
Fiche de procédure d'installation et de déploiement d'une application web

Introduction :

Django est un Framework qui va nous permettre de lancer rapidement notre application web. Il inclut un serveur de développement qui va nous permettre de tester notre code localement mais pour tout ce qui est lié à cette SAE, un serveur web. Dans cette fiche de procédure, les éléments comment l'installation et configuration Django dans un environnement virtuel Python seront citées. Nous allons configurer un serveur Apache qui va héberger notre projet Django.

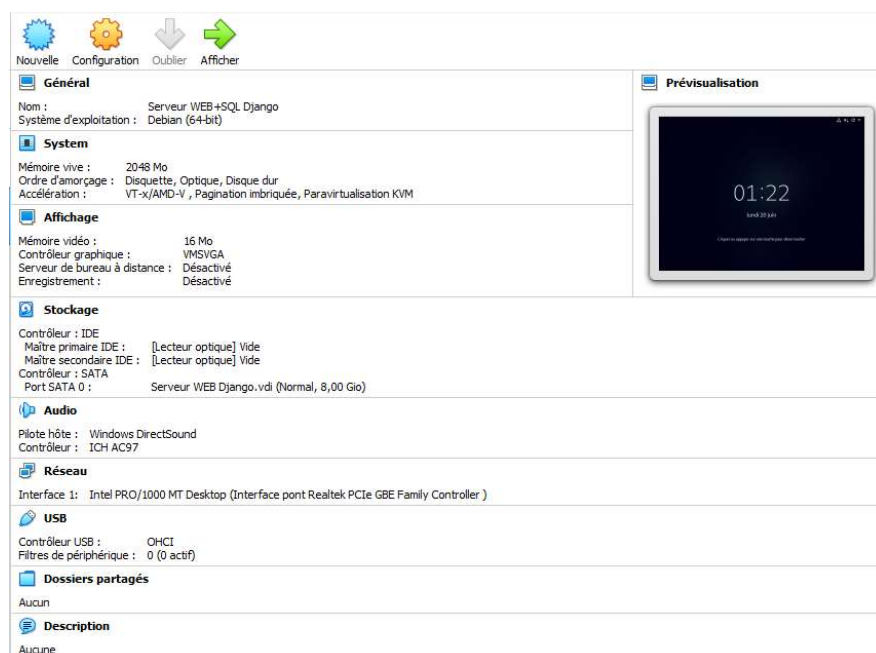
Prérequis :

VM : Debian (64 bits)

Paquets : python3-pip, apache2, libapache2, mod-wsgi-py3

Installation de la VM

Voici la configuration de la machine virtuelle :



Lorsque la configuration et la création de la VM a bien été effectué, nous pouvons lancer la VM.

Installation des paquets :

1^{ère} étape : Se mettre en Super Utilisateur

```
toto@toto:~$ su -  
Mot de passe :  
root@toto:~# █
```

Commande : su -

2^{ème} étape : Installation des paquets

```
root@toto:~# apt-get install python3-pip apache2 libapache2-mod-wsgi-py3  
Lecture des listes de paquets... Fait  
Construction de l'arbre des dépendances... Fait  
Lecture des informations d'état... Fait  
apache2 est déjà la version la plus récente (2.4.53-1~deb11u1).  
libapache2-mod-wsgi-py3 est déjà la version la plus récente (4.7.1-3+b1).  
python3-pip est déjà la version la plus récente (20.3.4-4+deb11u1).  
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.  
root@toto:~#
```

Configuration d'un environnement virtuel Python :

