



Test du Logiciel (Certification ISTQB)

Chapitre I: Introduction au Test des Logiciels

Support de cours de:
Wafa Boussellaa

Filières : LCE2_IOT ; LCS2

Année : 2023-2024

[Plan de Cours]

Chapitre I : Introduction au Test des Logiciels

Chapitre II : Le test structurel

Chapitre III : Le test fonctionnel

[Chapitre I. Introduction au Test des Logiciels]

1. Introduction
2. Définition du Test
3. Difficultés du Test
4. Quelques Notions Fondamentales
5. Classification des tests selon la phase du cycle de vie
6. Les Techniques de test
7. Outils de Test
8. La certification en Test logiciel
9. Bibliographie

[Introduction (Les bugs logiciels)]

Motivations du test → Coûts d'un Bug logiciel

- En 1990 : Réseau téléphonique AT&T bloqué durant 9h.
 - Problème dans un logiciel installé sur 114 routeurs.
 - bloqué 5 Millions d'appels à travers les USA.
- En 1996 : la fusée Ariane 5 explose après 30s de vol.
 - Erreur de conversion entre données numériques
 - \$500M en matériel + \$7B en développement
- En 2000 : « Bug de l'an 2000 »
 - format de date sur deux chiffres (année 2000 codée par 00)
 - 300, 1600, ou même 5000 Milliards de \$ de coûts.
- Etc..

[Introduction (Les bugs logiciels)]

Remarquons que les bugs cités sont provoqués suite à des erreurs

Erreur
Spécification, conception, programmation



Défaut dans le logiciel



Anomalie de fonctionnement

La chaîne causale de l'erreur est donc :

Erreur → Défaut → Anomalie (ou dites défaillance)

Objectif : S'assurer de la qualité des logiciels

Introduction (Les Méthodes de Vérification et de Validation V&V)

Deux aspects de la notion de qualité :

➤ Conformité avec la définition : **VALIDATION**

- Réponse à la question :

Faisons-nous le bon produit ? ;

Est-ce que le logiciel réalise les fonctions attendues ?;

- Contrôle en cours de réalisation, le plus souvent avec le client
- **Défauts** par rapport aux besoins que le produit doit satisfaire

➤ Correction d'une phase ou de l'ensemble : **VERIFICATION**

- Réponse à la question :

Faisons-nous le produit correctement ?;

Est-ce que le logiciel fonctionne correctement ?;

- Tests
- **Erreurs** par rapport aux définitions précises établies lors des phases antérieures de développement

[Définition du Test]

Définition 1 : Selon l'IEEE (Standard Glossary of Software Engineering Terminology ; « **Le test est l'exécution ou l'évaluation d'un système ou d'un composant par des moyens automatiques ou manuels, pour vérifier qu'il répond à ses spécifications ou identifier les différences entre les résultats attendus et les résultats obtenus** »).

Définition 2 : Selon G. J. Myers (The art of Software Testing)
« **Tester, c'est exécuter le programme dans l'intention d'y trouver des anomalies ou des défauts** ».

Difficultés du Test

Le test logiciel est la méthode la plus populaire pour vérifier un logiciel. Cette méthode représente environ 40-60% du coût total du projet.

Difficultés associées au processus d'introduction des défauts

Il s'agit du processus de transformations successives réalisé lors des étapes de la démarche de programmation provoquant la perte des informations et l'introduction des erreurs

Difficultés d'ordre psychologique ou « culturel »

la raison principale est que, pour améliorer la qualité d'un programme, le test passe par une étape destructive, car un bon test est de trouver une erreur et la programmation est un processus constructif.

Difficultés formelles

le test exhaustif est impossible à réaliser, car il n'existe pas un algorithme général capable de démontrer l'exactitude d'un programme. C'est le problème de l'indécidabilité de la correction d'un programme. Selon E. W. Dijkstra "Testing can reveal the presence of errors but never their absence".

