

Créer un pipeline pour une application

Nous allons voir dans ce TP la mise en place d'un pipeline pour une application.

Mettre le projet sur Github dans un repo privé

Mettre le projet dans un repository privé permet de protéger l'accès au code des utilisateurs non souhaités.

1. Créer un repo distant sur Github privé

Afin de versionner le code de l'application dans un repository distant, n'oubliez pas de le créer sur github.

2. Envoyer ce projet sur votre repo Github

Par ligne de commande, en utilisant votre IDE favori, envoyez votre code dans le repository distant.

3. Vérifier que vous pouvez le cloner

Vérifiez que vous avez bien accès à ce repository distant en le clonant manuellement en utilisant la ligne de commande et non pas votre IDE. En effet, cela sera un premier test pour s'assurer que l'on puisse y accéder depuis votre pipeline CI/CD.

Créer un pipeline de CI/CD as code dans une démarche Top-down

Il est temps désormais de créer un pipeline CI/CD.

1. Créer un nouveau pipeline dans Jenkins

Créer un nouveau pipeline dans Jenkins en écrivant directement le contenu de ce pipeline en déclaratif à l'intérieur de la page de Jenkins, c'est-à-dire sans passer par un Jenkins file dans un premier temps. Cela nous permettra de tester le pipeline pour voir s'il fonctionne bien sans à avoir faire des commits inutiles dans le repository. Je vous rappelle ci-dessous, un pipeline typique (sans la quality gate) : checkout/clone > clean > compile > tests > packaging > archive/deploy to repository

2. Vérifier qu'il fonctionne bien

Lancer un build pour bien vérifier que cela fonctionne bien, vérifiez bien le dossier workspace et le dossier jobs afin de voir si le traitement a été fait correctement jusqu'à l'archivage de l'artefact.

Ajouter l'inspection de la qualité

Inspecter la qualité du code permet de vérifier que le code est maintenable sur le long terme. Nous pouvons automatiser également cette inspection grâce à sonarqube.

1. Créer un projet dans sonarqube

Créer un projet à l'intérieur de sonarqube pour désigner le projet qu'on va inspecter. On choisira comme nom, le nom du repository dans le SCM.

2. Exécuter une inspection pour vérifier que tout fonctionne bien

Exécuter une inspection pour vérifier que tout fonctionne bien et regarder dans sonarqube pour bien vérifier le rapport de qualité de l'application.

3. Modifier le pipeline pour ajouter l'inspection avant le packaging

Ajouter dans le pipeline l'inspection juste avant le packaging afin de bien vérifier que la qualité du code passe les exigences de la quality gate.