

Département Informatique

Matthieu Le Lain,
Xavier Roirand

BUT — 3^e année

R5.A.09 — TP 3

Gitlab CI

-

Objectif du TP

L'entreprise CarPart est très satisfaite de vos deux applications FastAPI. Pour poursuivre avec vous la transition vers le cloud, elle souhaite que vous mettiez en place une chaîne d'intégration continue sur Gitlab afin de construire automatiquement les conteneurs Docker et les pousser sur le Docker Hub de l'entreprise.

Appuyer vous sur le cours pour mettre en œuvre le TP.

1. Création d'un compte Docker Hub

Afin de pouvoir envoyer les images de conteneur sur le Hub, vous devez disposer d'un compte sur Docker Hub.

- Créez un compte Docker Hub
- Créez les différents Token Docker Hub dont vous aurez besoin

2 – Préparation du CI/CD

2.1 – Compte et dépôt Gitlab

Vous devez disposer d'un compte Gitlab afin de pouvoir créer un projet et pousser votre code applicatif sur un dépôt Gitlab. Si ce n'est pas le cas, créer un compte Gitlab et ajouter votre code. Attention le choix d'avoir un ou deux dépôts pour chacune des solutions applicatives aura un impact sur l'automatisation de votre déploiement (voir 2.2)

2.2 – Gitlab CI

Préparer le fichier gitlab-ci.yml permettant de :

- Tester l'installation de votre solution
- Builder votre solution applicative pour générer un conteneur avec Buildah et le pousser sur le gitlab registry
- Builder votre solution applicative pour générer un conteneur avec Docker et le pousser vers le Docker Hub

Deux possibilités s'offrent à vous :

- Un seul dépôt Gitlab contenant les deux solutions applicatives avec un seul fichier Gitlab-ci pour le build et le push
- Deux dépôts Gitlab contenant chacune des solutions applicatives avec un fichier gitlab-ci par dépôt permettant de gérer plus finement les déploiements.

En finalité, l'entreprise souhaite avoir, deux conteneurs, pour chacune des solutions applicatives, dans deux registry différents sur Docker Hub.