4 |
| Dossier de conception technique | 2.0 |
| Procès-verbal de livraison | 1.0 |

# Pré-requis

## Système

Cette partie reprend la description technique des composants du système.

L’ensemble des composants du système seront hébergés chez DIGITAL OCEAN. Ils occuperont un emplacement virtuel. Ci-dessous, le diagramme de déploiement, extrait du dossier de conception.



Ce diagramme reprend donc l’ensemble des composants principaux du système à déployer.

*PS : L’artefact « cookie.txt » sera installé automatiquement après la première connexion de l’utilisateur. Il n’est donc pas concerné par les procédures de déploiement, de démarrage et de maintenance.*

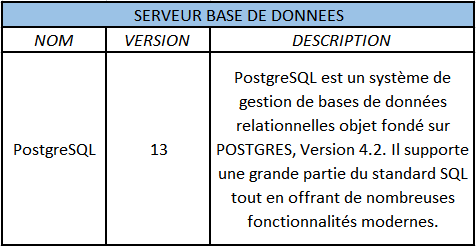
### Serveur de Base de données

Serveur de base de données hébergeant le/les schémas/base Xxxx….

Le serveur de base de données utilisé est de type **Postgresql**.

#### Caractéristiques techniques

Le tableau ci-dessous reprend la description technique du serveur de base de données.



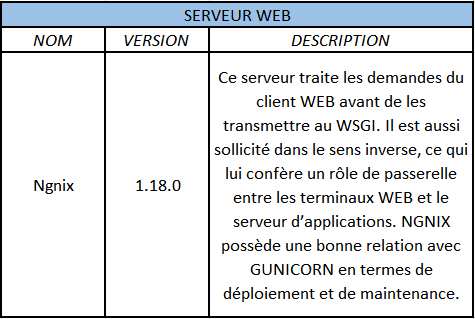
### Serveur Web

Serveur physique ou virtuel hébergeant l'application web.

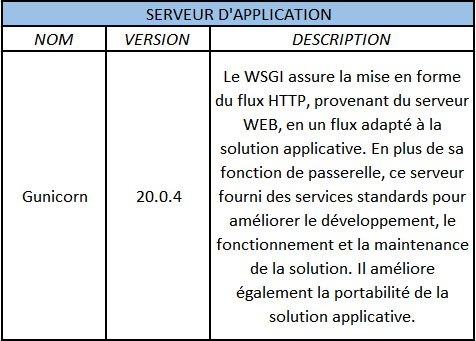
Le serveur web est de type **NGNIX**. Il est le support du serveur d’application web de type **Gunicorn**.

#### Caractéristiques techniques

Le tableau ci-dessous reprend la description technique du serveur web.



Le tableau ci-dessous reprend la description technique du serveur d’application web.



### Serveur de Batches

...

### Serveur de Fichiers

...

## Bases de données

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

* **… :** version XXX

## Web-services

Les web services suivants doivent être accessibles et à jour :

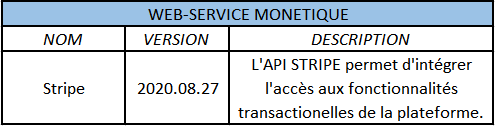
Les web-services sont des composants externes à la solution qui apportent des fonctionnalités supplémentaires au système. Elles sont implémentées via leur API respective.

*Pour de plus d’informations sur le rôle et la mise en place des web-services de la solution, consultez le* ***dossier de conception technique*** *à la partie* ***Règles de gestion Architecture Technique****.*

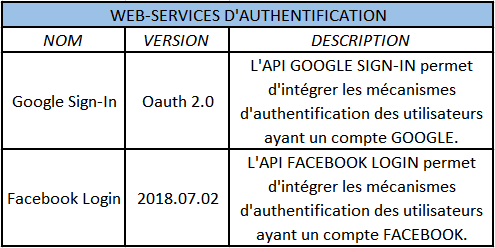
Le système utilise trois web-services classés en deux groupes :

* Un web-service monétique : Stripe
* Deux web-services d’authentification : Google Sign-in et Facebook Login

Le tableau ci-dessous reprend la description technique du web-service monétique.



Le tableau ci-dessous reprend la description technique des web-services d’authentification.



## Autres Ressources

...

# Procédure de déploiement

La procédure de déploiement suivante doit être respectée, afin de garantir le bon déploiement du site web **OC PIZZA**.