Évolution de la perception du test dans le temps

Années 50

test = mise au point

1960 - 1980

test = «démonstration» (preuve de correction)

1980

test = «destruction»

« détruire » = trouver des défauts

Aujourd'hui

test = «évaluation»

moyen d'évaluation de la qualité

[Quelques Notions Fondamentales]

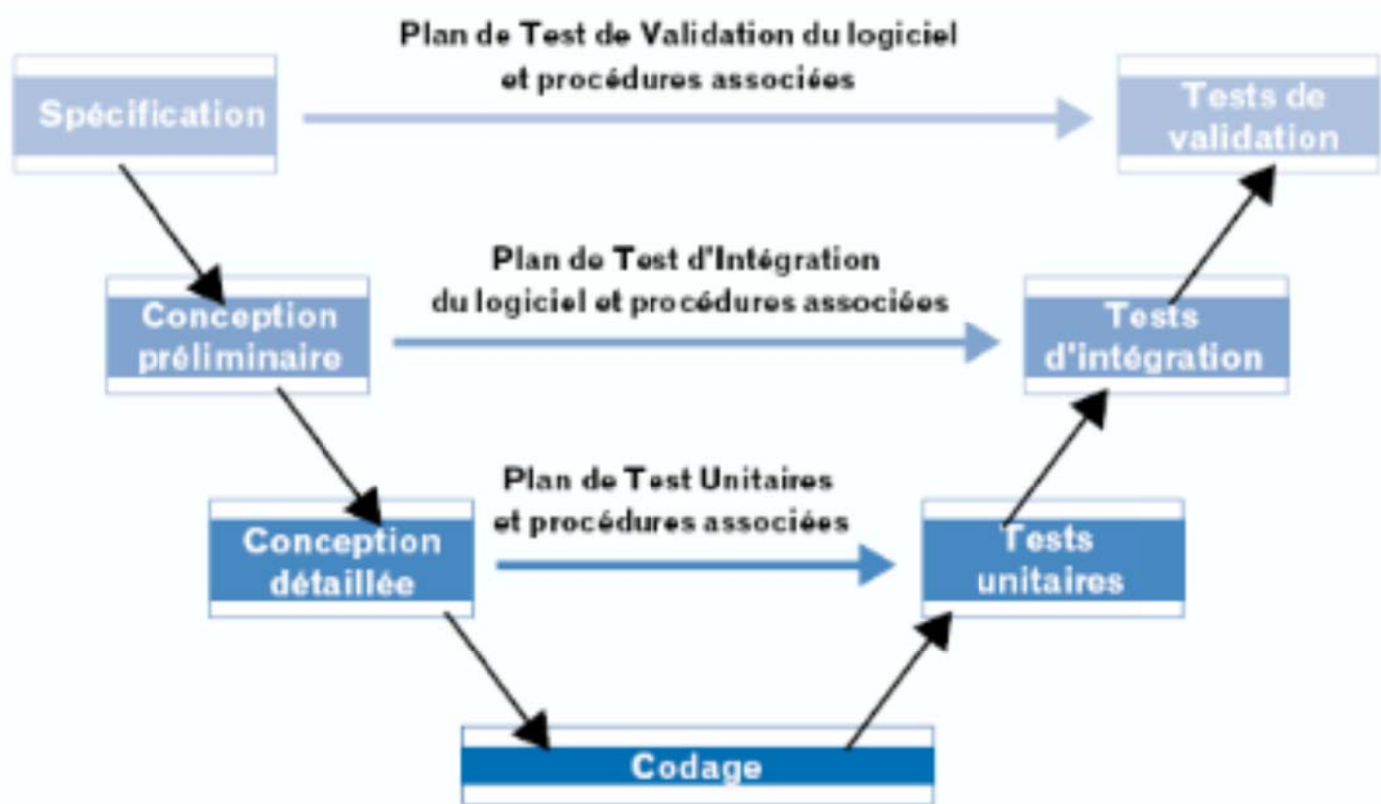
Données de test (DT) : Ensemble d'entrées définies à fournir au logiciel lors d'une exécution. L'ensemble de données de test constitue *un jeu de test*. Pour soumettre ces jeux de test, une séquence d'actions doit être effectuée. On note souvent une DT en utilisant une notation ensembliste (Exemple : $DT1 = \{x=5, y=5, b=56\}$)

Scénario de test : Un scénario de test est une séquence d'actions nécessaire à l'exécution du jeu de test.

