

## Test Driven Developpement (TDD) Ingénierie Logicielle Agile

Mettre en œuvre le TDD, l'intégration continue et l'automatisation des tests

Mettre en oeuvre le TDD, l'intégration continue et l'automatisation des tests.

### Détails

- **Code** : UL-ILA
- **Durée** : 2 jours ( 14 heures )

#### Public

- Architectes
- Chefs de projets
- Développeurs

#### Pré-requis

- Expérience en testing

### Objectifs

- ?Appréhender le développement Agile basé sur des techniques modernes : TDD, intégration continue, automatisation des tests

### Programme

#### LE TEST EN AGILE

- Qualité et agilité
- Approche globale et pilotage par les tests
- TDD, BDD, ATDD, automatisation des tests, outils, reproductibilité
- Types de tests, démos

#### LES LIBRAIRIES XUNIT

- Déclarer un test
- Les assertions
- Préparation et nettoyage du test
- Organisation des tests

#### PRINCIPES ET BONNES PRATIQUES DU TDD

- Principes du TDD
- Objectifs et avantages
- Principe du Test First
- Émergence du design, à partir des tests

#### STYLES DE TDD

- Tests basés sur l'état ou le comportement
- Utilisation de doublures (Mocks, Stubs, Spy, Fakes...)

#### REFACTORING ET CODE TESTABLE

- SOLID
- Inversion de dépendances et découplage
- Polymorphisme
- Nommage
- Elimination de la duplication
- Code smells et refactoring

#### TDD ET CODE LEGACY

- Test et code intestable
- Stratégies de refactoring du code legacy
- L'intégration continue
- Le pair programming

### Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages