

Jenkins pipelines : Intégration et Livraison Continues

TP 5

EFREI

Notions Abordées :

- Pipeline entant que séquence de jobs

Objectif

L'objectif de ce TP est d'utiliser le serveur jenkins pour compiler, tester et déployer régulièrement notre application. Nous allons suivre un processus complet de l'intégration continue.

(A) Partie I : Séquence de Jobs

Dans cette partie, nous verrons comment créer des jobs, les lancer, faire des tests unitaires avec JUnit.

- (a) Vous allez commencer par mettre le projet sous le serveur GITHub. (voir TP git).
- (b) Par la suite, sur Jenkins, on va créer une séquence de Jobs permettant de compiler et de tester un projet maven.
 - i. Créer un premier job comme étant un **projet freestyle**. Ce job à pour objectif de compiler un projet maven.
 - Créer le job sur **nouveau item > Créer un projet freestyle**
 - Placez vous sur la section "**Gestion de code source**" puis cochez l'option **git**, et entrez votre dépôt (repository).

The screenshot shows the Jenkins configuration interface for a new job. The 'Gestion de code source' tab is active. Under 'Gestion de code source', the 'Git' radio button is selected. In the 'Repositories' section, the 'Repository URL' is 'https://github.com/safaa-09/Triang07.git' and the 'Credentials' dropdown is set to 'safaa-09/***** (Git-id)'. There is an 'Ajouter' button next to the credentials. In the 'Branches to build' section, the 'Branch Specifier (blank for \'any\')' is set to '*/master'. There are buttons for 'Avancé...', 'Add Repository', and 'Add Branch'.

Note:

- Le plugin git doit être déjà installé sur Jenkins.

- Si votre build génère une erreur de connexion Git, vous allez devoir lui indiquer le chemin de **git.exe** sur votre machine. Pour ce faire, placez vous sur : **Jenkins > Administrer Jenkins > Configuration Globale des outils > Git > Path to Git executable.**

Git

Git installations

Git

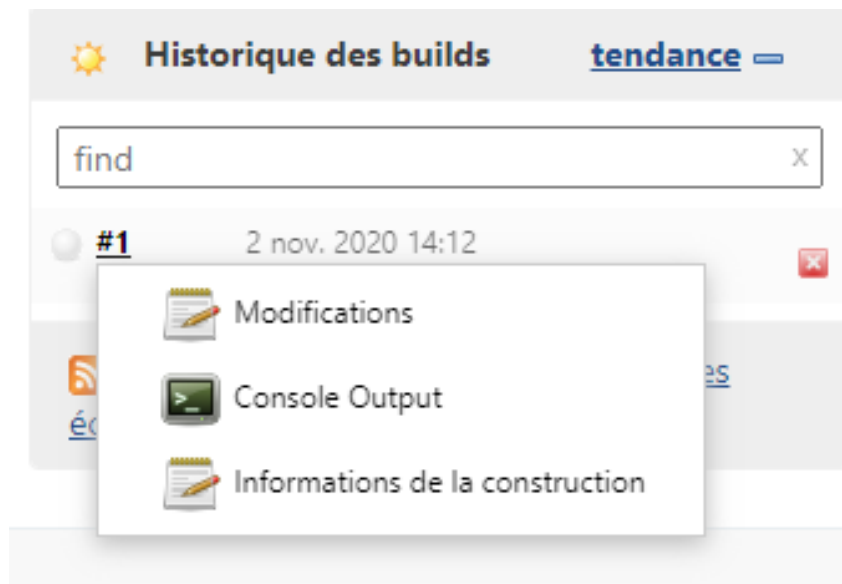
Name

Path to Git executable

☐ Install automatically

[Delete Git](#)

- Sur la section **Build**, sélectionner **Ajouter une étape du build > Invoquer les cibles Maven de haut niveau**
- Cibles maven, **compile** (puisque'on veut compiler notre application maven)
- Sauvegarder et lancer le job. Vous pouvez voir ce qui se passe durant le build du job, avec la console qui vous informe des actions qui se déroulent.



- Créer un deuxième job comme étant un **projet freestyle**. Ce job a pour objectif de tester un projet maven.

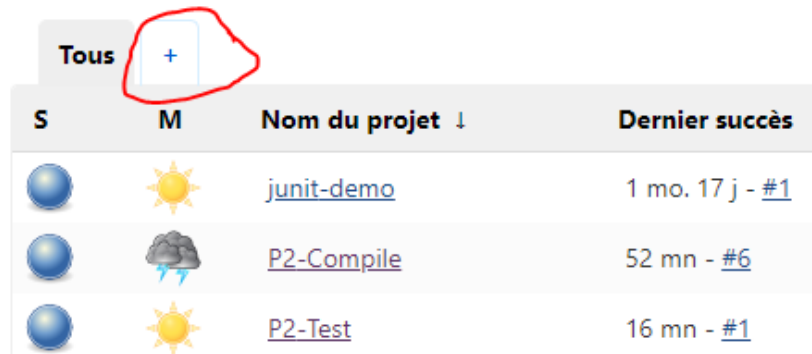
De la même façon

- Créer le job sur **nouveau item > Créer un projet freestyle**
- Placez vous sur la section "**Gestion de code source**" puis cochez l'option **git**, et entrez votre dépôt (repository).
- Dans "Cibles maven", entrez l'option **test** (puisque'on veut lancer les tests unitaires de notre application maven)
- Dans ce job, dans la section "Ce qui déclenche le build", Cocher l'option "Construire après le build sur d'autres projets", et choisissez le job précédent.
- Sauvegarder et lancer le job.

- De la même façon, créer un troisième job, permettant de mettre l'application dans un jar. Ce job doit se déclencher après que les deux premiers aient terminé.

(c) Télécharger le plugin "Delivery Pipeline View"

(d) Créer une delivery pipeline view



| S | M | Nom du projet ↓ | Dernier succès |
|---|---|----------------------------|---------------------------------|
| | | junit-demo | 1 mo. 17 j - #1 |
| | | P2-Compile | 52 mn - #6 |
| | | P2-Test | 16 mn - #1 |

Nom de la vue

☐ **Delivery Pipeline View**
Continuous Delivery pipelines, perfect for visualization on information radiators. Shows one or more delivery pipeline instances, based on traditional Jenkins jobs with upstream/downstream dependencies.

☐ **Delivery Pipeline View for Jenkins Pipelines**
Continuous Delivery pipelines, perfect for visualization on information radiators. Shows one or more delivery pipeline instances, based on Jenkins pipelines (created using the Pipeline or Workflow plugin).

☐ **Ma vue**
Cette vue affiche automatiquement tous les jobs auxquels l'utilisateur a accès.

☐ **Vue liste**

- (e) Dans la section *Pipeline > Composant*, sélectionnez le premier job à déclencher dans votre pipeline et sauvegardez.
- (f) Relancez le premier job, sur votre pipeline vous devez avoir la représentation graphique suivante :

