



Jenkins

Utilisation d'un Jenkinsfile

- Modifier le fichier Jenkinsfile de votre projet spring-framework-petclinic sous GitHub de la façon suivante :

```
pipeline {
  agent any

  stages {
    stage('Checkout') {
      steps {
        echo 'Checkout...'
        git 'https://github.com/XXXXXX/spring-framework-petclinic.git'
      }
    }
    stage('Build') {
      steps {
        parallel(
          build: {
            echo 'Building...'
            sh 'mvn clean package'
            archiveArtifacts 'target/*.war'
          },
          sonar: {
            echo "Analyse Sonar..."
          }
        )
      }
    }
    stage('Deploy') {
      steps {
        echo 'Deploying...'
        sh 'cp target/*.war /var/tmp'
      }
    }
  }
}
```

- Remplacer XXXXXX dans l'URL Git par votre compte GitHub

Pour plus d'informations sur la construction du Jenkinsfile, aller à l'adresse suivante : <https://jenkins.io/doc/book/pipeline/jenkinsfile/>

Création de l'item Petclinic_pipeline

- Description : cet item va récupérer automatiquement le Jenkinsfile à la racine du projet et va permettre de lancer les différentes tâches pour le déploiement.
- Nom : Petclinic_deployQualification
- Type : Pipeline

Saisissez un nom

Petclinic Pipeline

» Champ obligatoire



Construire un projet free-style

Ceci est la fonction principale de Jenkins qui sert à builder (construire) votre projet. Vous pouvez intégrer tous les systèmes de build. Il est même possible d'utiliser Jenkins pour tout autre chose qu'un build logiciel.



Construire un projet maven

Construit un projet avec maven. Jenkins utilise directement vos fichiers POM et diminue radicalement l'effort est encore en bêta mais elle est disponible afin d'obtenir vos retours.



Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (form organizing complex activities that do not easily fit in free-style job type.

- Déclarer un pipeline avec la définition "**Pipeline script from SCM**"
- Choisir le SCM "**Git**" et renseigner les informations du repository

Pipeline

Definition: Pipeline script from SCM

SCM: Git

Repositories

Repository URL:

Credentials: [Ajouter](#)

[Avancé...](#)

[Add Repository](#)

Branches to build

Branch Specifier (blank for 'any'): [Add Branch](#)

Navigateur de la base de code: (Auto)

Additional Behaviours: [Ajouter](#)

Script Path:

[Lightweight](#) checkout ☒

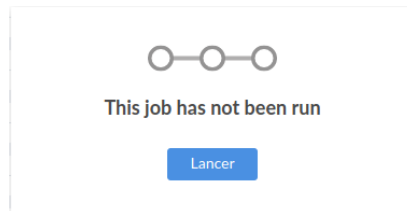
- Ouvrir le pipeline avec "**Open Blue Ocean**"

Jenkins

Jenkins > Petclinic Pipeline

- Back to Dashboard
- Status
- Changes
- Lancer un build
- Supprimer Pipeline
- Configurer
- Full Stage View
- Open Blue Ocean**
- Rename
- Pipeline Syntax

- Lancer le build



- Build en cours de construction :

Petclinic Pipeline 10 Pipeline Modifications Tests Artefacts [Settings] [Build] Déconnexion

Branche: — 10s Aucune modification
Commit: — - Démarré par l'utilisateur Administrateur

Build / build - 4s [Restart Build] [Download]

Step	Duration
> Building... — Print Message	<1s
> mvn clean install — Shell Script	4s

- Fin de la construction :

Petclinic Pipeline 10 Pipeline Modifications Tests Artefacts [Settings] [Build] Déconnexion

Branche: — 2m 22s Aucune modification
Commit: — a few seconds ago Démarré par l'utilisateur Administrateur

Build / build - 2m 14s [Restart Build] [Download]

Step	Duration
> Building... — Print Message	<1s
> mvn clean install — Shell Script	2m 14s
> target/*.war — Archiver des artefacts	<1s