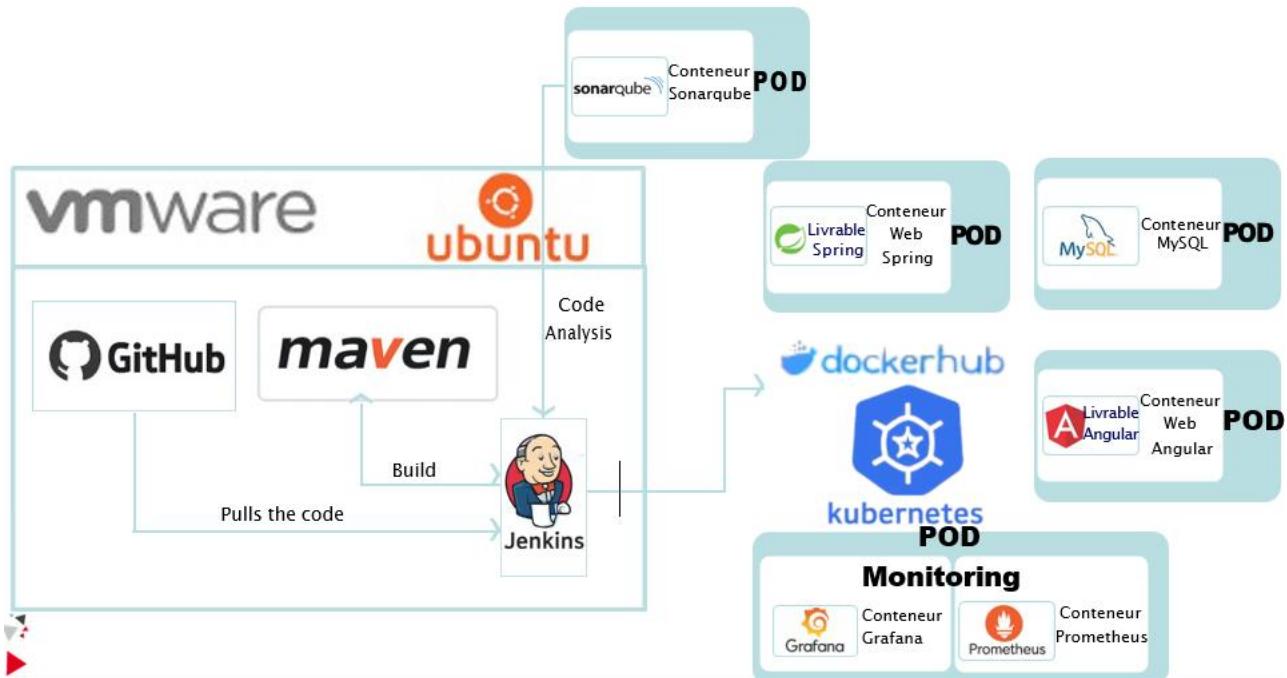


Projet DevOps

Travail Individuel – Compte pour 60% du Contrôle Continu

Architecture



Ce projet a pour objectif d'évaluer votre capacité à mettre en place une chaîne DevOps complète autour d'une application Spring Boot, depuis le code source jusqu'à la livraison continue. L'étudiant devra démontrer sa maîtrise des outils DevOps (Maven, Git, SonarQube, Jenkins, Docker, Grafana) ainsi que sa capacité à automatiser l'intégration et le déploiement.

1. Les Codes sources Projet Spring sont sur OneDrive.
2. Chaque étudiant récupère ce même projet déjà opérationnel et le fait tourner sur sa propre machine.
3. Chaque étudiant applique les commandes Maven (en ligne de commande) sur le projet (clean, install, package, ...).
4. Commiter ce code modifié sur GIT.
5. Lancer Sonar sur ce projet.

6. Configurer **Jenkins** pour automatiser les tâches ci-dessus.
7. Livraison Continue avec **Docker (Spring, MySQL)**.
8. Utiliser **Grafana** pour le **Continuous Monitoring**.

a. Git :

Le nom du repo Git doit suivre le format suivant : **classe_nomEtudiant**
(exemple : 5SIM1-nomEtudiant)

b. Image Docker :

Le nom de l'image Docker doit suivre le format **nomPrenom_classe_nomProjet**
(exemple : FoulenBENFOULEN-5SIM1-nomProjet).

Améliorations possibles :

- How to Configure Email Notification in Jenkins?
<https://www.360logica.com/blog/email-notification-in-jenkins/>
- webhook
- Jenkins distant
- ngrok
- IaC (Infrastructure as a Code). Vous pouvez utiliser Terraform pour automatiser la configuration des conteneurs Docker et Ansible pour automatiser la configuration des outils dans les conteneurs.
- Vous pouvez transformer le projet Spring en un projet en microservices.
<https://spring.io/guides/gs/service-registration-and-discovery/>
- Kubernetes.
- Ajouter la partie Angular et la déployer dans un conteneur Docker.

