

Chapitre 5: Organisation des tests

Les degrés d'indépendance allant d'un niveau d'indépendance faible à un niveau élevé

1. Pas de testeurs indépendants, les développeurs testent leur propre code

2. Des développeurs ou des testeurs indépendants au sein des équipes de développement ou de l'équipe de projet

3. Une équipe de test indépendante ou un groupe de testeurs au sein de l'organisation

4. Des testeurs indépendants appartenant à l'organisation métier ou à la communauté d'utilisateurs, ou spécialisés dans des types de tests spécifiques

5. Des testeurs indépendants externes à l'organisation, travaillant soit sur site (insourcing) soit hors site (outsourcing)

Les avantages potentiels de l'indépendance	Les inconvénients potentiels de l'indépendance
<ul style="list-style-type: none"> • Les testeurs indépendants sont susceptibles de détecter des types de défaillances différents par rapport aux développeurs. • Un testeur indépendant peut vérifier, contester ou réfuter les hypothèses formulées par les parties prenantes. 	<ul style="list-style-type: none"> • Les testeurs peuvent être isolés de l'équipe de développement • Les développeurs peuvent perdre le sens des responsabilités vis-à-vis de la qualité • Les testeurs indépendants peuvent ne pas disposer de certaines informations importantes.

Les tâches du Test Manager	Les tâches des testeurs
<ul style="list-style-type: none"> • Une politique de test et une stratégie de test, plan de test • Préparer et fournir des rapports d'avancement des tests et des rapports de synthèse des tests • Lancer l'analyse, la conception, l'implémentation et l'exécution des tests • Gestion de configuration • Les métriques • La sélection et la mise en œuvre des outils • Décider de l'implémentation du ou des environnements de test • Développer les compétences et les carrières des testeurs 	<ul style="list-style-type: none"> • Revoir et contribuer aux plans de test • Analyser, revoir et évaluer les exigences • les conditions de test, et saisir la traçabilité entre les cas de test, les conditions de test et les bases de test • Concevoir, configurer et vérifier le ou les environnement(s) de test • Exécuter des tests • Automatiser les tests • Les tests non-fonctionnelles • Revoir des tests développés par d'autres

1. Planification et estimation des tests

Objectif :

- Un plan de test décrit les activités de test pour des projets de développement et de maintenance.
- Au fur et à mesure que le projet et la planification des tests progressent, davantage d'informations deviennent disponibles et plus de détails peuvent être inclus dans le plan de test.
- La planification des tests est une activité continue et s'effectue tout au long du cycle de vie du produit.
- Des retours d'information sur les activités de test devraient être utilisés pour identifier les risques qui évoluent afin que la planification puisse être ajustée
- La planification peut être documentée dans un plan de test maître et dans des plans de test séparés pour des niveaux de test, tels que les tests système et les tests d'acceptation, ou pour des types de test séparés.

Les activités de planification des tests:

- Déterminer le périmètre, les objectifs et les risques des tests
- Définir l'approche générale des tests
- Intégrer et coordonner les activités de test dans les activités du cycle de vie du logiciel
- Prendre des décisions sur ce qu'il faut tester, les personnes et les autres ressources nécessaires pour effectuer les diverses activités de test et la façon dont les activités de test seront effectuées
- Planifier les activités d'analyse, de conception, d'implémentation, d'exécution et d'évaluation des tests
- Sélectionner les métriques pour le pilotage et le contrôle des tests
- Budgéter les activités de test
- Déterminer le niveau de détail et la structure de la documentation des tests

Stratégie de test et approche de test:

Une stratégie de test fournit une description générale du processus de test, généralement au niveau du produit ou de l'organisation.

Stratégie & approche de test	Description
Analytique	<ul style="list-style-type: none"> • Ce type de stratégie de test est basé sur l'analyse d'un facteur (p. ex., exigences ou risques). • Le test basé sur les risques est un exemple d'approche analytique, où les tests sont conçus et priorisés en fonction du niveau de risque.
Basée sur des modèles	<ul style="list-style-type: none"> • Dans ce type de stratégie de test, les tests sont conçus à partir d'un modèle d'un aspect requis du produit, comme une fonction, un processus métier, une structure interne ou une caractéristique non-fonctionnelle (par exemple, la fiabilité). • Les modèles de processus métier, les modèles d'état et les modèles de fiabilité en sont des exemples.
Méthodique	<ul style="list-style-type: none"> • Ce type de stratégie de test repose sur l'utilisation systématique d'un ensemble prédéfini de tests ou de conditions de test, tels qu'une taxonomie des types de défaillances les plus courantes ou probables, une liste des caractéristiques de qualité importantes.
Conforme à un processus	<ul style="list-style-type: none"> • Ce type de stratégie de test implique l'analyse, la conception et l'implémentation de tests basés sur des règles et normes externes, telles que celles spécifiées par des normes spécifiques à l'industrie.
Dirigée (ou consultative)	<p>Ce type de stratégie de test est principalement dicté par les recommandations, les conseils ou les instructions des parties prenantes, des experts du domaine métier ou des experts techniques, qui peuvent se trouver en dehors de l'équipe de test ou de l'organisation elle-même.</p>
Anti-régressions	<ul style="list-style-type: none"> • Ce type de stratégie de test est motivé par le désir d'éviter la régression au niveau des fonctionnalités existantes. • Cette stratégie de test comprend la réutilisation des testware existants, une forte automatisation des tests de régression et des suites de test standard.
Réactive	<ul style="list-style-type: none"> • Dans ce type de stratégie de test, le test est adapté aux composants ou systèmes testés en réaction aux événements se produisant pendant l'exécution des tests, plutôt que préplanifié • Les tests sont conçus et implémentés, et peuvent être exécutés immédiatement en réponse aux informations obtenues à partir des résultats de tests antérieurs. • Les tests exploratoires sont une technique couramment employée dans les stratégies réactives.

