

Agile Test Practice Improvement **AGILE TPI**

Version 2.4 mars 2017



Informations sur les versions

Version	Date d'émission	Remarques	Auteur
0.1	5 septembre 2014	Mise en place et concept initial	Gerrit de Vries
0.2	24 septembre 2014	Ajout des contributions de : - Yvonne van Zaanen (Triodos) - Jos Punter (Sogeti NL) - Bert Linker (Sogeti NL) - Ben Visser (Sogeti NL)	Gerrit de Vries
0.3	25 septembre	Ajout des contributions de : James Redpath (Sogeti USA)	Gerrit de Vries
1.0	Décembre 2014	Version finale	Gerrit de Vries
1.1	Février 2015	Ajustements mineurs	Gerrit de Vries
1.2	Avril 2015	Contributions ajoutées de Arne Bergs (Sogeti DE)	Gerrit de Vries
1.3	Juillet 2015	Synchronisé avec l'outil ATIM	Gerrit de Vries
1.4	Août 2016	L'ATIM est passé en Agile TPI	Gerrit de Vries
2.0	Août 2016	Synchronisé avec l'outil TPI Agile Changements mineurs Contribution d'Eric Belt et de Ben Visser	Gerrit de Vries
2.4	Mars 2017	Quelques points de contrôle adaptés Synchronisé avec l'outil 2.4 Contribution d'Eric Belt et de Ben Visser	Gerrit de Vries
2.4 FR	Janvier 2021	Traduction en Français	Bruno LEPRETRE

Résumé

Agile, Scrum et TPI

Les méthodologies agiles ont mis l'accent sur de nombreuses excellentes pratiques du point de vue des tests, telles que le développement piloté par les tests et les tests exploratoires. Néanmoins, la pratique montre également que l'approche et le processus global de test sont souvent moins que satisfaisants dans les projets agiles.

Comme pour les modèles traditionnels, il est nécessaire d'améliorer les activités ou les processus de test. Cet aspect a été développé dans le livre de TPI NEXT® intitulé Business Driven Test Process Improvement¹, où le chapitre 7.2 explique comment le modèle peut être utilisé dans des situations agiles.

Cependant, nous avons appris que les organisations ont du mal à mettre en œuvre et à utiliser des méthodes agiles. Dans de nombreuses situations, nous constatons qu'il s'agit de méthodes dont l'agilité se limite à leur nom, ou dans lesquelles seuls quelques éléments spécifiques de Scrum sont adoptés comme les stand ups (quotidiens). Nous avons également appris que le testeur professionnel est capable de contribuer aux équipes agiles grâce à ses compétences et surtout à son expérience. Par son travail de testeur, il devient le "centre des opérations", travaillant en étroite collaboration avec les analystes, les développeurs et les utilisateurs. À partir de ce poste, le testeur est également capable de convaincre les autres des avantages des tests, créant ainsi un état d'esprit axé sur la qualité avec les autres membres de l'équipe et les parties prenantes.

Sur la base de notre expérience et de nos activités de test dans des environnements agiles, nous pensons que ce document aidera les organisations, les équipes projet et les individus à appliquer le Manifeste Agile qui stipule

Nous découvrons de meilleures façons de développer des logiciels en le faisant et en aidant les autres à le faire. Grâce à ce travail, nous en sommes venus à valoriser :

Individus et interactions	plus que	processus et outils
Logiciels opérationnels	plus que	documentation exhaustive
Collaboration avec les clients	plus que	négociation de contrats
Répondre au changement	plus que	suivre un plan

Si les objets de droite ont de la valeur, ceux de gauche en ont davantage !

Bien que le terme "Agile" ait de nombreuses variantes, nous utiliserons la terminologie Scrum pour ce document, car Scrum est une méthodologie couramment utilisée. En outre, nous faisons référence à "TMap NEXT® in Scrum"², où les auteurs utilisent la méthodologie Scrum comme point de départ.

Développeur ou testeur

Dans l'idéal, l'équipe Scrum sera composée de développeurs, sans rôle ni expertise spécifiques : *Scrum ne reconnaît aucune sous-équipe dans l'équipe de développement, indépendamment des domaines particuliers qui doivent être abordés comme les tests ou l'analyse business ; il n'y a pas d'exception à cette règle*³.

