

Les fichiers de configuration des services sur nos VM

Sur la VM SERVEUR :

1. Mettre à jour le système : `sudo apt update`
2. Installations de Python, pip et venv : `sudo apt install python3 python3-pip python3-venv -y`
3. Pour phase de Production : installer Gunicorn et Nginx →
4. Création et activer l'environnement virtuel Python : `python3 -m venv .venv`
`source .venv/bin/activate`

Installer les dépendances Python : `pip install -r requirements.t`

1. Configurer les variables d'environnement (ex : `.env`)
Renseigner les accès à la base de données/Permet de créer un fichier statique pour le css:

Dans `settings.py` → `ALLOWED_HOST = ['169.254.6.45', 'localhost', '127.0.0.1']`

```
→ DATABASE = {  
    'default': {  
        'ENGINE': 'django.db.backends.mysql',  
        'NAME': 'adminserv',  
        'USER': 'toto',  
        'PASSWORD': 'toto',  
        'HOST': '169.254.8.48',  
        'PORT': '3306',  
    }  
}
```

→ `import os`

`STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')`

1. Appliquer les migrations Django : `python manage.py migrate`
2. Créer un superutilisateur Django : `python manage.py createsuperuser`

Collecter les fichiers statiques : `python manage.py collectstatic`

1. Démarrer le serveur via l'application pycharm : `python manage.py runserver 0.0.0.0:8000`

2. Démarrer le serveur via ssh :

`ssh vboxuser@169.254.6.45`

(*si pas de ssh :*

`sudo apt update`

`sudo apt install openssh-server`

`sudo systemctl enable ssh)`

`source .venv/bin/activate`

`cd ~/PycharmProjects/SAE23LM-master/monprojet`

`pip install -r requirements.txt (si pas déjà fait)`

`python manage.py runserver 0.0.0.0:8000`

Accéder à l'application via <http://169.254.6.45/> depuis un navigateur

1. Démarrer le serveur via ssh :

`ssh vboxuser@169.254.6.45`

(*si pas de ssh :*

`sudo apt update`

`sudo apt install openssh-server`

`sudo systemctl enable ssh)`

`source .venv/bin/activate`

`cd ~/PycharmProjects/SAE23LM-master/monprojet`

`pip install -r requirements.txt (si pas déjà fait)`

`sudo gunicorn -----`

Sur la VM MariaDB :

1. Installation de MariaDB :

```
sudo apt update
```

```
sudo apt install mariadb-server
```

2. Sécurisation de l'installation :

```
sudo mysql-secure-installation
```

3. Lancement et activation du service MariaDB :

```
sudo systemctl start mariadb
```

```
sudo systemctl enable mariadb
```

4. Connexion au client MariaDB :

```
sudo mysql -u root -p
```

5. Création de la base de données:

```
CREATE DATABASE adminserv CHARACTER SET utf8mb4;
```

6. Création de l'utilisateur :

```
CREATE USER 'toto'@'localhost' IDENTIFIED BY 'toto';
```

7. Donner tous les droits à l'utilisateur 'toto' :

```
GRANT ALL PRIVILEGES ON adminserv.* TO 'toto'@'localhost';
```

```
FLUSH PRIVILEGES;
```

8. Connection à la base de données :

```
USE adminserv;
```

9. Création des 6 CRUDS MYSQL :

```
CREATE TABLE TypeServeurs (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    type VARCHAR(100) NOT NULL,  
    description_serveurs TEXT NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE Serveurs (  
    id INT AUTO_INCREMENT PRIMARY KEY,
```

```
nom_serveurs VARCHAR(100) NOT NULL DEFAULT 'serveur_test',
type_serveur_id INT NOT NULL,
nombre_processeur_serveurs INT NOT NULL DEFAULT 100,
capacite_memoire_serveurs INT NOT NULL DEFAULT 100,
capacite_stockage_serveurs INT NOT NULL DEFAULT 100,
FOREIGN KEY (type_serveur_id) REFERENCES TypeServeurs(id) ON DELETE
CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE Services (
id INT AUTO_INCREMENT PRIMARY KEY,
nom_services VARCHAR(100) NOT NULL,
date_lancement DATE NOT NULL,
espace_memoire_utilise INT NOT NULL,
memoire_vive_necessaire INT NOT NULL,
serveur_lancement_id INT NOT NULL,
FOREIGN KEY (serveur_lancement_id) REFERENCES Serveurs(id) ON DELETE
CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE Utilisateurs (
id INT AUTO_INCREMENT PRIMARY KEY,
nom_utilisateurs VARCHAR(100) NOT NULL,
prenom_utilisateurs VARCHAR(100) NOT NULL,
email_utilisateurs VARCHAR(254) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE Applications (
id INT AUTO_INCREMENT PRIMARY KEY,
nom_applications VARCHAR(100) NOT NULL,
```

```
logo_applications VARCHAR(100), -- Chemin du fichier image, nullable
utilisateur_applications_id INT NOT NULL,
memoire_utilisee INT NOT NULL DEFAULT 0,
cpu_utilise INT NOT NULL DEFAULT 1,
    FOREIGN KEY (utilisateur_applications_id) REFERENCES Utilisateurs(id) ON
DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE UsageRessource (
    id INT AUTO_INCREMENT PRIMARY KEY,
    applications_id INT NOT NULL,
    services_id INT NOT NULL,
    type_ressource VARCHAR(50) NOT NULL,
    FOREIGN KEY (applications_id) REFERENCES Applications(id) ON DELETE
CASCADE,
    FOREIGN KEY (services_id) REFERENCES Services(id) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```