



# Qualité logicielle et tests - Les fondamentaux

Mise à jour févr. 2025

**Durée** 4 jours (28 heures )

« Délai d'accès maximum 1 mois »

Nantes / Rennes : 2615 € HT

Brest / Le Mans : 2615 € HT

Certification : NON

## OBJECTIFS PROFESSIONNELS

- 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

## PRE-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++

## MOYENS PEDAGOGIQUES

- Réflexion de groupe et apports théoriques du formateur
- Travail d'échange avec les participants sous forme de 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.
- Remise d'un support de cours.

## MODALITES D'EVALUATION

- Feuille de présence signée en demi-journée,
- Evaluation des acquis tout au long de la formation,
- Questionnaire de satisfaction,
- Positionnement préalable oral ou écrit,
- Evaluation formative tout au long de la formation,
- Evaluation sommative faite par le formateur ou à l'aide des certifications disponibles,
- Sanction finale : Certificat de réalisation, certification éligible au RS selon l'obtention du résultat par le stagiaire

## MOYENS TECHNIQUES EN PRESENTIEL

- Accueil des stagiaires dans une salle dédiée à la formation, équipée d'ordinateurs, d'un vidéo projecteur d'un tableau blanc et de paperboard. Nous préconisons 8 personnes maximum par action de formation en présentiel

## MOYENS TECHNIQUES DES CLASSES EN CAS DE FORMATION DISTANCIELLE

- A l'aide d'un logiciel comme Teams, Zoom etc... un micro et éventuellement une caméra pour l'apprenant,
- suivez une formation uniquement synchrone 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-Entreprise 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. Nous préconisons 4 personnes maximum par action de formation en classe à distance

## ORGANISATION

- Les cours ont lieu de 9h à 12h30 et de 14h à 17h30.

## 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.

#### A L'ATTENTION DES PERSONNES EN SITUATION DE HANDICAP

- Les personnes atteintes de handicap souhaitant suivre cette formation sont invitées à nous contacter directement, afin d'étudier ensemble les possibilités de suivre la formation.

## Programme de formation

### Introduction (02h45)

- Pratiques d'ingénierie logicielle et méthodes Agiles
- Le développement incrémental et itératif
- L'équipe Agile. Scrum et XP

### Les tests agiles (01h45)

- Définition et périmètre des tests agiles
- Cycle de développement : origine du TDD (Test Driven Development), ATDD, TDR, les types de tests...

### Les tests développeurs (05h30)

- 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 (02h45)

- 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 (04h30)

- 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 (03h45)

- Les doubles de test, leur utilisation
- Le "Mock Object" pour vérifier certaines hypothèses
- Le "Fake", pour la simulation
- Le "Stub" : fournir une réponse prédéfinie à un appel

### Le test comme cahier des charges, la notion d'ATDD

#### (03h45)

- Les principes et avantages de l'ATDD
- Du scénario au test de recette.
- Combiner ATDD, BDD et TDD
- Les outils (Fitnesse, Cucumber...)

### Conclusions (02h45)

- Les bénéfices du TDD, le coût des tests
- Les autres types de tests (interface graphique, Web..)
- Quelques outils