



# GIT et JENKINS

Mise à jour janv. 2023

**Durée** : 3 jours - 21 heures

## OBJECTIFS PÉDAGOGIQUES

- Comprendre les principes de l'intégration continue en vue de son implémentation
- Intégrer Jenkins avec les autres outils (SCM, gestionnaire de tickets...)
- Mettre en place un serveur Jenkins automatisant les build
- Automatiser les tests, les audits de code et les déploiements sur la plateforme d'intégration Jenkins
- Présentation et utilisation de GIT

## PRÉREQUIS

- Développeurs, chefs de projet
- Connaissance du langage Java et notions du cycle de développement

## PARTICIPANTS

### MOYENS PÉDAGOGIQUES

- Réflexion de groupe et apports théoriques du formateur
- Travail d'échange avec les participants sous forme de réunion-discussion
- Utilisation de cas concrets issus de l'expérience professionnelle
- Validation des acquis par des questionnaires, des tests d'évaluation, des mises en situation et des jeux pédagogiques.
- Alternance entre apports théoriques et exercices pratiques (en moyenne 30 à 50%)
- Remise d'un support de cours.

### MODALITÉS D'ÉVALUATION

- Feuille de présence signée en demi-journée
- Évaluation des acquis tout au long de la formation
- Questionnaire de satisfaction
- Attestation de stage à chaque apprenant
- Positionnement préalable oral ou écrit
- Évaluation formative tout au long de la formation
- Évaluation sommative faite par le formateur ou à l'aide des certifications disponibles

### MOYENS TECHNIQUES EN PRÉSENTIEL

- Accueil des stagiaires dans une salle dédiée à la formation, équipée d'ordinateurs, d'un vidéo projecteur, d'un tableau blanc et de paperboard

### MOYENS TECHNIQUES DES CLASSES À DISTANCE

- A l'aide d'un logiciel comme Teams, Zoom... un micro et éventuellement une caméra pour l'apprenant,
- suivez une formation en temps réel et entièrement à distance. Lors de la classe en ligne, les apprenants interagissent et communiquent entre eux et avec le formateur
- Les formations en distanciel sont organisées en Inter-Entreprises comme en Intra-Entreprise
- L'accès à l'environnement d'apprentissage (support de cours, labs) ainsi qu'aux preuves de suivi et d'assiduité (émargement, évaluation) est assuré
- Les participants recevront une convocation avec lien de connexion
- Pour toute question avant et pendant le parcours, une assistance technique et pédagogique est à disposition auprès de notre équipe par téléphone au 03 25 80 08 64 ou par mail à [secretariat@feep-entreprises.fr](mailto:secretariat@feep-entreprises.fr)

### ORGANISATION

- Les cours ont lieu de 9h à 12h30 et de 13h30 à 17h

### PROFIL FORMATEUR

- Nos formateurs sont des experts dans leurs domaines d'intervention
- Leur expérience de terrain et leurs qualités pédagogiques constituent un gage de qualité.

### ACCESSIBILITÉ

- Notre organisme peut vous offrir des possibilités d'adaptation et/ou de compensations spécifiques si elles sont nécessaires à l'amélioration de vos apprentissages sur l'ensemble de nos formations. Aussi, si vous rencontrez une quelconque difficulté, nous vous invitons à nous contacter directement afin d'étudier ensemble les possibilités de suivre la formation

# Programme de formation

## Introduction à l'intégration continue

- Principes de l'intégration continue.
- Développement agile et intégration continue.
- Prérequis à l'intégration continue.
- Techniques de développement adaptées à l'intégration continue.
- Différents types de tests et d'audit d'une application.
- Cycle de vie du processus de développement.
- Outils annexes et intégration.

## Mise en place et automatisation du Build

- Mise en place du serveur, les différents types d'installation.
- Configuration : page principale de la configuration, configuration de Git/SVN, serveur de mail.
- Jenkins et le Build, les meilleures pratiques et méthodes recommandées.
- Stratégies et techniques de notification.
- Fixer les dépendances entre les travaux de Build.
- Jenkins et Maven : rappel sur Maven, configuration du Build Maven, déploiement dans un repository Maven.

## Qualité du code

- Introduction, intégration de la qualité dans le processus de build.
- Outils d'analyse : Checkstyle, FindBugs, CPD/PMD.
- Configuration du rapport qualité avec le plugin Violations.
- Rapport de complexité, sur les tâches ouvertes.

## Automatisation des tests

- Introduction, terminologie du test.
- Automatisation des tests unitaires et d'intégration.
- Configuration des rapports.
- Mesurer la couverture de test.
- Automatisation des tests d'acceptance.

- Automatisation des tests de performance avec JMeter.
- Optimiser les temps d'exécution des tests.

## Automatisation du déploiement

- Mise en place du script de déploiement.
- Mise à jour des bases de données.
- Tests minimaux. Retour en arrière.

## Administration d'un serveur Jenkins

- Activation de la sécurité et mise en place simple.
- Différents types de bases utilisateurs.
- Gestion des autorisations et des rôles.
- Journalisation des actions utilisateur.
- Gestion de l'espace disque.
- Monitoring de la charge CPU.
- Sauvegarde de la configuration.

## Présentation de Git

- Concepts de base du contrôle de version.
- La gestion centralisée ou distribuée.
- Les différentes solutions de gestion de versions : (Git, CVS, SVN, Mercurial, Bazaar...).
- Apports la décentralisation. Principe de fonctionnement.

## Configuration

- Configuration du fichier gitconfig. La console.
- Déclaration d'un outil d'analyse de différentiel de versions.

## Utilisation de Git, les fondamentaux

- Le modèle objet Git : blob, tree, commit et tag.
- Le répertoire Git et le répertoire de travail.
- L'index ou staging area.
- Création et initialisation un dépôt.
- Les concepts de branche, tag et de dépôt.
- Outil de visualisation Gitk.