QUALITÉ LOGICIELLE ET TESTS - LES FONDAMENTAUX

Durée	4 jours	Référence Formation	4-IT-QBAS

Objectifs

Comprendre la problématique de la complexité des développements logiciels
Comprendre le bénéfice de l'intégration continue pour améliorer la qualité des développements
Adopter les bonnes pratiques de rédaction des tests logiciels
Mettre en place une stratégie de campagne de tests automatisés avec les Framework et outils du marché
Travailler avec un gestionnaire de code source tel que Git, et adopter les réflexes du travail collaboratif
Comprendre le rôle des différents outils d'une chaîne d'intégration continue
Utiliser un outil de construction logiciel pour automatiser les tâches de développement
Mettre en place une plateforme d'intégration continue autour de Jenkins

Participants

Développeurs, chefs de projets, architectes logiciels

Pré-requis

Posséder les connaissances et compétences équivalentes aux fondamentaux du développement Java ou aux fondamentaux du développement .NET ou aux fondamentaux de la programmation orientée objet en C++

PROGRAMME

・Introduction
Pratiques d'ingénierie logicielle et méthodes Agiles.
Le développement incrémental et itératif.
L'équipe Agile. Scrum et XP.
・Les tests agiles
Définition et périmètre des tests agiles.
Cycle de développement : origine du TDD (Test Driven Development), ATDD, TDR, les types de tests...
・Les tests développeurs
Définition et objectifs : les patterns basiques XUnit.
Principe des tests unitaires automatisés.
Règles de simplicité : règle des "3 A" (Arrange, Act, Assert).
Mise en œuvre de tests unitaires avec JUnit, le framework de test en Java.
Lanceur de tests (TestRunner).
Les méthodes d'Assertions.
・Le TDD, développement guidé par les tests
Le cycle de développement.
Le principe du TDD : "test first", "tester, coder, refactorer".
TDD et pratiques agiles (XP) : l'intégration continue, le Pair Programming.
・"Refactoring", le remaniement de code
Principes du refactoring.
Réduire l'apparition de la dette technique, rendre le code compréhensible.
Comment identifier le code à risque ? La notion de "Code Smells", signes de danger potentiel.
Les principales opérations de refactoring.
Rappel sur les Design Patterns.
・Isolation des tests
Les doubles de test, leur utilisation.
Le "Mock Object" pour vérifier certaines hypothèses.

CAP ÉLAN FORMATION - Marseille – Toulon
www.capelanformation.fr - Tél : 04.86.01.20.50
Mail : contact@capelanformation.fr
Organisme enregistré sous le N° 76 34 0908834
[version 2022]

Le "Fake", pour la simulation.
Le "Stub" : fournir une réponse prédéfinie à un appel.
・Le test comme cahier des charges, la notion d'ATDD
Les principes et avantages de l'ATDD.
Du scénario au test de recette.
Combiner ATDD, BDD et TDD.
Les outils (Fitnesse, Cucumber...).
・Conclusions
Les bénéfices du TDD, le coût des tests.
Les autres types de tests (interface graphique, Web..).
Quelques outils.

Moyens pédagogiques

Accueil des stagiaires dans une salle dédiée à la formation équipée d'un vidéo projecteur, tableau blanc et paperboard ainsi qu'un ordinateur par participant pour les formations informatiques.
Positionnement préalable oral ou écrit sous forme de tests d'évaluation, feuille de présence signée en demi-journée, évaluation des acquis tout au long de la formation.
En fin de stage : QCM, exercices pratiques ou mises en situation professionnelle, questionnaire de satisfaction, attestation de stage, support de cours remis à chaque participant.
Formateur expert dans son domaine d'intervention
Apports théoriques et exercices pratiques du formateur
Utilisation de cas concrets issus de l'expérience professionnelle des participants
Réflexion de groupe et travail d'échanges avec les participants

Pour les formations à distance : Classe virtuelle organisée principalement avec l'outil ZOOM. Assistance technique et pédagogique : envoi des coordonnées du formateur par mail avant le début de la formation pour accompagner le bénéficiaire dans le déroulement de son parcours à distance.