

# Fondamentaux des tests et de la recette applicative

Maîtriser les principes clés des tests logiciels et de la recette applicative pour garantir la qualité et la conformité des applications

Présentiel ou en classe à distance



2 jours (14 h)

Prix inter : 1.990,00 € HT  
Forfait intra : 4.390,00 € HT

Réf.: TE200

Les **tests logiciels** et la **recette applicative** occupent une place stratégique dans la réussite des projets numériques. Au-delà de la simple validation technique, ils garantissent la qualité, la conformité et la performance fonctionnelle des applications déployées. Cette formation apporte une vision claire et opérationnelle du rôle des tests dans le cycle de vie d'un projet, en abordant les bonnes pratiques de conception, d'exécution et de suivi des tests, tout en intégrant les enjeux de traçabilité et de gestion des risques.

Conçue pour renforcer la collaboration entre MOA, MOE, développeurs et testeurs, cette formation permet aux participants d'acquérir une méthode structurée pour **piloter efficacement la recette fonctionnelle**. Grâce à des mises en pratique et à l'utilisation d'outils tels que **Squash, Xray ou Jira**, elle offre une montée en compétence concrète sur la planification, la communication et la résolution des anomalies, dans une logique d'**amélioration continue de la qualité logicielle**.

## A qui s'adresse cette formation ?



### Pour qui

- Testeurs, développeurs, chefs de projet, analystes métier, responsables qualité, utilisateurs clés, membres de la MOA et MOE



### Prérequis

- Une connaissance générale du cycle de vie d'un projet informatique et des principes du développement applicatif est recommandée

## Programme

### 1 - Introduction aux tests logiciels et à la recette applicative

- Définitions clés : test, recette, validation, vérification
- Objectifs et enjeux des tests et de la recette
- Typologie des tests : unitaires, d'intégration, fonctionnels, techniques, d'acceptation
- Place des tests et de la recette dans le cycle de vie (cycle en V, Agile, DevOps)

Atelier

Quiz interactif

Echanges sur les pratiques des participants

### 2 - Méthodologies pour organiser les tests et recettes

- Acteurs impliqués : testeurs, MOA, MOE, développeurs, testeurs, utilisateurs
- Organisation des tests dans les projets : interne, fonctionnelle, utilisateur, homologation
- Méthodologies : approche classique, Agile, DevOps
- Niveaux de recette (interne, utilisateur, homologation)

Atelier

Identifier les rôles dans un projet fictif

Construire une matrice RACI simplifiée sur les tests d'un projet fictif

### 3 - Elaboration des tests et des plans de recette

- Processus de test : conception, exécution, validation, capitalisation
- Plan de test et plan de recette : contenu, structure, logique de couverture
- Rédaction de cas de tests : critères d'entrée/sortie, jeux de données, vérification Outils d'aide à la conception et à la documentation Bonnes pratiques de formalisation  
Atelier

À partir d'un cahier des charges simplifié, concevoir une grille de tests

Rédiger deux cas de test fonctionnels

### 4 - Exécution, traçabilité et outils

- Suivi des campagnes de tests
- Traçabilité des exigences
- Indicateurs qualité (taux de couverture, taux de réussite/échec)
- Introduction aux outils : Xray (Jira), Squash, TestLink, etc.  
Atelier

Exemple d'usage d'un outil de gestion de test

### 5 - Analyse des résultats et gestion des anomalies

- Typologie des anomalies : bloquantes, majeures, mineures
- Évaluation de la conformité et performance applicative
- Suivi des anomalies : cycle de vie, criticité, priorisation
- Outils de gestion d'anomalie : Jira, Mantis, Redmine  
Atelier

Analyser un jeu de résultats de tests et identification des anomalies critiques

### 6 - Gérer les risques et assurer la traçabilité

- Principes de gestion des risques applicatifs
- Risques applicatifs et projet : détection, anticipation, couverture
- Lien entre exigences, tests et conformité
- Traçabilité des exigences (matrice de couverture)
- Capitalisation des tests  
Atelier