Dans la réalité, nous constatons cependant que les développeurs ont un rôle spécifique, en particulier le testeur. Avec ce document, nous ne voulons pas remettre en cause le Scrum Guide mais, en nous basant sur la situation pratique, soutenir et améliorer les activités de test au sein des équipes. En outre, nous visons une situation dans laquelle chaque membre de l'équipe est capable d'effectuer les activités de test de la bonne manière ; lorsque nous nous adressons au "testeur", nous impliquons chaque membre de l'équipe qui effectue les activités de test. Le test dans une équipe Scrum est une activité d'équipe : en tant qu'équipe, vous développez des scénarios de test, en tant qu'équipe, vous décidez ce qui sera testé, où et comment (test manuel ou automatisé). Les tâches découlant de cette activité, et indiquées sur le Scrum board, peuvent être choisies par n'importe quel membre de l'équipe.

Comment lire

¹ Voir : Business Driven TPI NEXT® ; Gerrit de Vries et autres - UTN Publishers 2009.

² Voir : TMap NEXT® in Scrum ; Leo van de Aalst et Cecile Davis, Sogeti Nederland B.V. 2012.

³ Voir : The Scrum Guide, juillet 2013

Le chapitre suivant présente une introduction au monde de l'amélioration des pratiques de test et la façon dont nous abordons l'amélioration des tests Agiles. Le chapitre 3 se concentre sur les compétences du testeur (ou de celui qui effectue les activités de test), qu'il s'agisse de compétences logicielles ou techniques.

Au chapitre 4, nous décrivons, sur la base des points de contrôle du modèle TPI NEXT®, les pratiques qui peuvent être appliquées aux activités de test. Cette question est abordée selon trois points de vue opérationnels :

- P : Pratiques de test pour le développeur individuel (Professionnel)
- T : Pratiques pour l'ensemble de l'équipe (Team)
- O : Pratiques au niveau des Organisations, des programmes ou des projets.

Le document contient également une annexe. Il s'agit du texte du livre (Business Driven test Process Improvement), chapitre 4. L'annexe peut être utilisée comme référence pour obtenir plus d'informations sur les secteurs clés ou les points de contrôle qui s'y rapportent.

Outillage

Ce document est accompagné d'un outil AGILE TPI qui peut être utilisé pour évaluer et analyser les pratiques et les organisations. Pour la situation Agile, nous avons adapté l'outil TPI NEXT®, afin qu'il puisse être facilement utilisé pour trouver les goulots d'étranglement dans les tests et obtenir des indications d'amélioration. L'outil (et son manuel d'utilisation) peut être téléchargé sur le site www.tpinext.com. Le manuel de l'utilisateur conseillé en annexe vous aidera à utiliser correctement l'outil.

Table des matières

1	Introduction	1
1.1	Besoin d'amélioration ?	1
1.2	Activités de test distinguées.... mais non séparées	1
1.3	Différents points de vue et niveaux	2
2	Le modèle AGILE TPI	3
2.1	Secteurs clés	3
2.2	Catégories	3
2.3	Points de contrôle	4
2.4	Fixer des priorités	5
2.5	Clusters	6
3	Stratégie d'amélioration	7
3.1	Déterminer où vous êtes	7
3.2	Partir de zéro	7
3.3	La situation hybride	7
3.4	Amélioration progressive avec les clusters	7
3.4.1	Première étape : Groupe A	8
3.4.2	Etape suivante : Groupe B	9
3.4.3	Groupe C	10
3.4.4	Groupe D	11
3.4.5	Groupe E	13
3.4.6	Groupe F	14
3.4.7	Groupe G	15
4	Améliorer les capacités de test	16
4.1	Se concentrer sur les compétences de test	16
4.1.1	Introduction	16
4.1.2	Compétences techniques	16
4.1.3	Compétences relationnelles	16
4.1.4	Le testeur en tant que coach	18
5	Annexe A : Secteurs clés et leur signification	20
6	Annexe B : Guidelines pour l'outil AGILE TPI	23

1 Introduction

1.1 Besoin d'amélioration ?

Les activités de test jouent un rôle important dans chaque programme de développement de logiciel, que ce soit dans les environnements dits "en cascade" ou lorsque des méthodes plus agiles et itératives sont utilisées. Les organisations courent le risque de voir apparaître des défauts et des failles dans leurs programmes logiciels, leurs systèmes ou leurs solutions informatiques complètes de bout en bout. C'est l'une des principales raisons des tests. Un autre facteur important pour les activités de test est de gagner la confiance des utilisateurs et des entreprises dans les produits informatiques. Par conséquent, les "tests" doivent donner un aperçu de la qualité des produits et des solutions.

