

Introduction à l'équivalence des classes de test

Tous les testeurs sont confrontés à la problématique de la conception des cas de test, que ces derniers soient manuels ou automatisés.

Cette activité de conception peut être faite de plusieurs manières différentes.

- celle basée sur les spécifications, aussi appelée boîte noire;
- celle basée sur la structure, aussi appelée boîte blanche (ou boîte en verre);
- celle basée sur l'expérience.

La technique des partitions d'équivalence appartient à la catégorie basée sur les spécifications.

L'équivalence des classes de test est une technique essentielle dans le domaine de l'assurance qualité logicielle. Elle permet de réduire le nombre de cas de test tout en s'assurant de couvrir l'ensemble des scénarios possibles. Cette approche se concentre sur l'identification de groupes d'entrées similaires qui produiront des résultats identiques, appelés classes d'équivalence.

En maîtrisant ces principes, les équipes de développement peuvent concevoir des jeux de test plus efficaces et optimiser considérablement le processus de validation.



Définition et principes de l'équivalence des classes

1 Définition

L'équivalence des classes est une technique de test qui consiste à diviser l'ensemble des entrées possibles d'un programme en groupes distincts, appelés classes d'équivalence. Chaque groupe représente un ensemble d'entrées qui devraient produire des résultats identiques. Cette approche permet de réduire le nombre de cas de test nécessaires tout en garantissant une couverture complète des scénarios.

2 Principes clés

Les principes de l'équivalence des classes reposent sur l'identification des conditions d'entrée valides et non valides, ainsi que sur la sélection d'un nombre représentatif de cas de test pour chaque classe. L'objectif est de couvrir l'ensemble des scénarios possibles tout en minimisant le nombre de cas de test à exécuter. Cela permet de gagner du temps et des efforts lors de la conception et de l'exécution des tests.

3 Avantages

Cette approche permet de réduire considérablement le temps et les efforts nécessaires à la conception et à l'exécution des tests, tout en améliorant la couverture et la qualité globale du système testé. En identifiant les classes d'équivalence et en sélectionnant judicieusement les cas de test, il est possible d'obtenir une validation efficace et complète du logiciel.

Application de la technique

- applicable à tous les niveaux de test: unitaire, intégration, système et acceptation;
- adaptée lorsque tous les éléments d'un même ensemble de valeurs sont supposés être traités de la même façon. Notons que l'ensemble peut être ordonnée ou non:
 - exemple d'ensemble ordonné: une application prenant en entrée l'âge d'un individu.
 - exemple d'ensemble non ordonné: une application prenant en entrée des formes géométriques de différentes dimensions → Carré, Cercle, Triangle, Rectangle, ...
- pertinente pour des tests fumigatoires sur une nouvelle version ou une nouvelle livraison afin de rapidement déterminer si les fonctionnalités de base fonctionnent;
- peut être utilisée pour déterminer des ensembles de données pour tester la performance de l'application.

Avantages de l'équivalence des classes

Réduction des Efforts de Test

L'équivalence des classes permet de réduire considérablement le nombre de cas de test à concevoir et à exécuter, en se concentrant sur un nombre représentatif de scénarios qui couvrent l'ensemble des possibilités. Cela se traduit par des économies de temps et de ressources significatives.

Amélioration de la Couverture

En identifiant les classes d'équivalence, les équipes de test s'assurent de couvrir l'ensemble des scénarios possibles, réduisant ainsi le risque de laisser passer des défauts critiques.

Meilleure Compréhension du Système

Le processus d'identification des classes d'équivalence oblige les équipes à mieux comprendre le fonctionnement du système, ce qui se traduit par une meilleure qualité des tests et une réduction des défauts.

Limites de la technique

Une des difficultés est de correctement choisir les partitions. Si au sein d'une partition, toutes les données ne sont pas traitées de manière identiques, il y a un risque de laisser passer des défauts.

Une limitation de la technique est de choisir une donnée au hasard dans la partition et considérer que toutes les autres ne poseront pas de problème. C'est un risque à considérer comme pour toutes réductions de couverture de tests. Le second article de cette série qui se focalisera sur l'analyse des valeurs limites permettra de réduire ce risque.



Identification des classes d'équivalence

1

Analyse des Exigences

La première étape consiste à analyser attentivement les exigences du système pour identifier les différents paramètres d'entrée et leurs conditions valides et non valides.