Les critères d'entrée	<ul style="list-style-type: none"> • Disponibilité d'exigences testables, d'User Stories et/ou de modèles • Disponibilité d'éléments de test qui ont satisfait aux critères de sortie pour tous les niveaux de test précédents • Disponibilité de l'environnement de test. • Disponibilité des outils de test nécessaires. • Disponibilité des données de test et autres ressources nécessaires.
Les critères de sortie	<ul style="list-style-type: none"> • Les tests planifiés ont été exécutés • Un niveau défini de couverture a été atteint • Le nombre de défauts non résolus est limité à ce qui a été défini • Le nombre estimé de défauts restants est suffisamment faible • Les niveaux évalués de qualité pertinentes sont suffisants
Calendrier d'exécution des tests	<ul style="list-style-type: none"> • Idéalement, les cas de test devraient être exécutés en fonction de leur niveau de priorité • Si un cas de test avec une priorité plus élevée dépend d'un cas de test avec une priorité plus faible, le cas de test de priorité plus faible doit être exécuté en premier.
Facteurs influençant l'effort de test	<p>Caractéristiques du produit: • Les risques associés au produit • La qualité des bases de test</p> <ul style="list-style-type: none"> • La taille du produit • Les exigences relatives aux caractéristiques de qualité • Le niveau de détail requis pour la documentation des tests. <p>Caractéristiques du processus de développement : • La stabilité et la maturité de l'organisation</p> <ul style="list-style-type: none"> • Le modèle de développement utilisé • L'approche de test • Les outils utilisés • Le processus de test • La pression des délais <p>Caractéristiques liées aux personnes: • Les compétences et l'expérience des personnes impliquées, en particulier avec des projets et des produits similaires</p> <ul style="list-style-type: none"> • La cohésion et le leadership de l'équipe <p>Résultats des tests: • Le nombre et la sévérité des défauts trouvés</p> <ul style="list-style-type: none"> • La quantité nécessaire de travail à refaire
Techniques d'estimation des tests	<p>La technique basée sur les métriques: • estimer l'effort de test sur la base de métriques d'anciens projets similaires ou sur la base de valeurs représentatives</p> <p>La technique basée sur l'expertise: • estimation de l'effort de test sur la base de l'expérience des responsables des tâches de test ou par des experts.</p>

2. Pilotage et contrôle des tests

L'objectif :

- Le but du pilotage des tests est de recueillir des informations et de fournir un retour et de la visibilité sur les activités de test.
- Le contrôle des tests décrit toutes les mesures correctives ou d'orientation prises à la suite de l'information et des métriques recueillies et (éventuellement) rapportées.

Exemples d'actions de contrôle des tests :

- Re-prioriser les tests lorsqu'un risque identifié se produit
- Modifier le calendrier des tests en raison de la disponibilité ou de l'indisponibilité d'un environnement de test ou d'autres ressources
- Réévaluer si un élément de test répond à un critère d'entrée ou de sortie à cause d'une modification

Métriques utilisées pour les tests	Description
Les métriques de test les plus courantes	<ul style="list-style-type: none">• Pourcentage du temps de travail prévu réalisé pour la préparation des cas de test (ou le pourcentage des cas de test prévus implémentés)• Exécution des cas de test (p. ex. nombre de cas de test exécutés/non exécutés, cas de test réussis/échoués, et/ou conditions de test réussies/échouées)• Informations sur les défauts• Couverture des exigences, des User Stories, des critères d'acceptation, des risques ou du code• Degré d'achèvement des tâches, affectation et utilisation des ressources, et temps passé, Coût des tests
Rapport de test	<ul style="list-style-type: none">• Le but du rapport de test est de synthétiser et de communiquer les informations sur l'activité de test, à la fois pendant et à la fin d'une activité de test• Le rapport de test préparé pendant une activité de test peut être appelé rapport d'avancement de test, tandis qu'un rapport de test préparé à la fin d'une activité de test peut être appelé rapport de synthèse de test. <p>Les rapports d'avancement de test peuvent également inclure les éléments suivants:</p> <ul style="list-style-type: none">• Le statut des activités de test et l'avancement par rapport au plan de test• Les facteurs qui freinent l'avancement• Les tests prévus pour la prochaine période de suivi• La qualité de l'objet de test

Gestion de configuration

- Le but de la gestion de la configuration est d'établir et de maintenir l'intégrité du composant ou du système, du testware et de leurs relations mutuelles tout au long du cycle de vie du projet et du produit. • Au cours de la planification des tests, les procédures de gestion de la configuration et l'infrastructure (outils) devraient être identifiées et implémentées.

