

# Rapport – Docker

Auteur : NGHIEM Trang

Email : [tthutrang.nghiem@gmail.com](mailto:tthutrang.nghiem@gmail.com)

## 1. Demande :

Le but de ce projet est de créer un pipeline CI/CD pour tester une API de façon automatique, avant déploiement.

Les trois points d'entrée de l'API à tester sont :

- */permissions* : renvoie les permissions d'un utilisateur
- */v1/sentiment* : renvoie l'analyse de sentiment en utilisant un vieux modèle
- */v2/sentiment* : renvoie l'analyse de sentiment en utilisant un nouveau modèle

Les trois tests suivant seront effectués :

- **Authentication** : sur le point d'entrée */permissions*. Les inputs sont *username* et *password*. Les résultats sortis indiquent si l'utilisateur est permis/non-permis d'utiliser l'API.
- **Authorization** : sur les points d'entrées */v1/sentiment* et */v2/sentiment*. Les inputs sont *username*, *password* et *sentence*. Les inputs des 3 utilisateurs sont proposés pour différents scénarios.
- **Content** : sur le point d'entrée */v1/sentiment* et */v2/sentiment* pour un seul utilisateur en proposant 2 phrases différentes (*sentence*).

Afin d'automatiser ces tests, il faut utiliser docker-compose.

## 2. Réalisation :

Afin de clarifier les étapes à effectuer, ce rapport est accompagné par les fichiers suivants :

- `docker-compose.yml`
- 3 fichiers Python utilisés dans les images Docker : `test_authentication.py`, `test_authorization.py`, `test_content.py`.
- 3 Dockerfile utilisés pour construire les images : `Dockerfile_authentication`, `Dockerfile_authorization`, `Dockerfile_content`
- Fichier des commandes utilisées : `setup.sh`
- Fichier des résultats : `log.txt`

Dans un dossier, il faut ajouter tous les fichiers `docker-compose.yml`, les 3 fichiers Python, les 3 fichiers Dockerfile et `setup.sh`. Ensuite, nous allons créer 3 dossiers suivants : `test_authentication`, `test_authorization`, `test_content`.

### 3. Résultats

Une fois lancer docker-compose par la commande ./setup.sh, les étapes sont affichées sur le terminal comme suit :

```
ubuntu@ip-172-31-6-211: ~/docker_pratique
ubuntu@ip-172-31-6-211:~/docker_pratique$ vi setup.sh
ubuntu@ip-172-31-6-211:~/docker_pratique$ vi setup.sh
ubuntu@ip-172-31-6-211:~/docker_pratique$ ./setup.sh
building authentication docker image
Sending build context to Docker daemon 21.5kB
Step 1/5 : FROM debian:latest
--> 48c562c706d6
Step 2/5 : RUN apt-get update && apt-get install python3-pip -y && pip install requests
--> Using cache
--> dc64ce4076a5
Step 3/5 : ADD test_authentication.py /test_authentication/test_authentication.py
--> 3d7fd1a1e540
Step 4/5 : WORKDIR /test_authentication/
--> Running in f7455e90e54c
Removing intermediate container f7455e90e54c
--> e357e8a501ec
Step 5/5 : CMD python3 test_authentication.py
--> Running in b11e97868f86
Removing intermediate container b11e97868f86
--> f4c0e7a794c9
Successfully built f4c0e7a794c9
Successfully tagged test_authentication:latest
building authorization docker image
Sending build context to Docker daemon 21.5kB
Step 1/5 : FROM debian:latest
--> 48c562c706d6
Step 2/5 : RUN apt-get update && apt-get install python3-pip -y && pip install requests
--> Using cache
--> dc64ce4076a5
Step 3/5 : ADD test_authorization.py /test_authorization/test_authorization.py
--> 08a62329d1cd
Step 4/5 : WORKDIR /test_authorization/
--> Running in 018305b84bbd
Removing intermediate container 018305b84bbd
--> 81c71fec94c4
Step 5/5 : CMD python3 test_authorization.py
--> Running in c019f6055f5e
Removing intermediate container c019f6055f5e
--> 71bc23bc8bc2
Successfully built 71bc23bc8bc2
Successfully tagged test_authorization:latest
building content docker image
Sending build context to Docker daemon 21.5kB
Step 1/5 : FROM debian:latest
--> 48c562c706d6
Step 2/5 : RUN apt-get update && apt-get install python3-pip -y && pip install requests
--> Using cache
--> dc64ce4076a5
Step 3/5 : ADD test_content.py /test_content/test_content.py
--> 3db37e8dc38b
Step 4/5 : WORKDIR /test_content/
--> Running in 70209bffb1f1
```