2

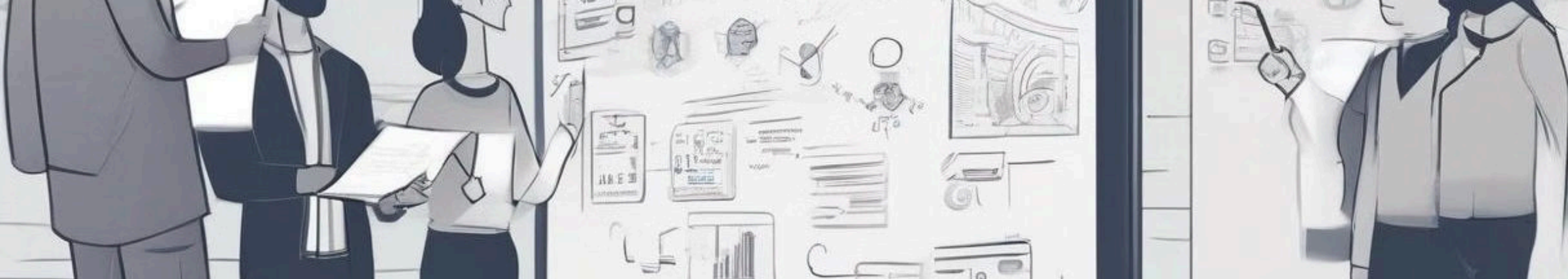
Définition des Classes

À partir de cette analyse, les équipes de test définissent les classes d'équivalence en regroupant les entrées qui devraient produire des résultats identiques.

3

Validation des Classes

Enfin, les classes d'équivalence identifiées sont validées pour s'assurer qu'elles couvrent l'ensemble des scénarios possibles et qu'aucune entrée n'est oubliée.



Conception détaillée des cas de test

1

Sélection des Cas de Test

Une fois les classes d'équivalence définies, les équipes de test sélectionnent un ou plusieurs cas de test représentatifs pour chaque classe. L'objectif est de couvrir un large éventail de scénarios valides et non valides. Ces cas de test serviront de base pour évaluer la qualité et le bon fonctionnement du logiciel.

2

Conception des Cas de Test

Les cas de test sont alors conçus en détail, en spécifiant les entrées, les résultats attendus et les étapes à suivre pour l'exécution des tests. Chaque cas de test doit être clairement défini, en incluant toutes les informations nécessaires pour un testeur de comprendre et d'exécuter les tests de manière cohérente.

3

Validation des Cas de Test

Enfin, les cas de test sont soumis à une validation rigoureuse pour s'assurer qu'ils couvrent correctement les classes d'équivalence identifiées. Les testeurs vérifient également que les cas de test permettent de détecter les défauts potentiels et de valider le bon fonctionnement du logiciel dans différents scénarios.

Exécution et évaluation des tests

Exécution des Tests

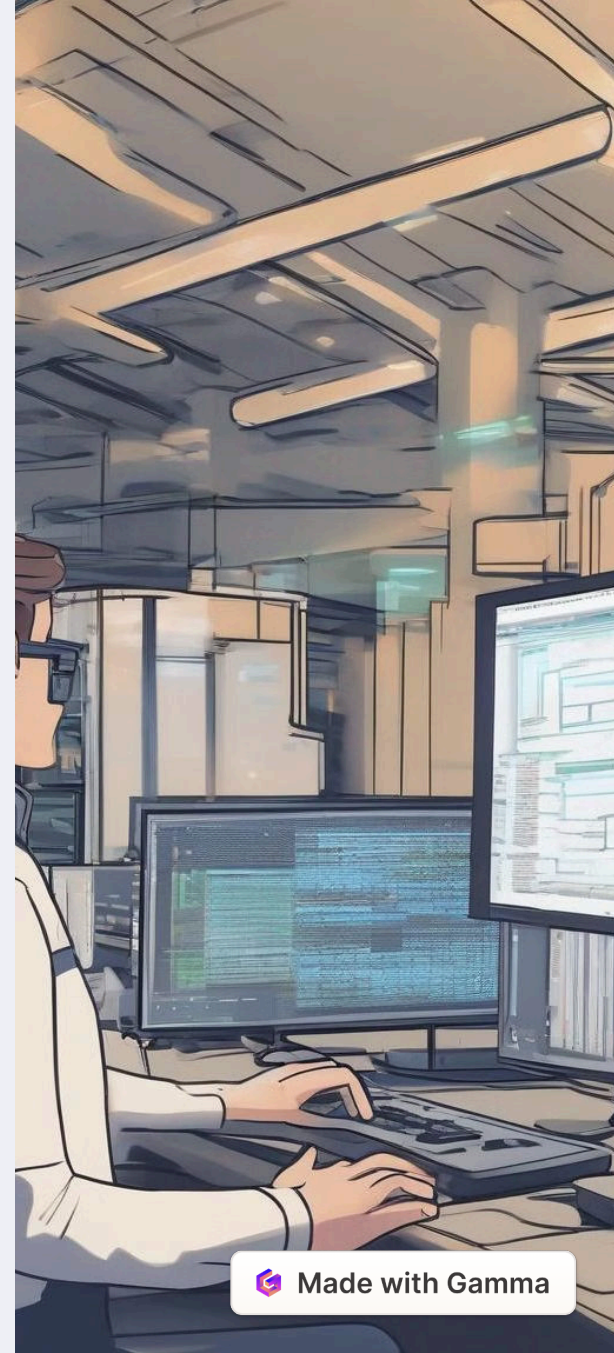
Les cas de test conçus à l'étape précédente sont alors exécutés, en s'assurant de suivre scrupuleusement les étapes définies et en enregistrant soigneusement les résultats.

