

# Formation Développement Agile

## Appliquer l'Agilité au développement logiciel

Référence : DEV-AGILE-03

Durée : 3 jour(s)

### Présentation

Les méthodes Agiles sont aujourd'hui incontournables pour la gestion des projets informatiques. Étonnamment, alors que les rôles de référent métier et de responsable d'équipe technique sont largement détaillés, le rôle des membres de l'équipe de développement ne l'est que partiellement. En effet, seule l'organisation du travail est généralement traitée en omettant d'aborder les conséquences sur la pratique quotidienne du développement.

Quelles sont les spécificités d'un développeur Agile ?

Quels sont les impacts des méthodes Agiles sur le développement logiciel ?

Cette formation illustrera le Développement Agile au travers de trois aspects primordiaux : le Test Agile, l'Intégration Continue et la Conception Agile. Des travaux pratiques permettront d'appliquer ces principes sur une application Java/JEE réelle.

### Objectifs

- Comprendre l'impact des méthodes Agile sur le développement logiciel
- Mettre en place une politique de Test Agile pour augmenter la qualité du produit
- Elaborer une stratégie d'Intégration Continue pour limiter les risques du projet
- Comprendre la Conception Agile pour limiter la dette technique d'une application

**Répartition:** 40% Théorie, 60% Pratique

**Public:** Développeur, Responsable d'équipe, Architecte, Responsable QA

**Pré-requis:** Avoir participé à un projet informatique et avoir lu un ouvrage sur l'Agilité/Scrum ou avoir suivi la journée de formation "Découverte de l'Agilité" de Zenika.

### Programme

#### Jour 1 : Test Agile

- Automatisation des builds avec Jenkins et Maven

#### Test Driven Development

- Description de l'approche TDD et de ses bénéfices
- Comment améliorer la qualité et le design du code par les tests ?
- Les différents types de tests : unitaires, intégration, fonctionnels

#### Les tests unitaires avec JUnit

- Comprendre le cycle de vie d'un test dans le framework JUnit
- Respect du principe d'isolation et contrôle du résultat des tests
- Les mocks ou comment simuler le code périphérique au code testé
- Utilisation de frameworks de simulation de code comme EasyMock

#### Organisation des activités de test

- Méthodologie de correction d'un Bug
- Automatiser l'exécution de JUnit
- Génération de rapports
- Intégration avec Ant, Maven et Eclipse

#### Tests d'intégration

- Comprendre la notion de test in-container par opposition aux tests unitaires
- Test d'une couche de persistance avec DBUnit

#### Tests fonctionnels d'applications web avec Selenium

- Passage du test technique à un test de scénario fonctionnel
- Intégration avec le navigateur
- Création et exécution du script de test

#### Jour 2 : Intégration Continue

#### Présentation de l'Intégration Continue

- Principes, environnement et infrastructure
- Eviter les 5 risques liés à l'intégration
- Les types de processus d'intégration
- Les solutions d'Intégration continue : Hudson, Jenkins, CruiseControl, Bamboo, TeamCity, Apache Continuum, ...

#### Jenkins : la solution de référence du monde Open-Source

- Tableau de bord et santé des projets
- Architecture et principe d'extension
- Installation, configuration et utilisation de Jenkins

#### Métriques

- Les risques de dérive du code source
- Lien entre rapports de construction et qualité logicielle
- Respect des standards de développement avec Checkstyle et PMD
- Couverture de test avec Cobertura

#### Notification

- Intérêt des indicateurs visuels et bonnes pratiques
- Indicateurs de Jenkins et indicateurs étendus
- Historisation et suivi des tendances

#### Jour 3 : Conception Agile

#### Principes Objet fondamentaux

- Fondamentaux Objets
- Conception et développement Agile : SRP, DIP, ISP, OCP, LSP
- Gestion des dépendances


#### Conception de Frameworks

- Définition et principes des frameworks
- Quand choisir de créer un framework ?
- Principe d'extensibilité et mise en oeuvre

#### Refactoring

- Risque de « casse » du code
- Où ? Quand ? Comment refactorer ?
- Les 7 catégories d'outils de refactoring
- Best practises et limitations

## OFFERT EN INTER-ENTREPRISE

 Le petit déjeuner croissants, jus d'orange, café)

 Le déjeuner

 Une qualification téléphonique si nécessaire avec l'un de nos consultants

Tel: +33(0)1 45 26 19 15

Fax : +33(0)1 75 43 49 92

Email : training@zenika.com

### Auteur du cours

 zenika  
ARCHITECTURE INFORMATIQUE



Architecte logiciel, Agiliste convaincu et Scrum Master, Laurent Delvaux aide les décideurs et les équipes de développement à s'organiser pour augmenter leur productivité et maximiser les chances de réussite des projets. Il est membre du bureau du French Scrum User Group et intervient dans les écoles d'ingénieurs pour sensibiliser les étudiants aux pratiques Agiles.



zenika  
ARCHITECTURE INFORMATIQUE