ubuntu@ip-172-31-6-211: ~/docker\_pratique

```
fastapi INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
fastapi INFO: 172.28.1.4:48068 - "GET /v1/sentiment?username=alice&password=wonderland&sentence=life+is+beautiful HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.4:48070 - "GET /v2/sentiment?username=alice&password=wonderland&sentence=life+is+beautiful HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.4:48072 - "GET /v1/sentiment?username=alice&password=wonderland&sentence=that+sucks HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.4:48074 - "GET /v2/sentiment?username=alice&password=wonderland&sentence=that+sucks HTTP/1.1" 200 OK
test_content username : alice
test_content {'username': 'alice', 'version': 'v1', 'sentence': 'life is beautiful', 'score': 0.5994}
test_content username : alice
test_content {'username': 'alice', 'version': 'v2', 'sentence': 'life is beautiful', 'score': 0.5994}
test_content username : alice
test_content {'username': 'alice', 'version': 'v1', 'sentence': 'that sucks', 'score': -0.3612}
test_content username : alice
test_content {'username': 'alice', 'version': 'v2', 'sentence': 'that sucks', 'score': -0.3612}
fastapi INFO: 172.28.1.2:48604 - "GET /permissions?username=alice&password=wonderland HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.2:48606 - "GET /permissions?username=bob&password=builder HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.2:48608 - "GET /permissions?username=clementine&password=mandarine HTTP/1.1" 403 Forbidden
test_authentication username : alice
test_authentication {'username': 'alice', 'permissions': ['v1', 'v2']}
test_authentication 200 : SUCCESS
test_authentication username : bob
test_authentication {'username': 'bob', 'permissions': ['v1']}
test_authentication 200 : SUCCESS
test_authentication username : clementine
test_authentication {'detail': 'Authentication failed'}
test_authentication 403 : FAILURE
fastapi INFO: 172.28.1.3:47944 - "GET /v1/sentiment?username=alice&password=wonderland&sentence=life+is+beautiful HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.3:47946 - "GET /v2/sentiment?username=alice&password=wonderland&sentence=life+is+beautiful HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.3:47948 - "GET /v1/sentiment?username=bob&password=builder&sentence=life+is+beautiful HTTP/1.1" 200 OK
fastapi INFO: 172.28.1.3:47950 - "GET /v2/sentiment?username=bob&password=builder&sentence=life+is+beautiful HTTP/1.1" 403 Forbidden
fastapi INFO: 172.28.1.3:47952 - "GET /v1/sentiment?username=clementine&password=mandarine&sentence=life+is+beautiful HTTP/1.1" 403 Forbidden
fastapi INFO: 172.28.1.3:47954 - "GET /v2/sentiment?username=clementine&password=mandarine&sentence=life+is+beautiful HTTP/1.1" 403 Forbidden
test_authorization username : alice
test_authorization {'username': 'alice', 'version': 'v1', 'sentence': 'life is beautiful', 'score': 0.5994}
test_authorization 200 : SUCCESS
test_authorization username : alice
test_authorization {'username': 'alice', 'version': 'v2', 'sentence': 'life is beautiful', 'score': 0.5994}
test_authorization 200 : SUCCESS
test_authorization username : bob
test_authorization {'username': 'bob', 'version': 'v1', 'sentence': 'life is beautiful', 'score': 0.5994}
test_authorization 200 : SUCCESS
test_authorization username : bob
test_authorization {'detail': 'This service is not included in your plan.'}
test_authorization 403 : FAILURE
test_authorization username : clementine
test_authorization {'detail': 'Authentication failed'}
test_authorization 403 : FAILURE
test_authorization username : clementine
test_authorization {'detail': 'Authentication failed'}
test_authorization 403 : FAILURE
test_authentication exited with code 0
test_authorization exited with code 0
test_content exited with code 0
```

Les résultats sont enregistrés dans le fichier log.txt. Au lieu de renvoyer les scores  $-I$  et  $I$ , la version 1 donne les mêmes scores que la version 2. Cette version 1 doit être corrigée par les développeurs.

```
ca. ubuntu@ip-172-31-6-211: ~/docker_pratique
```

```
200 : SUCCESS
```

```
Test d'autorization http://172.28.1.1:8000/v2/sentiment
bob
{'detail': 'This service is not included in your plan.'}
403 : FAILURE
```

```
Test d'autorization http://172.28.1.1:8000/v1/sentiment
clementine
{'detail': 'Authentication failed'}
403 : FAILURE
```

```
Test d'autorization http://172.28.1.1:8000/v2/sentiment
clementine
{'detail': 'Authentication failed'}
403 : FAILURE
```

```
Test de l'analyse du contenu : http://172.28.1.1:8000/v1/sentiment
alice
life is beautiful
0.5994
```

```
Test de l'analyse du contenu : http://172.28.1.1:8000/v2/sentiment
alice
life is beautiful
0.5994
```

```
Test de l'analyse du contenu : http://172.28.1.1:8000/v1/sentiment
alice
that sucks
-0.3612
```

```
Test de l'analyse du contenu : http://172.28.1.1:8000/v2/sentiment
alice
that sucks
-0.3612
```

```
Test authentication :http://172.28.1.1:8000/permissions
alice
{'username': 'alice', 'permissions': ['v1', 'v2']}
200 : SUCCESS
```

```
Test authentication :http://172.28.1.1:8000/permissions
bob
{'username': 'bob', 'permissions': ['v1']}
200 : SUCCESS
```

```
Test authentication :http://172.28.1.1:8000/permissions
clementine
{'detail': 'Authentication failed'}
403 : FAILURE
```