A l’état initial, l’ensemble de la solution se trouve sur GITHUB à l’adresse : [***https://github.com/StephenAOGOLO/P9\_Documentez\_votre\_systeme\_de\_gestion\_de\_pizzeria.git***](https://github.com/StephenAOGOLO/P9_Documentez_votre_systeme_de_gestion_de_pizzeria.git)

Les étapes de déploiement

Le site web de type DJANGO sera installé sur une machine publique de type UBUNTU. Des composants LINUX-UBUNTU seront donc installés pour prendre en charge et servir les applications DJANGO.

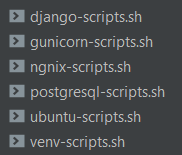
La base de données POSTGRESQL sera configurée afin de gérer les données applicatives.

Le serveur d’application GUNICORN sera installé et configuré pour s’interfacer avec les applications du site.

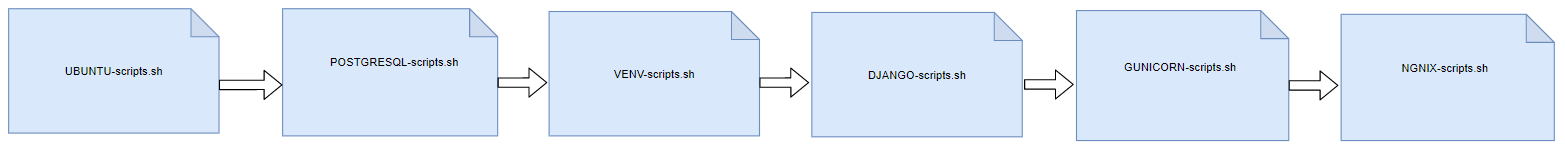
Enfin, le serveur web NGNIX sera installé et configuré pour gérer la communication entre le service GUNICORN et internet, et ainsi, profiter des fonctionnalités de sécurité et de performance.

## Déploiement des Batches

Le déploiement du site web **OC PIZZA** est semi-automatique. Il s’appuie sur l’exécution de plusieurs scripts partiellement rédigés nécessitant des précisions complémentaires. Il y a six scripts ou artefacts à configurer avant qu’elles ne soient utilisées pour le déploiement semi-automatique du site :

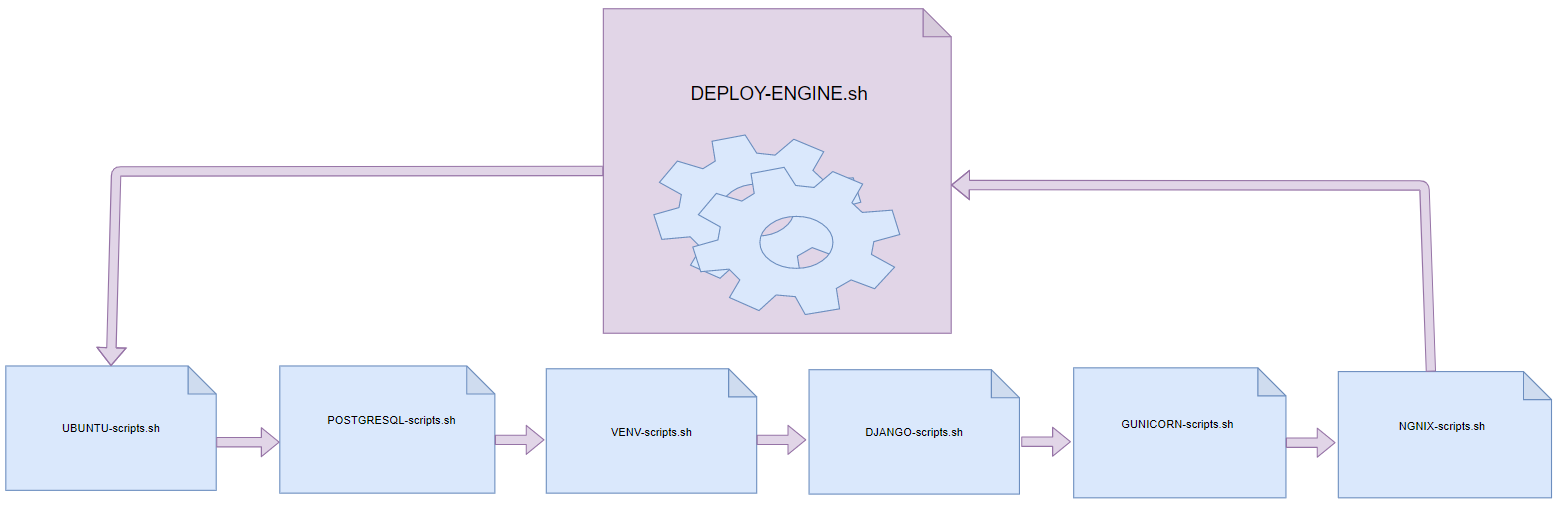


Ces artefacts seront exécutés dans l’ordre précis illustré ci-dessous :



Chaque artefact est une étape de déploiement. Elle contient chacune les opérations d’installation, de configuration et de lancement de cette dernière. Avant l’exécution de ces scripts, il est important de les édités afin de corriger et confirmer leur contenu.

