

Java, tests et qualité

Cours Pratique de 3 jours - 21h

Réf : JTS - Prix 2024 : 1 870€ HT

La qualité des logiciels est devenue un enjeu primordial. Vous mettrez en œuvre des outils Open Source afin d'industrialiser vos développements et garantir cette qualité. Vous apprendrez à gérer les sources, la documentation, les anomalies et les évolutions d'une application ainsi que l'ensemble des tests associés.

OBJECTIFS PÉDAGOGIQUES

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

Industrialiser le développement logiciel

Gérer les sources d'un développement via un système centralisé ou décentralisé

Gérer les anomalies et les évolutions fonctionnelles d'une application

Mettre en place les normes de codage et la documentation d'un logiciel

Mettre en œuvre différents types de tests (tests unitaires, fonctionnels, de robustesse et de performance)

Automatiser des tâches définies

LE PROGRAMME

dernière mise à jour : 08/2018

1) Introduction générale

- Qu'est-ce que la qualité logicielle ? Pourquoi en faire ? Comment industrialiser le développement logiciel ?

2) Gestion des sources

- Définition, intérêt dans l'organisation du travail, unicité de l'environnement de développement.

Travaux pratiques : Présentation des logiciels libres de gestion des sources CVS et/ou Subversion (SVN).

3) Outil de gestion des bugs, anomalies et évolutions fonctionnelles

- Description fonctionnelle, intérêt pour les développeurs, les utilisateurs, les managers.
- Définition et intérêt du tri des bugs, anomalies et évolutions fonctionnelles.

Travaux pratiques : Présentation des logiciels libres de gestion des bugs, anomalies et évolutions fonctionnelles Bugzilla et/ou Mantis.

4) Règles de codage et documentation

- Intérêt (au-delà des oppositions de clans). Au-delà du style : règles d'organisation des fichiers...

- Intérêt de la documentation, documentation du code et autre documentation (HOWTO...).

Travaux pratiques : Revue des recommandations et règles de codages de Sun. Présentation et paramétrage de CheckStyle. Exercices de documentation de code.

5) Tests

- Pourquoi faire des tests ?

- Présentation des différents types de tests : tests unitaires, fonctionnels, de robustesse et de performance.

PARTICIPANTS

Développeurs Java/Java EE, responsables qualité et chefs de projets.

PRÉREQUIS

Bonnes connaissances en programmation Java.

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.

- Quels tests lancer et quand ?
- Utilité des objets "Mock" et "Fake" durant les tests unitaires. Couverture des tests unitaires.

Travaux pratiques : Pratique des tests unitaires avec JUnit (logiciel libre) et d'autres outils. Présentation des tests fonctionnels avec les outils Mercury (logiciel commercial), HttpUnit (logiciel libre).

6) Procédures de livraison

- Intérêt des procédures de livraison. Etapes classiques et non classiques (tests manuels, de robustesse).

7) Automatisation des tâches

- Présentation de tâches devant être automatisées.

Travaux pratiques : Présentation et introduction à Ant (logiciel libre) et/ou Maven.

8) Conclusion générale : mesures de la qualité

- Synthèse des mesures qualité : convention de codage, documentation, couverture de tests, automatisation des procédures.
- Mise en place d'un tableau de bord de la qualité.

LES DATES

Nous contacter