Pour supporter correctement les tests, la gestion de la configuration peut impliquer de s'assurer de ce qui suit :

- Tous les éléments de test sont identifiés de façon unique, versionnés, suivis pour les changements et reliés les uns aux autres
- Tous les éléments du testware sont identifiés de manière unique, versionnés, suivis pour les changements, liés les uns aux autres et liés aux versions des éléments de test afin que la traçabilité puisse être maintenue tout au long du processus de test
- Tous les documents et logiciels identifiés sont référencés sans ambiguïté dans la documentation de test

Risques et tests

Définition du risque

- Le risque implique la possibilité d'un événement futur qui a des conséquences négatives.
- Le niveau de risque est déterminé par la **probabilité** de l'événement et son impact (le préjudice).

Risques produit	Risque projet
<p>Le risque produit implique la possibilité qu'un produit d'activités (par exemple, une spécification, un composant, un système ou un test) puisse ne pas satisfaire les besoins de ses utilisateurs et/ou parties prenantes. Voici des exemples de risques produit:</p> <ul style="list-style-type: none"> • Le logiciel peut ne pas remplir les fonctions attendues selon la spécification. • Une architecture système peut ne pas répondre de manière adéquate à certaines exigences non fonctionnelles • Un traitement particulier peut être effectué de manière incorrecte dans certaines circonstances. • Une structure de contrôle de boucle peut être mal codée <p>Les délais de réponse peuvent être insatisfaisants pour un système de traitement des transactions à haut niveau de performance .</p> <ul style="list-style-type: none"> • L'expérience utilisateur (UX)pourrait ne pas satisfaire les attentes pour le produit 	<ul style="list-style-type: none"> • Le risque projet implique des situations qui, si elles se produisent, peuvent avoir un effet négatif sur la capacité d'un projet à atteindre ses objectifs. Voici des exemples de risques projet : • Problèmes liés au projet • Problèmes organisationnels • Questions techniques • Problèmes liés aux fournisseurs

Gestion des défauts

Objectives des rapports de défauts	Les éléments du rapport de défaut
<ul style="list-style-type: none"> • Fournir aux développeurs et aux autres intervenants de l'information sur tout événement indésirable survenu • Fournir aux Test Managers un moyen de suivre la qualité du produit d'activités testé et l'impact sur les tests • Fournir des idées pour le développement et l'amélioration du processus de test 	<ul style="list-style-type: none"> • Un identifiant • Un titre et un bref résumé du défaut signalé • La date du rapport de défaut, l'organisation émettrice et l'auteur • L'identification de l'élément de test (élément de configuration testé) et de l'environnement • La ou les phases du cycle de développement au cours desquelles le défaut a été observé <ul style="list-style-type: none"> • Une description du défaut pour permettre la reproduction et la résolution, y compris des logs, des captures de base de données, des captures d'écran ou des enregistrements • Les résultats attendus et obtenus • La portée ou le degré d'impact (sévérité) • Urgence/priorité de la correction • L'état du rapport de défaut • Les conclusions, recommandations et approbations • L'historique des modifications, notamment la séquence des mesures prises par les membres de l'équipe de projet quant au défaut pour isoler, réparer et confirmer qu'il est corrigé • Les références, y compris le cas de test qui a révélé le problème

- Au cours du processus de gestion des défauts, certains rapports peuvent s'avérer contenir des faux positifs, et non des défaillances réelles dues à des défauts. .
- Par exemple, un test peut échouer lorsqu'une connexion réseau est interrompue ou a un temps de réponse trop long. Ce comportement ne résulte pas d'un défaut dans l'objet de test, mais est une anomalie qui doit être examinée. Les testeurs devraient faire en sorte de réduire au minimum le nombre de faux positifs signalés comme étant des défauts.
- Certains de ces détails peuvent être automatiquement inclus et/ou gérés lors de l'utilisation d'outils de gestion des défauts, par exemple, l'attribution automatique d'un identifiant, l'attribution et la mise à jour de l'état du rapport de défaut pendant le workflow, etc
- Les défauts détectés au cours des tests statiques, en particulier les revues, seront normalement documentés d'une manière différente, par exemple, dans les notes de réunion de revue.