1^{ère} étape : Installation du paquet « virtualenv » afin de créer des environnements virtuels :

```
root@toto:/home/toto# pip3 install virtualenv  
Requirement already satisfied: virtualenv in /usr/local/lib/python3.9/dist-packages (20.14.1)  
Requirement already satisfied: filelock<4,>=3.2 in /usr/local/lib/python3.9/dist-packages (from virtualenv) (3.7.1)  
Requirement already satisfied: six<2,>=1.9.0 in /usr/lib/python3/dist-packages (from virtualenv) (1.16.0)  
Requirement already satisfied: distlib<1,>=0.3.1 in /usr/local/lib/python3.9/dist-packages (from virtualenv) (0.3.4)  
Requirement already satisfied: platformdirs<3,>=2 in /usr/local/lib/python3.9/dist-packages (from virtualenv) (2.5.2)  
root@toto:/home/toto# █
```

2^{ème} étape : Création d'un répertoire dans lequel nous allons mettre notre projet :

```
root@toto:/home/toto# mkdir django-app █
```

3^{ème} étape : On se rend dans le répertoire du projet et on crée un environnement virtuel nommé « env » :

```
root@toto:/home/toto/django-app# virtualenv env
```

4^{ème} étape : On active l'environnement virtuel afin d'installer les exigences Python du projet

```
root@toto:/home/toto# source django-app/env/bin/activate
(env) root@toto:/home/toto#
```

5^{ème} étape : Installation de Django

```
(env) root@toto:/home/toto# pip install Django
Requirement already satisfied: Django in ./django-app/env/lib/python3.9/site-packages (4.0.5)
Requirement already satisfied: sqlparse>=0.2.2 in ./django-app/env/lib/python3.9/site-packages (from Django) (0.4.2)
Requirement already satisfied: asgiref<4,>=3.4.1 in ./django-app/env/lib/python3.9/site-packages (from Django) (3.5.2)
WARNING: You are using pip version 22.0.4; however, version 22.1.2 is available.
You should consider upgrading via the '/home/toto/django-app/env/bin/python -m pip install --upgrade pip' command.
(env) root@toto:/home/toto#
```

Configuration du projet Django :

1^{ère} étape : On clone le projet Django dans le répertoire créé précédemment

```
(env) root@toto:/home/toto/django-app# git clone "url"█
```

2^{ème} étape : J'ajuste les paramètres du projet Django (settings.py) :

```
(env) root@toto:/home/toto/django-app# nano ludotheque/settings.py
ALLOWED_HOSTS = ['*']
STATIC_URL = '/static/'
STATIC_ROOT = os.path.join(BASE_DIR, 'static/')
```

3^{ème} étape : Création d'un utilisateur administratif pour le projet :

```
(env) root@toto:/home/toto/django-app# ./manage.py createsuperuser
```

4^{ème} étape : Pare-feu

```
(env) root@toto:/home/toto/django-app# ufw allow 8000█
```

5^{ème} étape : IPTABLES

```
(env) root@toto:/home/toto/django-app# iptables -I INPUT -p tcp --dport 8000 -j ACCEPT
```

Teste du projet :