L’exécution des six scripts cités précédemment est géré par un gestionnaire de déploiement « deploy\_engine ». Il assure la configuration et l’installation générale des étapes de déploiement.



### Artefacts

Les batches de l’application Xxx sont construits sous la forme d'une archive ZIP contenant les répertoires :

* **bin** : les scripts SH de lancement des différents batches
* **conf** : les fichiers de configuration
* **...** : ...

Extraire l'archive **XXX.zip** dans le répertoire :

**/xxx/yyy**

Positionner les droits d'exécution sur les scripts SH de lancement des batches.

….

### Variables d'environnement

Voici les variables d'environnement reconnues par les batches de l’application XXX :

|  |  |  |
| --- | --- | --- |
| Nom | Obligatoire | Description |
| XXX\_HOME | non | Répertoire racine de l’installation de l’application |
|  |  |  |

Définissez les variables d’environnement nécessaires comme ceci...

...

### Configuration

Voici les différents fichiers de configuration :

* **xxx.yyy** : fichier de configuration des logs
* **zzz.ttt** : fichier de configuration de l'application...
* **aaa** : fichier de configuration de la ressources XXX
* ...

#### Fichier xxx.yyy

…

#### Fichier zzz.ttt

...

#### Fichier ...

### Ressources

...

### Vérifications

Afin de vérifier le bon déploiement des batches, faire ceci…

## Déploiement de l'Application Web

### Artefacts

...

### Environnement de l’application web

#### Variables d’environnement

Le serveur d'application JOnAS doit être exécuté avec la variable d'environnement suivante définie au démarrage. Elle est nécessaire afin de récupérer le répertoire contenant les fichiers de configuration de l'application :

**-Dcom.ocpizza.apps.conf=$home\_application\_conf\_directory**

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.

### Répertoire de configuration applicatif

Le répertoire de configuration applicatif doit être créé sur le système de fichier et définit de la façon suivante :

**$home\_application\_conf\_directory/applicationX**

… fichiers de configuration… :

* ...

#### Fichier xxx.yyy

...

### DataSources

Les accès aux bases de données doivent se configurer à l'aide des fichiers…

Le fichier de drivers **postgresql (postgresql-9.2.x.)** doit être déposé dans le répertoire :

**$home\_server/lib/ext**

...

### Ressources

...

### Vérifications

Afin de vérifier le bon déploiement de l’application, faire ceci…

# Procédure de démarrage / arrêt

## Base de données

## Batches

## Application web

# Procédure de mise à jour

## Base de données

## Batches

## Application web

# Supervision/Monitoring

## Supervision de l’application web

Afin de tester que l’application web est toujours fonctionnelles, faire ceci…

# Procédure de sauvegarde et restauration

# Glossaire

|  |  |
| --- | --- |
| **API** | Sigle anglais signifiant « Application Programming Interface », interface de programmation applicative en français. C’est une librairie de classes, fonctions et méthodes qui proposent un service WEB particulier. |
| **BATCH** | Terme anglais utilisé en informatique. C’est un proche synonyme de « script informatique ». C’est donc un fichier exécutable par une machine qui contient des instructions rédigées par un humain. Selon sa rédaction, il peut à son tour demander l’exécution d’un ou plusieurs autres scripts, on parle donc de « batch ou script en cascade ». |
| **DIGITAL OCEAN** | Fournisseur d’infrastructure cloud américain, hébergeur américain en d’autres termes dont le siège est basé à New York. La couverture en centre données occupe le monde entier. Cet hébergeur propose la plus part formule standard d’hébergement, du physique au virtuel. |
|  |  |
| **WSGI** | Sigle anglais signifiant « Web Server Gateway Interface ». C’est un type de serveur, voire une spécification qui définit une interface entre des serveurs et des applications web pour le langage Python. |
|  |  |