Oracle automatique de test : *Outil* qui évalue les résultats produits par le logiciel d'une manière automatique, en comparant les résultats du test avec les spécifications (résultats souhaités) du logiciel.

Classification des tests selon la phase du cycle de vie

Les cas de tests sont élaborés (plans de test) lors de la partie descendante d'un cycle en V typique, en parallèle des phases de spécification, conception, et de codage.



Classification des tests selon la phase du cycle de vie

Les tests unitaires : Test d'un programme ou d'un module isolé dans le but de s'assurer qu'il ne comporte pas d'erreur d'analyse ou de programmation.

Les tests d'intégration : Tester les interfaces entre les composants. C'est au cours de cette phase que sont généralement pris en compte d'éventuels composants logiciels qui n'ont pas été développés dans le cadre du projet.

Les tests de validation ou conformité : Validation de l'adéquation aux spécifications (s'assurer que le logiciel réalise effectivement tout ce que le client est en droit d'en attendre).

Les Techniques de test

Le Test fonctionnel ou « Test boîte noire » : Technique de test où les cas de tests sont élaborés à partir d'une analyse des spécifications des fonctionnalités d'un composant ou système sans référence à sa structure interne.

Les méthodes de test :

- Analyse partitionnelle
- Test aux limites
- Graphes cause-effet
- Tests syntaxiques
- Tests aléatoires

[Les Techniques de test]

Le Test structurel ou « Test boîte blanche » ou « test en verre » :

Technique de test où les cas de tests sont basés sur une analyse de la structure interne d'un composant ou système (le code source). Deux classes de tests existent :

- **Test structurel statistique :** revues de code, estimation de la complexité, etc.
- **Test structurel dynamique :**
- Basés sur le graphe de flot de contrôle (toutes les instructions, toutes les branches, tous les chemins, ...)
- Basés sur la couverture du flot de données (toutes les définitions de variable, toutes les utilisations, ...)

[Les outils de test]

Les outils de test automatique ou « oracle de test » sont classés selon le cas de test en question (fonctionnel ou structurel).

Exemples :

