

Java, tests et qualité



MR-179 3 Jours (21 Heures)

Description

La qualité des logiciels est aujourd'hui un défi majeur. Vous allez implémenter les outils Open Source pour industrialiser vos développements et assurer cette qualité. Vous apprendrez à gérer les sources, la documentation, les anomalies et les évolutions d'une application ainsi que tous les tests connexes.

À qui s'adresse cette formation ?

Pour qui

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

Prérequis

Bonnes connaissances en programmation Java.

Les objectifs de la formation

- Industrialisation du développement logiciel
 - Gestion des sources de développement au moyen d'un système centralisé ou décentralisé
 - Gestion des anomalies et des changements fonctionnels dans une application
 - Mettre en place les normes de codage et la documentation du logiciel
 - Implémenter différents types de tests (tests unitaires, fonctionnels, robustes et de performance)
- Automatisation de tâches définies

Programme de la formation

Introduction générale

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

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).

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.

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.

Tests

- Pourquoi faire des tests ? Présentation des différents types de tests : tests unitaires, fonctionnels, de robustesse et de performance.
- 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).

Procédures de livraison

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

Automatisation des tâches

- Présentation de tâches devant être automatisées.
- Travaux pratiques Présentation et introduction à Ant (logiciel libre).

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.