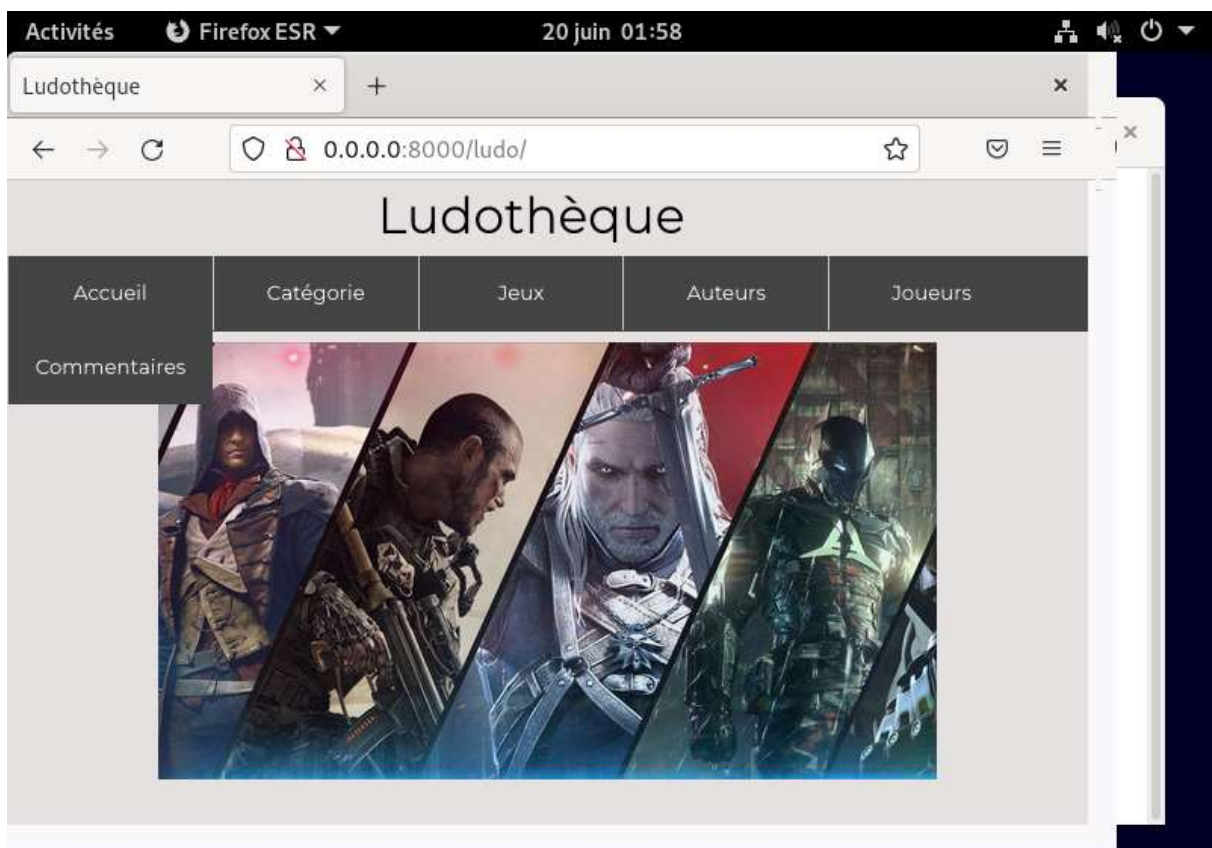
On peut tester le projet en lançant le serveur de développement Django avec cette commande :

```
(env) root@toto:/home/toto/django-app# ./manage.py runserver 0.0.0.0:8000
Watching for file changes with StatReloader
Performing system checks...
```

```
System check identified no issues (0 silenced).
June 19, 2022 - 23:58:10
Django version 4.0.5, using settings 'ludotheque.settings'
Starting development server at http://0.0.0.0:8000/
Quit the server with CONTROL-C.
```

Dans un navigateur web, je copie-coller l'ip « 0.0.0.0 :8000 ».

Affichage :



Cela fonctionne !

