

TD5 – Docker

Dans ce TD nous allons nous intéresser au processus permettant de faciliter le déploiement et l'installation de releases. Pour ce faire, il nous faudra dans un premier temps commencer par créer un compte Docker en se rendant sur le lien suivant :

- <https://hub.docker.com/signup>

Malheureusement, Docker ne fonctionne pas sur les machines du CREMI, nous allons donc utiliser une plateforme d'expérimentation de Docker en ligne. Une fois votre compte Docker crée, rendez-vous donc sur le lien suivant, cliquez sur Login puis Docker :

- <https://labs.play-with-docker.com/>

Par la suite, il vous faudra créer une nouvelle instance en cliquant sur « + ADD NEW INSTANCE », ce qui vous affichera un terminal vers un environnement virtuel. Tapez la commande « `docker -v` » pour confirmer que Docker est bien installé sur votre environnement virtuel.

Astuce : Si vous souhaitez uploader des fichiers dans votre environnement virtuel, il suffit de faire un glisser-déposer de votre fichier depuis votre gestionnaire de fichier vers le terminal affiché sur la page web.

Q1: Lancez la commande suivante sur votre environnement virtuel : « `docker pull oumaziz/m2-td5-q1` », une fois le téléchargement terminé lancez la commande « `docker images` » pour voir la liste des images présentes sur l'environnement virtuel.

Q2: Lancez le container à partir de l'image que vous avez téléchargé en faisant :

« `docker run -p 3000:3000 oumaziz/m2-td5-q1` »

Rendez-vous sur votre navigateur à l'url « `http://localhost:3000` », que se passe-t-il ?

Q3: Clonez le répertoire « <https://github.com/oumaziz/M2-TD5-workshop> » sur votre ordinateur puis complétez le Dockerfile afin qu'il puisse cloner le dépôt de votre rendu du TD4. Transférez votre Dockerfile modifié sur votre environnement virtuel en faisant un glisser-déposer sur le terminal dans la page web. Enfin, buildez ce Dockerfile afin d'obtenir une image en utilisant la commande suivante :

« `docker build . -t <username_dockerhub>/<nom_repo>` »

Q4: Vérifiez que votre image a bien été construite en faisant « `docker images` » puis lancez là en faisant « `docker run -d -p 3000:3000 <username_dockerhub>/<nom_repo>` ». Cliquez sur le bouton « 3000 » en haut de la page, que constatez-vous ?

Q5: Le container étant entrain de tourner en tâche de fond (detached), tapez la commande « `docker ps` » pour voir les containers en cours d'exécution. Terminez l'exécution du container en faisant la commande « `docker stop <premiers caractères du nom du container>` ».

Q6: Identifiez vous en faisant « docker login », puis faites la commande suivante:

« docker push <username_dockerhub>/<nom_repo> ».

Que se passe-t-il ?

Rendez-vous sur:

[https://hub.docker.com/r/](https://hub.docker.com/r/<username_dockerhub>/<nom_repo>) <username_dockerhub>/<nom_repo>

Que constatez-vous ?

Q7: Transférez le fichier « docker-compose.yml » depuis votre ordinateur vers votre environnement virtuel. Tapez la commande « docker-compose up », puis cliquez sur le bouton « 3000 » en haut de la page. Actualisez plusieurs fois la page, que constatez-vous ?

A faire

Travail à rendre: Terminer le TD, et envoyer sur Slack un lien vers votre répertoire Docker.