

Projet Docker - Application POZOS

Contexte

la tâche est de conteneuriser l'application web et API de l'entreprise Pozos afin de faciliter le déploiement et la maintenance de l'infrastructure.

Objectifs

- Construire l'image Docker de l'API POZOS en respectant les spécifications fournies
- Créer un fichier docker-compose.yml pour déployer l'application complète
- Mettre en place un registre Docker privé pour stocker les images

Étapes du projet

1. Construction de l'image Docker de l'API

- Utilisation de l'image de base "python:2.7-buster"
- Spécification du mainteneur de l'image
- Copie du code source de l'API dans le conteneur à la racine
- Installation des dépendances Flask, Flask-HTTPAuth, Flask-SimpleLDAP et python-dotenv
- Création d'un volume pour stocker le fichier "student_age.json"
- Exposition du port 5000 pour l'API
- Définition du point d'entrée pour lancer le script "student_age.py"

2. Déploiement avec Docker Compose

- Création du fichier docker-compose.yml
- Déploiement de deux services :

- "website" basé sur l'image php:apache, avec variables d'environnement, montage du répertoire web et dépendance au service API
- "API" utilisant l'image construite précédemment, avec montage du fichier "student_age.json"

3. Registre Docker privé

- Déploiement d'un registre Docker privé (ou utilisation de Portus)
- Envoi de l'image de l'API sur le registre privé
- Visualisation des images dans l'interface web du registre

Résultats

- L'application POZOS est désormais conteneurisée et facile à déployer grâce à Docker Compose
- Les images Docker sont stockées dans un registre privé sécurisé
- L'ensemble de l'infrastructure est modulaire, évolutive et portable

Compétences acquises

- Maîtrise de la construction d'images Docker personnalisées
- Utilisation avancée de Docker Compose pour le déploiement d'applications multi-services
- Mise en place d'un registre Docker privé pour la gestion des images

[illegible]

```
0.1
nzapa@nzapa-HP-EliteBook-840-G3:~/projet/easytraining/mini-projet-docker/student-list$ sudo docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
pozos-api	0.1	c272d8396682	6 minutes ago	1.05GB
localhost:5000/pozos-api	0.1	c272d8396682	6 minutes ago	1.05GB

```
nzapa@nzapa-HP-EliteBook-840-G3:~/projet/easytraining/mini-projet-docker/student-list$ sudo docker login localhost:5000
Username: pozos
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
nzapa@nzapa-HP-EliteBook-840-G3:~/projet/easytraining/mini-projet-docker/student-list$ sudo docker push localhost:5000/pozos-api:0.1
The push refers to repository [localhost:5000/pozos-api]
f5cd5450b2c5: Pushed
7c6ccec564ff: Pushed
ff1f999b787a: Pushed
6e2f63b0cf05: Pushed
45359261cd7a: Pushed
ad312497d9a5: Pushed
474c7af10697: Pushed
dcc1cfeee1ab: Pushed
eccb9ed74974: Pushed
53d40515380c: Pushed
6af7a54a0a0d: Pushed
0.1: digest: sha256:abdaece08d87ecba567208da782a1f5c764fb7d9c4a5638bcf2ed145b0e55bb0 size: 2637
nzapa@nzapa-HP-EliteBook-840-G3:~/projet/easytraining/mini-projet-docker/student-list$
```



