

Sujet TP 3 Mise en place pour une application en Python d'une chaîne d'intégration continue permettant de valider l'application avant de la packager dans une image de conteneur

Table des matières

Présentation du Contexte.....	2
Organisation.....	2
Description détaillé des éléments attendus.....	2
Fichiers de pipeline.....	2
Fichier d'image docker.....	3
Environnement d'exécution.....	3
Documentation.....	3
Rendu.....	3

Présentation du Contexte

Vous arrivez dans sur une équipe qui met en place une usine logiciel afin d'aider les développeurs à gagner du temps et à améliorer la qualité de leur applications. Votre rôle de DevOps est de mettre en œuvre une chaîne d'intégration continue pour les applications en Python.

Vous allez donc devoir :

- Choisir un outil d'orchestration continue (Jenkins, Gitlab CI, Github Actions ou un autre)
- Ecrire un pipeline qui fera les actions suivantes pour des applications Python:
 - Récupérer le code source dans un dépôt git
 - Lancer un linter qui vérifiera les normes de codage
 - Vérifier les copier-coller dans le code
 - Analyser la complexité cyclomatique
 - Lancer les tests unitaires de l'application
 - Builder l'image de Docker
 - Pusher l'image Docker sur le docker hub
- Le tout sera accompagner d'une petite application Python exemple ,vous pouvez utiliser par exemple celle-ci <https://github.com/vanessakovalsky/python-api-handle-it> (ou une autre de votre choix)

Organisation

Cet exercice se fait individuellement et sera à rendre sous la forme d'un dépôt Git public (sur github ou gitlab comme vous préférez) directement dans Teams

Le rendu sera à faire au plus tard le lundi 31/10/2023 à 23h59

Cet exercice permettra de valider vos compétences pour votre dossier d'évaluation de compétences individuelle dans le cadre de votre titre RNCP Administrateur Devops. Il concerne la validation de l'activité type 2 Déployer en continu une application

Description détaillé des éléments attendus

Fichiers de pipeline

Les fichiers de pipeline devra contenir les étapes suivantes :

- Récupérer le code source dans un dépôt git
- Lancer un linter qui vérifiera les normes de codage
- Vérifier les copier-coller dans le code
- Analyser la complexité cyclomatique

- Lancer les tests unitaires de l'application
- Produire des rapports et les rendre accessible pour les 3 étapes d'analyse qualité, ainsi que pour l'exécution des tests
- Builder l'image de Docker
- Pusher l'image Docker sur le docker hub

Le fichier devra également rendre paramétrable les éléments suivants :

- url du dépôt Git (et identifiant si nécessaire (dans le cas d'un dépôt privé)
- paramètre des différents outils d'analyse qualités (fichier de configuration et son chemin)
- chemin des tests unitaires et fichier de configuration
- nom de l'image et version
- chemin et contexte du build de l'image docker
- identifiant de connexion pour le hub docker

Fichier d'image docker

Le fichier de l'image docker devra contenir à minima les étapes suivantes (vous pouvez en ajouter d'autres si nécessaires):

- Utilisation d'une image officielle de Python
- Copy des fichiers
- Installations des dépendances avec pip et un fichier requirements.txt
- Lancement de l'application lors du lancement du conteneur

Environnement d'exécution

Un fichier docker compose permettant de lancer l'environnement (outils de CI + tout ce qui est nécessaire) doit être fourni afin de pouvoir tester le pipeline. Celui-ci devra contenir tous les conteneurs et paramètres nécessaires au démarrage de la CI (sans avoir besoin de se connecter aux conteneurs pour faire des actions manuelles)

Documentation

Rédiger une documentation qui contiendra :

- Description des fichiers fournis
- Description des pré-requis
- Schéma de présentation des étapes du pipeline
- Configuration des paramètres de pipeline

Rendu

Le rendu se fait sous la forme d'un dépôt git sur Teams contenant:

- un fichier avec le pipeline (format en fonction de l'outils de CI choisi)

- un fichier Dockerfile
- un fichier docker-compose pour tester votre pipeline
- une documentation sous la forme de votre choix
- Vous pouvez ajouter autant de scripts / fichiers / documents que vous jugerez nécessaire pour réaliser cet exercice