En outre, de plus en plus d'organisations exigent des tests *efficaces* : les résultats des activités de test sont produits avec un minimum d'efforts, en consacrant le moins de temps, d'argent et d'autres ressources possible pour le niveau de qualité requis. En termes de TMap®, cela signifie : Tests⁴ basés sur le risque et sur l'humain, en utilisant les bons instruments, de la bonne manière et au bon moment.

Depuis 1998, les organisations utilisent le modèle TPI NEXT(R) pour évaluer la maturité de leurs tests et les améliorer si possible. En 2009, nous avons mis en place le modèle TPI NEXT® pour améliorer l'approche d'une manière plus orientée vers le métier et se concentrer sur des environnements comme Agile. Le livre TPI NEXT® contient une section où l'utilisation du modèle dans ces environnements est décrite. Bien que les recommandations de cette section soient toujours valables, nous pensons qu'il y a également place à l'amélioration. Pas le moins du monde parce que certaines organisations ont encore du mal à tirer profit des excellentes activités de test dans un environnement Agile.

1.2 Activités de test distinguées.... mais non séparées

Fournir un article sur l'amélioration des activités de test Agile sonne comme une *contradictio in terminis* ;

- "Scrum ⁵ne reconnaît aucun titre aux membres de l'équipe de développement autres que celui de développeur, quel que soit le travail effectué par la personne ; il n'y a pas d'exception à cette règle ;
- Scrum ne reconnaît aucune sous-équipe dans l'équipe de développement, indépendamment des domaines particuliers qui doivent être traités comme les tests ou l'analyse business ; il n'y a pas d'exception à cette règle ;
- Les membres individuels de l'équipe de développement peuvent avoir des compétences spécialisées et des domaines d'intérêt, mais la responsabilité appartient à l'équipe de développement dans son ensemble".

On pourrait donc dire qu'il n'existe pas de "processus de test", et encore moins de rôle de testeur, d'ingénieur de test ou de test manager. Néanmoins, des *pratiques de test* existent dans le cadre d'Agile, de sorte que les activités de test peuvent être distinguées des autres activités de développement et peuvent donc être évaluées, appréciées et améliorées si nécessaire. En outre, les *compétences spécialisées* mentionnées méritent d'être examinées. Grâce à nos années d'expérience en tant que testeurs, nous pensons que les tâches et les activités du testeur ne sont pas seulement cruciales pour la qualité des produits finaux ; le testeur est également le "centre d'opérations" en ce qui concerne la qualité des exigences, des conceptions, des semi-produits, de l'assistance aux utilisateurs et de l'assistance à l'acceptation. Voir aussi : Le testeur en tant que coach.

Ce poste donne au testeur (ou plutôt : au développeur qui effectue les activités de test) une occasion unique d'encadrer les autres développeurs. Son expérience lui permet d'améliorer non seulement ses propres activités, mais aussi celles de toute l'équipe. C'est pourquoi nous avons distingué les activités de test dans les environnements Agiles ; non pas pour les séparer ou les mettre à part, mais pour les rendre aussi fortes que possible, ainsi que les autres activités de développement.

⁴ Pour en savoir plus sur le test TMap Human Driven, consultez le site www.tmap.net.

⁵ De : *The Scrum Guide*, juillet 2013

1.3 Différents points de vue et niveaux

L'amélioration des activités de test dans les environnements Agiles peut être initiée à différents niveaux et avec différents points de départ. En outre, de nombreuses organisations sont en transition vers Agile. Nous nous concentrerons sur la vision opérationnelle des activités de test :

- Activités de test réalisées par un ou plusieurs membres **individuels** de l'équipe.
- Activités de test réalisées par une ou plusieurs **équipes**.
- Activités de test du point de vue de l'**organisation**.

Dans les paragraphes suivants, nous utiliserons ces distinctions pour appliquer les éléments du modèle TPI NEXT® aux améliorations et aux lignes directrices.

