

# MISE EN PRODUCTION

## 1 - Créer un fichier robots.txt

C'est un fichier de configuration pour le ou les bots Google. C'est un fichier qui permet de donner des instructions aux robots d'indexation Google. C'est pour lui simplifier la tâche et potentiellement lui dire ce qu'il doit faire (indexer) et pas faire (ne pas indexer). Ex: n'indexe pas (ne référence pas) mon backoffice, mon js, mon style...

A déposer à la racine de votre projet.

## 2 - Créer un fichier sitemap.xml

C'est un fichier xml, qui permet de faire un plan de tous les urls de votre site, pour faciliter l'indexation (le référencement) de votre site par Google. Google en a besoin pour comprendre et pouvoir analyser plus facilement votre site. C'est un plan, l'architecture de votre site. On en a besoin si on veut demander à Google de commencer tout de suite le référencement de notre site. Sans ça, le référencement peut prendre des mois.

A déposer à la racine de votre projet.

=> <https://www.xml-sitemaps.com/>

## 3 - Configurer un fichier .htaccess

C'est un fichier de configuration serveur, il permet par exemple de gérer des redirection, de configurer les urls etc... C'est lui qu'on utilise quand on veut par exemple que si on tape <http://www.academiews.fr> ça redirige vers <https://www.academiews.fr>.

A déposer à la racine de votre projet.

=> composer require symfony/apache-pack

## 4 - Configurer un script pour la mise en production

Cette partie concerne l'industrialisation du code, c'est à dire toute la configuration liée aux tests, à la mise en production de votre projet.

### 1 - Se connecter sur le serveur en ssh

user@server\_ssh

ex : ssh [djbpkms@ssh03.cluster43.aaa.hosting.ovh.fr](mailto:djbpkms@ssh03.cluster43.aaa.hosting.ovh.fr)

### 2 - Installer composer sur votre serveur

=> <https://getcomposer.org/download/>

### 3 - Installer npm

```
curl -fsSL https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.3/install.sh | bash
source ~/.bashrc
command -v nvm
nvm install 18
nvm use 18
node -v
npm -v
```

### 4 - Configurer une clef rsa

```
ssh-keyscan -t rsa github.com >> ~/.ssh/known_hosts
=> permet d'autoriser GitHub sur votre serveur
```

## MISE EN PRODUCTION

```
ssh-keygen -t rsa -b 4096 -C « web.start.contact@gmail.com »
```

=> génère une clef rsa qu'il faudra connecter à Github

```
eval "$(ssh-agent -s)" ssh-add ~/.ssh/id_rsa
```

=> Active l'agent SSH et ajoute ta clé

```
ssh-add -l
```

=> Vérifie que la clé a bien été ajoutée

```
cat ~/.ssh/id_rsa.pub
```

=> Affiche ta clef publique

Va sur GitHub → SSH Keys

Clique sur «**Settings => deploy keys**»

**Colle la clé** et donne-lui un nom (ex: "Serveur OVH")

**Clique sur "Add SSH key"**

### 3 - Créer le dossier dans lequel votre projet sera déposé

### 4 - Créer le script **deploy.sh** qui exécutera les commandes suivantes :

- git pull (git checkout si première fois que tu importes le projet)
- composer update
- npm run build
- php bin/console doctrine:migrations:migrate (fera un database:create si la bdd

n'existe pas)

Pour exécuter le script (A faire une fois) :

chmod +x deploy.sh : donner les droits d'exécution sur notre fichier  
./deploy.sh : exécuter le script

La commande à lancer :

```
ssh djbpkms@ssh03.cluster43.aaa.hosting.ovh.fr "cd /test && ./deploy.sh"
```

### 5 - Lancer l'indexation de votre site dans la Google Search Console

=> <https://search.google.com/search-console/>

### Aide à la console :

mkdir <dossier> : créer un dossier  
rm <fichier> : supprimer un fichier  
touch <nom\_fichier> : créer un fichier  
vim <nom\_fichier> : modifier un fichier  
esc + : + x : sauvegarder et quitter le fichier  
i : modifier le fichier