Legende :

EXEMPLE

PREREQUIS

COMMANDE A EXECUTER POUR L'ACTION

SAMPLE DE CONFIG

COMMENTAIRE

## PROCEDURE DEPLOIEMENT DJANGO

Cette procédure est à réaliser (de préférence) sur une machine de type 'ubuntu-server22,04' Si le type d'OS varie, certaines action peuvent être susceptibles de varier.

Ordre de procédure	Description de l'action à réaliser	Aides contextuelles & Commandes à entrer dans le prompt
1	Télécharger la suite Docker & l'image souhaitée	-> sudo apt install docker -> sudo docker pull ubuntu
2	Créer le 'conteneur principal' qui contiendra les deux machines (SQL & DJANGO), cette étape peut être passée si le service doit être installé à même la machine (déconseillé).	-> docker run -p <port-entrant>:<port-sortant> -td -itname <nom_conteneur> <image_utilisée> -&gt; docker run -p 8000:8000 -p 3306:3306 -tdname CONTENEUR_GLOBAL ubuntu *Rediriger les ports de votre routeur de X vers 3306 pour SQL et Y vers 8000 pour django.* *Il est également possible de changer directement le port entrant du conteneur pour utiliser d'autres flux. * *Pour ajouter de la sécurité, il est intéressant d'installer un conteneur de même niveau que le conteneur global et de lui assigner un role de pare-feu.*</image_utilisée></nom_conteneur></port-sortant></port-entrant>
3	A l'intérieur du conteneur global, créer un conteneur uniquement prévu pour le service SQL.	*rejoindre le conteneur global : sudo docker exec -it CONTENEUR_GLOBAL* *télécharger l'image docker Debian : sudo docker pull debian* -> docker run -p 3306:3306 -tdname CONTENEUR_SQL debian
4	A l'intérieur du conteneur global, créer un second conteneur pour le service de déploiement django (ou Gunicorn/nginx en alternative).	-> docker run -p 8000:8000 -tdname CONTENEUR_DJANGO debian
5	Déplacer le dossier du projet sur le conteneur django	-> sudo docker cp URI_SOURCE:DESTINATION_URI
6	Installer les services mysql sur le conteneur SQL	* se rendre sur le conteneur sql ou envoyer la commande en target * -> sudo apt install mysql-server
7	Ajuster la config du serveur SQL	-> sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf *modifier les sections requise en fonction de votre installation* : [mysqld] # * Basic Settings
		# user = <utilisateur créé="" le="" mysql="" serveur="" sur=""> pid-file = /var/run/mysqld/mysqld.pid #défaut socket = /var/run/mysqld/mysqld.sock #défaut port = 3306 #défaut datadir = /var/lib/mysql #possibilité de dl notre bdd via github pour essayer sans créer un premier utilisateur a la main bind-address = <lp conteneur="" de="" désservissement="" prévu="" votre=""> #défaut : 127.0.0.1</lp></utilisateur>
8	Redémarrer le service SQL	-> sudo apt install systemctl -> sudo systemctl restart mysql.service -> sudo systemctl status mysql.service *vérifier que la config est bonne en essayant d'interagir avec le serveur SQL en dehors du conteneur*
	Dans le conteneur django, sur le fichier Settings.py, vérifier la configuration de la base de données pour le projet par rapport à votre installation	* rejoindre le conteneur django * -> cd /GESTION_ABSENCES -> sudo nano settings.py
9	Exemple de fichier de config avec une base SQL connectée	DATABASES = {     'default':{         'ENGINE': 'django.db.backends.mysql',         'NAME': ' <nom_de_la_bdd>', #defaul : DBABSENCES     'USER': '<nom mysql'="" utilisateur="">,         'PASSWORD': 'Mdp de l'utilisateur mysql&gt;',         'HOST': '<adresse conteneur="" ip="" sql'="">,         'PORT: '3306',         'OPTIONS': {             'autocommit': True,             },         } }</adresse></nom></nom_de_la_bdd>
x	*A ce stade, le serveur peut être démarrer et accessib *Pour ce faire	le depuis n'importe ou sur le net si la base de donnée fourni sur github à bien été déplacé dans le fichier de BDD de votre conteneur (/var/lib/mysql)* e, passer directement à la procédure No 13 puis reprenez à la procédure 16* *sinon, continuez avec la procédure No 10*

