Intégration continue, bonnes pratiques de mise en œuvre

Cours Pratique de 3 jours - 21h
Réf : DIN - Prix 2024 : 2 280CHF HT

Le prix pour les dates de sessions 2025 pourra être révisé

L'intégration continue est un des aspects essentiels des Méthodes Agiles. Elle vise à garantir la qualité et la fluidité du cycle de développement ainsi que le déploiement des nouvelles releases. Cette formation présente les bonnes pratiques et techniques associées pour la mettre en œuvre.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Comprendre les composantes et les principes de l'intégration continue

Manipuler un gestionnaire de contrôle des versions

Comprendre les mécanismes de la construction du logiciel et les outils de Build associés

Configurer un projet sur un serveur d'intégration continue

Déchiffrer les principaux métriques d'outils d'analyse de code

Appréhender le rôle des dépôts d'artefacts et la gestion des configurations

LE PROGRAMME

dernière mise à jour : 05/2019

1) Introduction

- Les différents types d'environnement : développement, recette, production.
- La gestion des configurations : système et applicative.
- Le rôle des conteneurs applicatifs : Docker.
- Centralisation des configurations : puppet, Ansible.
- Industrialisation des déploiements.
- Mettre en place une plateforme d'intégration continue.

2) Gestion des versions

- Les fonctionnalités.
- Les différents gestionnaires de sources : centralisée ou distribuée.
- Les problématiques d'intégration des changements.
- Le rôle des branches et tags.

Travaux pratiques : Manipulation d'un dépôt Git.

3) Automatisation des Builds

- Qu'est-ce que la construction du logiciel ?
- Les Builds, cycle de vie de projet.
- Les outils de Build avec Maven.
- Plug-ins, profils et modules.

Travaux pratiques : Mise en place de la construction d'un projet simple (Maven).

4) Gestion des dépôts

- Rôles des gestionnaires de dépôts.
- Les types de dépôts : public, interne.

PARTICIPANTS

Chefs de projet, développeurs, architectes.

PRÉREQUIS

Connaissances de base en développement logiciel.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques…
Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

• Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
• À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
• Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

- La gestion des livrables (artfacts): releases, snapshots, déploiement.
- Organisation des dépôts : développement, test, production.
- Intégration des gestionnaires des dépôts avec les outils de Build, avec les outils de gestion des taches.
- Exemples de solutions : Nexus, Artifactory...

Travaux pratiques : Publication d'artefacts dans Nexus.

5) Automatisation des tests

- Quels tests et pour faire quoi ?
- Les environnements de tests.
- Présentation d'outils de test.
- La mise en œuvre de l'automatisation des tests.

Travaux pratiques : Mise en place de tests automatisés avec JUnit.

6) Serveur d'intégration continue

- Le rôle du serveur d'intégration continue.
- Les grandes fonctionnalités.
- La gestion des tâches (job).
- La dépendance entre tâches.
- Les serveurs : Hudson/Jenkins, CruiseControle, Bamboo...

Travaux pratiques : Configuration du projet sur un serveur d'intégration continue Jenkins.

7) Mise en place de métriques

- La génération de rapports.
- Les outils d'analyse et de reporting (PMD, Findbugs, Cobertura, Emma, Checkstyle...).
- La publication des résultats.

Travaux pratiques : Mise en place des outils de métriques.

8) Gestion du feedback

- La gestion d'un cycle complet.
- La traçabilité des changements.

Travaux pratiques : Utilisation de Sonar.

LES DATES

CLASSE À DISTANCE
2024 : 18 nov.

2025 : 31 mars, 18 juin, 29 sept., 22 déc.