Évaluation des Résultats

Les résultats des tests sont ensuite évalués, en comparant les résultats obtenus aux résultats attendus. Toute divergence est considérée comme un défaut potentiel à investiguer.

Rapport de Test

Un rapport de test détaillé est alors généré, incluant les résultats, les défauts détectés et les recommandations pour la suite du processus de test.



Gestion des résultats et des défauts

Analyse des Défauts	Les défauts détectés lors des tests sont analysés en détail pour en comprendre les causes et les impacts potentiels.
Priorisation des Défauts	Les défauts sont alors priorisés en fonction de leur gravité et de leur impact sur le système, afin de guider les efforts de correction.
Suivi des Corrections	Les équipes de développement corrigent les défauts identifiés, et les équipes de test vérifient que les corrections sont efficaces.
Retour d'Expérience	Enfin, un retour d'expérience est effectué pour identifier les points forts et les axes d'amélioration du processus de test par équivalence des classes.
couverture	<p>La couverture est calculée en prenant le nombre de partitions pour lesquelles une valeur a été testée et en le divisant par le nombre de partitions définies.</p> <p>Utiliser plusieurs valeurs pour une même partition n'augmente pas le pourcentage de couverture.</p>

Cette technique est particulièrement efficace pour pointer des erreurs de spécifications: omissions, ambiguïtés, contradictions, manques de précision,

Elle sera d'une grande aide pour choisir des valeurs dans chaque partition qui seront ensuite utilisées dans d'autres techniques traitant de la combinatoire (Tables de décision, pairwise , classification arborescente entre autres).

Il convient de ne pas oublier que réduire la combinatoire impose de gérer le risque qualité associée.

Liaison avec les autres méthodes de test

Approche Complémentaire

L'équivalence des classes de test peut être utilisée en complément d'autres techniques comme les tables de décision ou l'analyse par arborescence de classification. Chaque méthode apporte une perspective unique pour une couverture de test optimale.

Gestion des Risques

Bien que réduire la combinatoire des tests soit un objectif, il faut veiller à ne pas compromettre la qualité. L'équivalence des classes doit donc être associée à une analyse rigoureuse des risques.

Automatisation des Tests

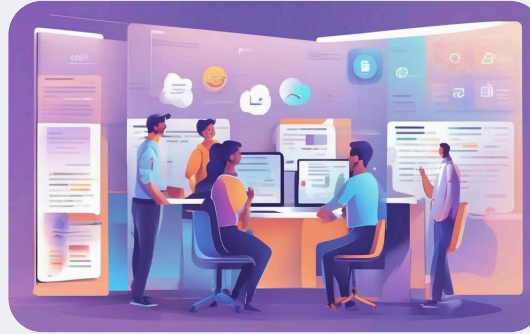
L'automatisation des tests basés sur l'équivalence des classes permet d'optimiser les cycles de régression et d'améliorer l'efficacité globale du processus de validation.

Outils et Utilisateurs de l'Équivalence des Classes de Test



Outils Populaires

Les équipes de test utilisent une variété d'outils pour appliquer l'équivalence des classes, tels que des gestionnaires de cas de test, des outils d'automatisation et des tableaux de bord d'analyse des résultats.



Collaboration Interdisciplinaire

L'équivalence des classes nécessite une étroite collaboration entre les analystes, les développeurs et les testeurs afin d'identifier les partitions pertinentes et de concevoir des cas de test efficaces.



Rôles Clés

Les principaux utilisateurs de l'équivalence des classes sont les responsables qualité, les chefs de projet et les équipes de test qui pilotent la conception, l'exécution et l'analyse des tests.

Conclusion et meilleures pratiques



Réflexion Constante

L'équivalence des classes est un processus itératif qui nécessite une réflexion constante sur les exigences, les scénarios de test et les résultats obtenus.



Collaboration Interdisciplinaire

La réussite de l'équivalence des classes repose sur une collaboration étroite entre les équipes de développement, de test et d'analyse des exigences.



Amélioration Continue

L'analyse des résultats et des défauts permet d'identifier les points à améliorer dans le processus de test, afin de le rendre toujours plus efficace.



Automatisation des Tests

L'automatisation des tests basés sur l'équivalence des classes peut considérablement optimiser le processus de validation et de regression.

NIKOUM MODESTE LORENE

21T2580

SIBAFO WISDOM

21T2915

TEGOMO STEPHANE

0000

EWONDJO JOSEPH

21T2332

**FANDJA DE TCHOUA
ANNAELLE**

21T2311

POSSEU RUTH

22W2286