10	Setup l'utilisateur SQL spécifié dans le settings.py de django et lui accorder les droits à la BDD	*se connecter en root à un terminal sq!* -> sudo mysql -u root -p sql -> CREATE USER 'Nom_utilisateur' PASSWORD 'Mdp_utilisateur'; sql -> GRANT ALL PRIVILEGES ON DBABSENCES.* TO 'Nom_utilisateur';
11	Effectuer les migrations du projets sur le conteneur mysql	*Depuis le conteneur django* -> cd GESTION_ABSENCES/ -> python3 manage.py makemigrations ABSENCES -> python3 manage.py migrate *si des problèmes apparaissent, voir les logs*
12	Vérification de la création des tables depuis le terminal SQL	sql -> use DBABSENCES; sql -> SHOW TABLES;
13	Démarrage de la Checklist Django pour le déploiement	*Sur le conteneur Django* → cd GESTION_ABSENCES; *Aller sur l'url du site ci-dessous et respecter la procédure à réaliser dans le settings.py du projet* <u>https://docs.djangoproject.com/en/4.2/howto/deployment/checklist/</u>
14	Création d'un admin pour la gestion du site	* Sur le conteneur django * -> python3 manage.py createsuperuser -> admin -> admin@mail.com -> toto -> toto -> toto
15	Création d'un object 'enseignant' pour pouvoir ensuite naviguer sur le serveur	*sur le navigateur : * http:/ <adresse_ip_conteneur_global>:<port_séléctionné>/admin * se connecter * Username : admin Password : toto *Il ne reste maintenantt plus qu'a, via l'interface admin, ajouter deux groupes : 'ENSEIGNANTS' &amp; 'ETUDIANTS'* *Ajouter un nouvel utilisateur 'ENSEIGNANTS'* mail : ens@mail.com Mdp : ens</port_séléctionné></adresse_ip_conteneur_global>
16	Terminer le déploiement du projet !	* la checklist doit être complète et cohérente (voir procédure No 13) * *depuis le conteneur Django* -> python3 manage.py checkdeploy SOLUTION SERV UNICORN TIERS : -> python -m pip install uvicorn gunicorn -> python -m gunicorn <nom_projet>.asgi:application -k uvicorn.workers.UvicornWorker SOLUTION DJANGO NATIF : -&gt; python3 manage.py runserver <adresse_ip_conteneur>:8000</adresse_ip_conteneur></nom_projet>
17	[OPTIONNEL] Ajouter un nom de domaine	SOLUTION 1 : Activer un service DNS sur votre serveur et l'associer à la route vers vos conteneurs afin de ne plus avoir à taper une IP dans le navigateur SOLUTION 2 : Via votre routeur (si disponible) utiliser l'interface DynDNS afin de récuperer un nom de domaine et un accès propre à votre serveur denuis le net
X	X	X
X X	X X	X X X
X X X	X X X X	X X X X
X X X X	X X X X X	X X X X X X
X X X X X	X X X X X X	X X X X X X X X
X X X X X X	X X X X X X X X	X X X X X X X X X X X X X X
X X X X X X X	X X X X X X X X X	X X X X X X X X X X X X X X
X X X X X X X	X X X X X X X X X	X X X X X X X X X X X X X X
X X X X X X X X X	X X X X X X X X X X	X X X X X X X X X X X X X X
X X X X X X X X X X	X X X X X X X X X X X X X	X       X
X X X X X X X X X X X	X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X	X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X         X <td< th=""></td<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>
X X X X X X X X X X X X X X X X X X X	X X X X X X X X X X X X X X X X X X X	X       X <t< th=""></t<>