



- Niveaux de tests**
- 1) Besoin client <=> 4) Test d'acceptation
 - 2) Exigences <=> 3) Test système
 - 3) Conception <=> 2) Test d'intégration
 - 4) Code <=> 1) Test unitaire

TEST

TYPES

Statique : Revue de code, de spécifications, de documents de design

Dynamique : Exécuter le code pour s'assurer d'un fonctionnement correct

- 1) Exécuter le programme sur un ensemble fini de données d'entrées
- 2) Contrôler la correction des valeurs de sortie en fonction de ce qui est attendu

Cycle de vie du processus de test

- 1) Conception des tests ==> Cas de tests
- 2) Préparation des données de tests ==> Données de tests
- 3) Exécution du programme avec tests ==> Résultats de tests
- 4) Comparaison des résultats ==> Rapport de tests

Niveaux :

Niveau de détail (situation dans le cycle de vie)
Renseignent l'OBJET du test

Techniques
COMMENT on teste

Structurel (white box)

Basé sur le graphe de contrôle (toutes les instructions, toutes les branches, tous les chemins, ...)

Basé sur la couverture du flot de données (toutes les définitions de variables, toutes les utilisations, ...)

Fonctionnel (black box) : test de conformité par rapport aux specs

Tests fonctionnels

Valide les résultats rendus par les services

Tests non-fonctionnels

Valide la manière dont les services sont rendus

Tests de performance

Tests d'ergonomie

Tests nominaux / de bon fonctionnement

Vérifie que le résultat calculé est le résultat attendu, en entrant des données valides au programme (*test to pass*)

Tests de robustesse

Vérifie que le programme réagit correctement à une utilisation non conforme, en entrant des données invalides (*test to fail*)

Tests de sécurité

Types

Ce que l'on veut tester (renseignent la NATURE du test mené)

1) **Test de composants (unitaire)**

Test de procédures, de modules, de composants
(coût bug : 50% du coût total du développement initial correspondant)

2) **Test d'intégration des composants**

Test de bon comportement lors de la composition des procédures, modules ou composants
(coût d'un bug dans cette phase : 10 fois celui d'un bug unitaire)

3) **Test du système global**

Validation de l'adéquation aux spécifications
(coût d'un bug dans cette phase : 100 fois celui d'un bug unitaire)

4) **Test d'acceptation (recette)**

Mêmes remarques que pour test système