Créer une matrice de traçabilité à partir de spécifications simples

### 7 - Outils de tests et de gestion de la recette

- Présentation d'outils du marché : Squash, TestLink, Xray (Jira), Azure Test Plans...
- Gestion des campagnes de tests et des anomalies
- Critères de choix d'un outil selon le contexte  
Atelier

Création de tests dans un outil

### 8 - Communication, collaboration

- Comprendre les enjeux de la communication entre MOA, MOE, testeurs et utilisateurs
- Identifier les points de friction et les risques liés à une mauvaise coordination
- Mettre en place des pratiques de collaboration efficaces : comités de suivi, outils partagés, feedback structuré
- Adopter des outils facilitant la transparence et le suivi (ex. : Jira, tableaux de bord collaboratifs)
- Favoriser une culture de l'échange pour résoudre plus vite les anomalies et garantir la qualité du livrable  
Atelier

Simulation d'un échange entre MOA et MOE pour valider un scénario de recette

### 9 - Tests logiciels et RSE

- Intégration de critères d'accessibilité dans les plans de tests fonctionnels.
- Ajout de critères d'éco-conception dans les grilles de tests (performance, efficacité énergétique, limitation des appels réseaux, optimisation des

## Après la session

- Un videocast sur la Gestion de tests logiciels : « Définir un périmètre de test : la stratégie », « Préparer les tests », « Exécuter les tests »
- Le référentiel général de l'écoconception des services numériques (RGESN)



### Les objectifs de la formation

- Comprendre le rôle des tests et des recettes dans le cycle de vie du développement logiciel
- Acquérir les méthodologies et les bonnes pratiques pour organiser les tests et recettes
- Savoir élaborer, exécuter et documenter des tests et des recettes fonctionnelles
- Analyser les résultats des tests et recettes pour en assurer la conformité et la performance
- Gérer les risques et garantir la traçabilité tout au long du processus de tests et recette
- S'initier à l'utilisation des outils de tests et de gestion de la recette
- Développer une communication efficace entre les équipes de développement, de tests et de recette



### Evaluation

- Pendant la formation, le formateur évalue la progression pédagogique des participants via des QCM, des mises en situation et des travaux pratiques. Les participants passent un test de positionnement avant et après la formation pour valider leurs compétences acquises.



### Les points forts de la formation

- L'apprentissage par la pratique : les phases théoriques sont complétées d'ateliers favorisant un ancrage durable des acquis
- Les nombreux retours d'expérience et conseils des consultants spécialistes du sujet L'utilisation d'environnements proches du contexte professionnel



## Dates et villes 2026 - Référence TE200



Dernières places disponibles



Session garantie

### Nantes

du 5 mars au 6 mars

du 18 juin au 19 juin

du 1 oct. au 2 oct.

### A distance

du 5 mars au 6 mars

du 18 juin au 19 juin

du 1 oct. au 2 oct.

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

### Lyon

du 5 mars au 6 mars

du 18 juin au 19 juin

du 1 oct. au 2 oct.

### Paris

du 5 mars au 6 mars

du 18 juin au 19 juin

du 1 oct. au 2 oct.

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

### Rennes

du 5 mars au 6 mars

du 18 juin au 19 juin

du 1 oct. au 2 oct.

### Bordeaux

du 5 mars au 6 mars

du 18 juin au 19 juin

du 1 oct. au 2 oct.

## Marseille

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

## Lille

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

## Rouen

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

## Aix-en-Provence

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

## Sophia Antipolis

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

## Strasbourg

du 29 avr. au 30 avr.

du 23 juil. au 24 juil.

du 12 nov. au 13 nov.

## Toulouse

du 29 avr. au 30 avr.

du 18 juin au 19 juin

du 1 oct. au 2 oct.