

## Guide des Tests Unitaires et Fonctionnels

<https://github.com/Antoine-Thomas/medit>

### Introduction :

Dans le domaine du développement web, la gestion des bugs est cruciale pour garantir la qualité et la fiabilité des sites web. Pour répondre à cette problématique, l'utilisation de tests unitaires et fonctionnels s'avère être une pratique efficace. Ce document vise à définir et expliquer ces deux types de tests, ainsi que leurs avantages et inconvénients.

### Définition d'un test unitaire :

Un test unitaire est une procédure automatisée permettant de vérifier le bon fonctionnement d'une unité de code isolée, telle qu'une fonction, une méthode ou une classe. Il s'agit de tester chaque composant logiciel de manière indépendante afin de détecter les éventuels bugs et assurer leur correction précoce.

### Définition d'un test fonctionnel :

Un test fonctionnel est une procédure automatisée qui évalue le comportement global d'une application ou d'une fonctionnalité du point de vue de l'utilisateur. Contrairement aux tests unitaires, les tests fonctionnels examinent l'application dans son ensemble, en simulant les interactions utilisateur pour vérifier si elle répond aux spécifications fonctionnelles.

### Projets adaptés aux tests unitaires :

Les tests unitaires conviennent particulièrement aux projets de développement logiciel où la modularité et la réutilisabilité du code sont importantes. Ils sont efficaces pour les applications complexes nécessitant une maintenance régulière et une évolutivité à long terme. Les frameworks modernes comme PHPUnit pour PHP ou Jest pour JavaScript sont couramment utilisés pour implémenter des tests unitaires.

### Projets adaptés aux tests fonctionnels :

Les tests fonctionnels sont appropriés pour les projets où l'accent est mis sur la garantie de la conformité aux spécifications fonctionnelles et à l'expérience utilisateur. Ils sont particulièrement utiles pour les applications web complexes comportant plusieurs interactions utilisateur et différentes fonctionnalités interdépendantes. Des outils tels que Postman pour les tests de navigateur ou Cypress pour les tests end-to-end sont souvent utilisés pour développer des tests fonctionnels.

## **5. Avantages et inconvénients :**

### **Tests unitaires :**

#### **Avantages :**

Détection des bugs au niveau du code dès leur apparition.  
Favorisent la modularité et la réutilisabilité du code.  
Permettent une validation rapide des changements effectués.

#### **Inconvénients :**

Nécessitent une mise en place initiale de la structure de test.  
Peuvent nécessiter des ressources importantes pour maintenir les tests à jour.

### **Tests fonctionnels :**

#### **Avantages :**

Garantissent la conformité aux spécifications fonctionnelles.  
Simulent le comportement réel de l'utilisateur.  
Identifient les problèmes d'intégration entre les différentes parties de l'application.

#### **Inconvénients :**

Requiert une configuration plus complexe et des ressources plus importantes.  
Peut être plus lent à exécuter en raison de son approche globale.

### **Conclusion :**

Les tests unitaires et fonctionnels sont des outils essentiels pour améliorer la qualité et la fiabilité des sites web. En combinant ces deux approches, les développeurs peuvent assurer une couverture complète de leurs applications, réduisant ainsi le nombre de bugs et garantissant une meilleure expérience utilisateur.