

TP1 : Installation en service, Configuration, Premier job

Objectifs

- Installation de Jenkins en service, configuration du service
- Configurer le système
- Configurer les outils de base
- Créer un premier job freestyle

1. 1 Installation en service

En fonction de votre OS, exécuter la procédure d'installation en service adéquat.

Avec un navigateur, accéder à localhost:8080 et continuer la procédure **sans** installer de plugin

Éditer le fichier `/etc/default/jenkins` :

- modifier les paramètres de la JVM

Effectuer un redémarrage du service :

```
sudo service jenkins restart
```

1.2 Configuration générale et outils

Configuration serveur de mail

Si nécessaire installer le plugin **mailer**

Ensuite aller dans la configuration système « *Administrer Jenkins* → *Configurer le système* » et renseigner les paramètres du serveur smtp

login : stageojen@plbformation.com

password : stageojen

smtp : smtp.plbformation.com

(port smtp 587)

Tester l'envoi de mail

Configuration des outils (JDK, Maven, Git)

1. Cliquer sur « *Administrer Jenkins* → *Configuration Globale des outils* »
2. Dans la section JDK, indiquer soit un JAVA_HOME pré-installé soit un installateur automatique
3. Dans la section Maven, utiliser l'installation automatique
4. Dans la section Git, indiquer votre installation de Git
(Si la section Gitt n'est pas présente installer le plugin **Git Client**)
5. Enregistrer vos modifications

1.3 : Création de job

Mise en place dépôt Git

Vérifier l'installation de git sur votre machine

Installer le plugin « **git** »

Reprendre les source du projet et les décompresser.

Initialiser le dépôt :

git init

Création job et test

1. De retour sur la page d'accueil, Activer le lien « Créer un nouveau job »
2. Donner un nom et choisir job freestyle
3. Configurer le SCM afin qu'il point vers un dépôt git local
4. Indiquer que le build est déclenché à chaque changement dans GIT et que le repository est interrogé toutes les 5 mns. (*/*5 * * * *)
5. Ajouter une étape de build affichant toutes les variables d'environnement disponibles
6. Ajouter ensuite une étape de build qui invoque la cible Maven « **clean package** », indiquer le chemin vers le *pom.xml*
7. Ajouter une étape « post-build » permettant d'afficher les résultats des tests : « Publier les rapports JUnit »
8. Ajouter une autre étape « post-build » pour archiver tous les jars produits
9. Lancer le build manuellement et observer la page d'accueil du projet

Déclenchement de job

1. Modifier un fichier du projet provoquant une erreur dans les tests, committer le changement dans le repository
Voir fichier *simple-parent/simple-command/src/test/java/org/sonatype/mavenbook/weather/yahoo/YahooParserTest.java*
2. Attendre le déclenchement du job
3. Observer la page d'accueil qui doit afficher un graphique de tendance sur l'exécution des tests.
4. Ensuite restaurer votre modification