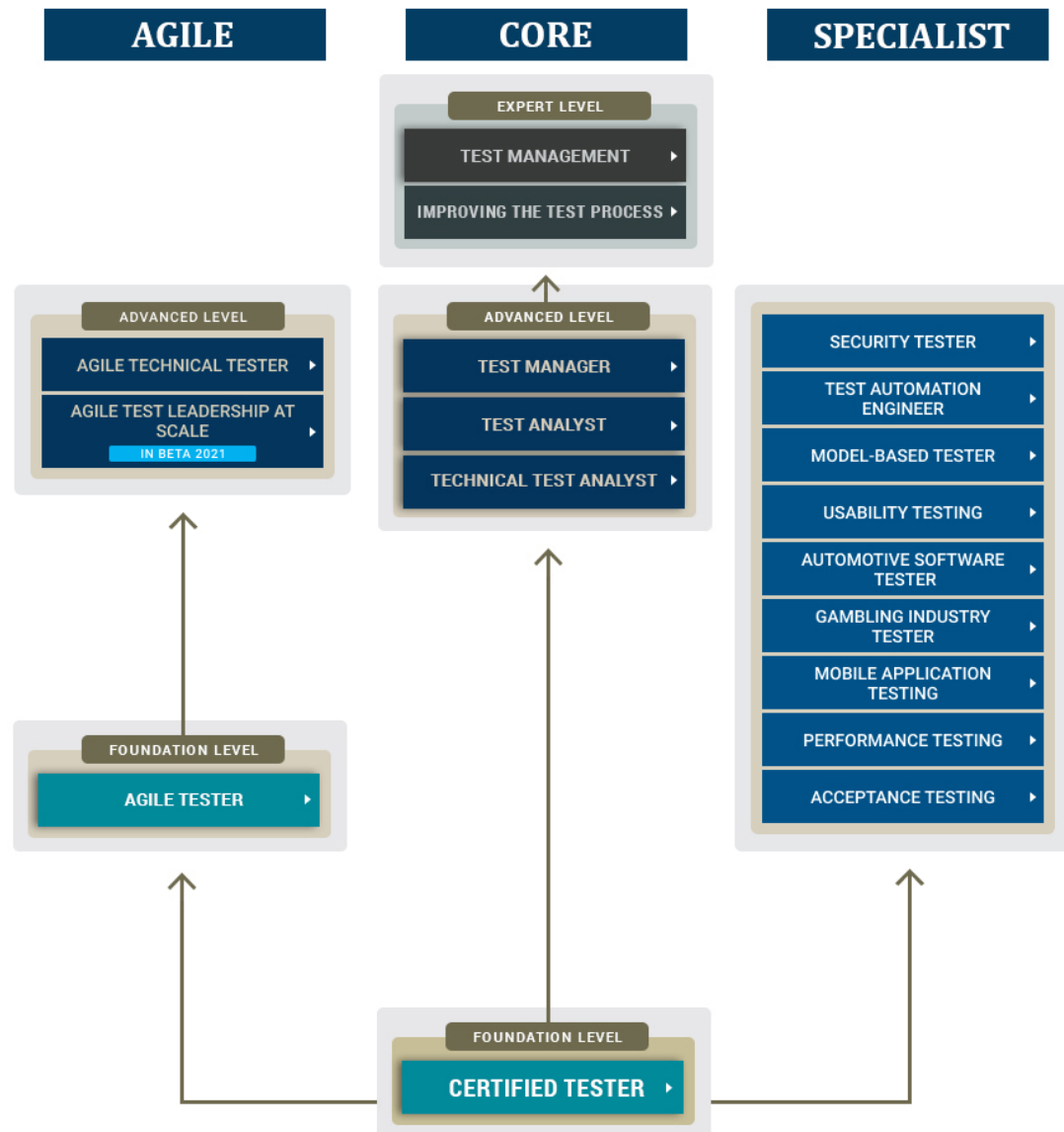
- ♦ **Test Designer** : outil qui génère automatiquement des tests depuis un modèle UML.
- ♦ **Junit** : oracle de test pour les programmes JAVA.
- ♦ **CPPunit et DeepCover** : oracle de test pour les programmes C++.

[La certification en Test Logiciel (ISTQB)]

- L'ISTQB (International Software Testing Qualifications Board) est une association internationale. Son but est de mettre en place, au niveau international, un schéma uniforme et universel de connaissances et compétences nécessaires à tout professionnel concerné par les tests de logiciel et de système.
- L'ISTQB offre un niveau fondation de qualification pour toutes les personnes impliquées dans les tests de logiciels. Ses personnes peuvent avoir plusieurs rôles : testeurs, analystes de tests, ingénieurs de tests, consultants en tests, Test Manager, testeurs en phase d'acceptation utilisateur et développeurs de logiciels.
- Grâce à cette standardisation, recruter un testeur certifié ISTQB est devenu pour les entreprises une garantie de connaissance, de compétence, de professionnalisme et d'éthique.

Parcours de la certification (ISTQB)

Parcours principal appelé CORE, à trois niveaux: fondation, avancé et expert. Parcours AGILE pour les spécialistes des projets agiles. un troisième parcours SPECIALIST pour les tests spécialisés sur un secteur d'activité, un type de test pour un support mobile. Le point d'entrée à tous ces parcours est le niveau fondation.



Bibliographie

Pratique des tests logiciels (Dunod, 2009)

J. Printz, J.-F. Peyre.

Le test des logiciels (Hermès, 2000)

S. Xanthakis, P. Régnier, C. Karapoulis.

*Tout sur le test logiciel : Préparation à la certification
ISTQB/Professeur Testeur (ellipses)*

Marc Hage Chahine.