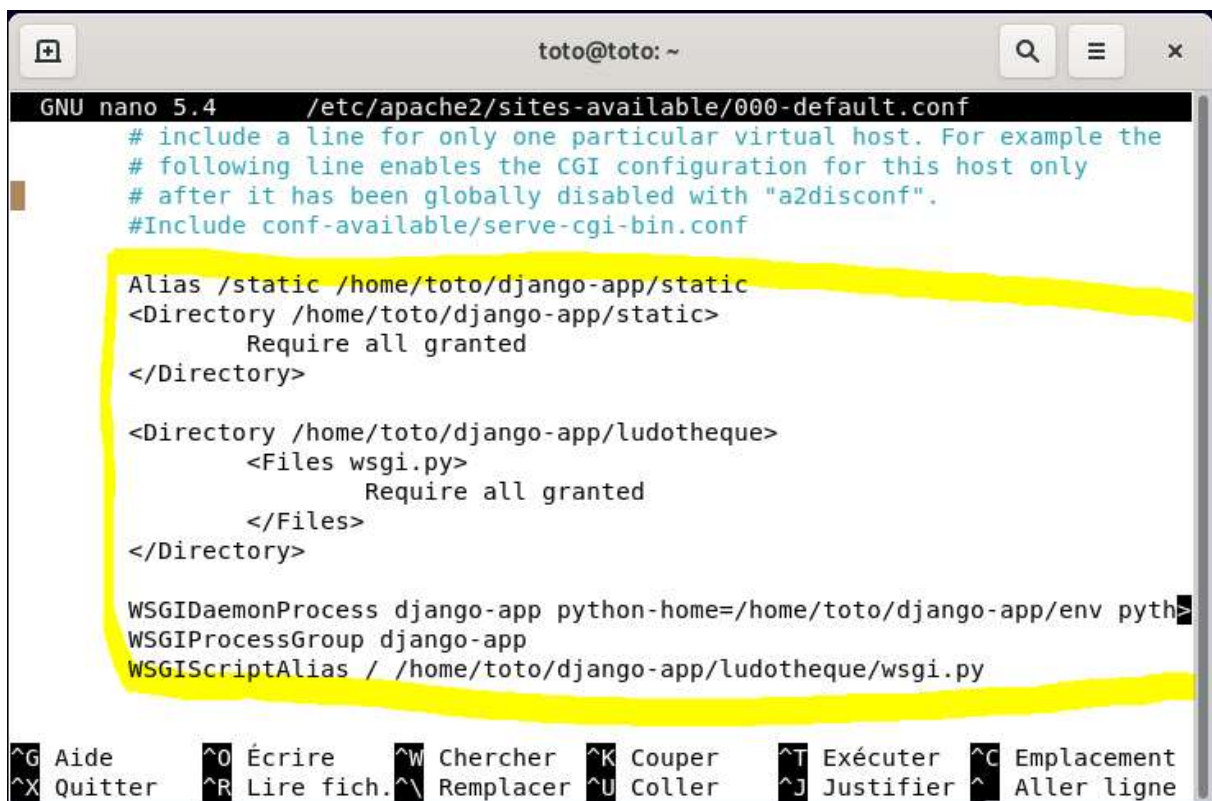
Configuration de Apache :

1^{ère} étape : Je configure le fichier d'hôte virtuel par défaut :

```
(env) root@toto:~# nano /etc/apache2/sites-available/000-default.conf
```

Mehdi REHM RT122

Je rajoute les lignes suivantes dans le fichier de configuration :



```
GNU nano 5.4 /etc/apache2/sites-available/000-default.conf
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf

Alias /static /home/toto/django-app/static
<Directory /home/toto/django-app/static>
    Require all granted
</Directory>

<Directory /home/toto/django-app/ludotheque>
    <Files wsgi.py>
        Require all granted
    </Files>
</Directory>

WSGIDaemonProcess django-app python-home=/home/toto/django-app/env python
WSGIProcessGroup django-app
WSGIScriptAlias / /home/toto/django-app/ludotheque/wsgi.py
```

Autorisations :

```
$ chmod 664 ~/myproject/db.sqlite3
$ chmod 775 ~/myproject
```

```
$ sudo ufw delete allow 8000
$ sudo ufw allow 'Apache Full'
```

```
$ sudo iptables -D INPUT -p tcp --dport 8000 -j ACCEPT
$ sudo iptables -I INPUT -p tcp --dport 80 -j ACCEPT
```

```
$ sudo systemctl restart apache2
```

Teste du projet avec Apache2 :

