

# Projet Pro



# Python

Vous devez développer un plugin Telegraf qui enregistre des valeurs données par powertop (<https://wiki.archlinux.org/index.php/powertop>) sur l'infrastructure docker hébergeant le blog (cf partie Docker) dans une DB InfluxDB. Afin d'y suivre les graphes de consommation électrique des machines. (<https://www.influxdata.com/training/write-your-own-telegraf-plugin/>)

Votre projet Golang sera testé et déployé en image Docker via GITLAB CI

## DOCKER

En plus du projet Golang, il vous sera demandé d'installer une infrastructure Web hébergeant un Blog (libre choix de la solution, mais obligation d'une BDD déportée). Tous les services devront être containerisés dans leur propre environnement en respectant :

- Un LoadBalancing est obligatoire
- Vous devez créer vos propres images Docker via dockerfile (**toutes récupérations d'image existante est éliminatoire**)
- Les images seront rajoutées à un DOCKER REGISTRY PRIVÉE.
- Le Docker registry devra également gérer l'authentification et les contrôles d'accès des utilisateurs via un « authentification server ». Vous aurez le choix de référencer vos utilisateurs via des ACL, ou MongoDB, ou LDAP, ou un fichier YAML (partie supplémentaire pour les quadrinômes)

Votre projet Blog sera testé et déployé en image Docker via GITLAB CI

## CI/CD

Vous utiliserez Gitlab-CI pour automatiser toutes les tâches de déploiement mais également les tests du bon fonctionnement des applications.

Nous vous proposons un découpage en trois stages :

- Build,
- Test
- Deploy.

La partie « Test » vérifiera que vos projets GO et BLOG sont fonctionnels.

**Le déploiement se fera sur 3 containers en Load Balancing avec l'outil de votre choix.**

# LE LIVRABLE

Pour permettre de consulter les sources et le fonctionnement de votre projet, vous aurez à fournir

- Une machine virtuelle comprenant votre usine logicielle.
- Un rendu type Blog ou Vlog (dans votre infrastructure Web hébergeant un Blog) expliquant comment utiliser le projet et comment le tester. N'oubliez pas de présenter des captures d'écran (ou des captures vidéo) pour faciliter les explications et de faire preuve de vulgarisation.
- Un répertoire qui contient les sources de votre développement Golang

La date limite de livraison est le 15 MAI 2023