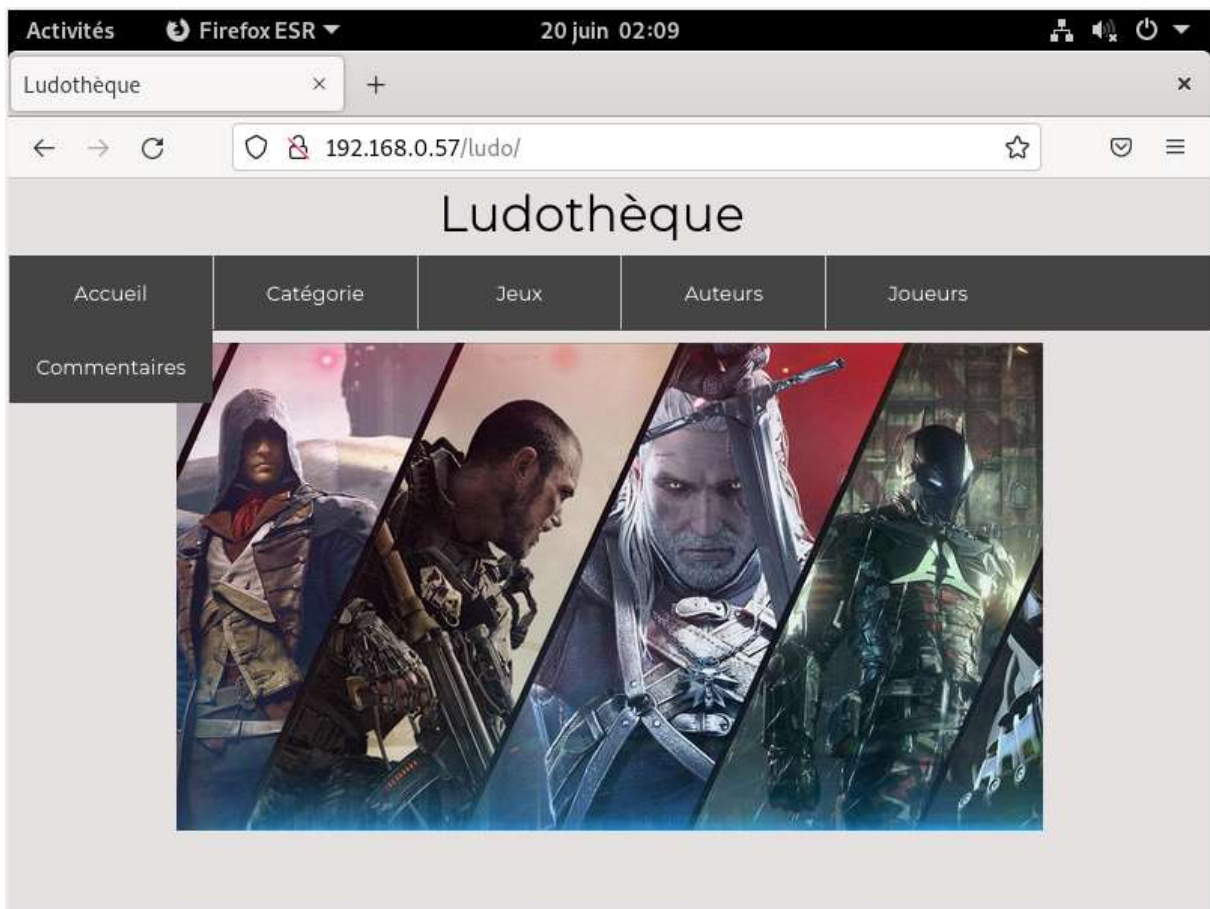
```
root@toto:~# systemctl start apache2
root@toto:~#
```


Mehdi REHM RT122

Je vérifie mon adressage IP :

```
root@toto:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:7e:f5:f4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.57/24 brd 192.168.0.255 scope global dynamic noprefixroute enp0s3
        valid_lft 39127sec preferred_lft 39127sec
    inet6 2a01:e0a:234:1170:69f2:b650:d5cd:d20a/64 scope global temporary dynamic
        valid_lft 86139sec preferred_lft 81849sec
    inet6 2a01:e0a:234:1170:a00:27ff:fe7e:f5f4/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 86139sec preferred_lft 86139sec
    inet6 fe80::a00:27ff:fe7e:f5f4/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
root@toto:~#
```

Je prends mon adressage IP et je la mets sur une page internet :



L'application web fonctionne avec apache2.