2 Le modèle AGILE TPI

Le modèle original TPI NEXT® a été utilisé comme point de départ pour établir des recommandations visant à améliorer les activités de test Agile. La principale différence avec le modèle original réside dans les niveaux de maturité. Nous avons adopté les vues opérationnelles mentionnées précédemment (individu/équipe/organisation) où un certain niveau de maturité peut être à la fois mesuré et amélioré. En outre, la méthode des clusters est utilisée pour définir l'amélioration par étapes.

2.1 Secteurs clés

Comme dans le modèle original, il y a 16 secteurs clés⁶ pour regrouper les différentes activités et aspects du test. Ces secteurs clés sont applicables aux trois vues opérationnelles.

		Individual	Team	Organization
1	Stakeholder commitment			
2	Degree of involvement			
3	Test strategy			
4	Test organization			
5	Communication			
6	Reporting			
7	Test process management			
8	Estimating and planning			
9	Metrics			
10	Defect management			
11	Testware management			
12	Methodology practice			
13	Tester professionalism			
14	Test case design			
15	Test tools			
16	Test environment			

2.2 Catégories

Dans le modèle, et plus précisément dans la matrice de maturité, les vues et catégories opérationnelles remplacent les niveaux de maturité initiaux. La matrice originale montrait quatre niveaux de maturité : Initial, Contrôlé, Efficient et Optimisé. Pour le modèle Agile TPI, le niveau initial a été supprimé, les trois autres niveaux ont été remplacés (voir l'illustration ci-dessous) par :

- Niveau individuel
- Niveau de l'équipe
- Niveau de l'organisation ou du projet

Dans le modèle Agile TPI, la maturité peut être indiquée et croître par vue opérationnelle (et bien sûr par secteur clé). En fonction des points de contrôle atteints et remplis, les niveaux sont indiqués avec un pourcentage (par exemple : 55 % de score au niveau individuel).

⁶ Voir l'annexe A pour de plus amples descriptions.

		Individual	Team	Organization
1	Stakeholder commitment			
2	Degree of involvement			
3	Test strategy			
4	Test organization			
5	Communication			
6	Reporting			
7	Test process management			
8	Estimating and planning			
9	Metrics			
10	Defect management			
11	Testware management			
12	Methodology practice			
13	Tester professionalism			
14	Test case design			
15	Test tools			
16	Test environment			

2.3 Points de contrôle ⁷

Afin d'évaluer le niveau de maturité d'un certain point de vue et d'indiquer des suggestions d'amélioration, des points de contrôle TPI NEXT adaptés sont utilisés. Dans l'illustration ci-dessous, les points de contrôle sont indiqués par leur secteur clé (première position), la vue opérationnelle (deuxième position) et le numéro de séquence (troisième position). Exemple : 1.P.1 signifie : le premier point de contrôle pour le niveau professionnel appartenant au premier secteur clé (engagement des parties prenantes).

	Individual				Team				Organization					
1 Stakeholder commitment	1		2		3		1		2		1			
2 Degree of involvement	1	2	3	4	1				1		2			
3 Test strategy	1	2	3	4	A		C		C		A		C	
4 Test organization	1	2	3	4	D		F		D		G			
5 Communication	1		1		1		2		1					
6 Reporting	1		2		3		1		2		3		1	
7 Test process management	1	2	3		1		2		3		1		2	
8 Estimating and planning	1		2		1		2		1					
9 Metrics	1				1				1		2			
10 Defect management	1	2	3		1	2	3	4	1		2			
11 Testware management	1		2		1		2		1		2			
12 Methodology practice	1				1				1	2	3	4		
13 Tester professionalism	1	2	3	4	1				1		2			
14 Test case design	1	2	3		1		2		3		1			
15 Test tools	1		2		1				1	2	3			
16 Test environment	1	2	3		1				1	2	3			

⁷ Dans la matrice de maturité des tests de l'outil AGILE TPI, l'identification des points de contrôle est remplacée par la lettre de leur groupe (A, B, C, etc.).

2.4 Fixer des priorités

Les priorités peuvent différer entre tous les points de contrôle ou recommandations ; un point de contrôle pourra être plus important qu'un autre. L'une des raisons de fixer des priorités pour les points de contrôle est l'aspect "business driven"⁸ du modèle : dans la situation Agile, les secteurs clés ont la pondération suivante car ils contribuent plus ou moins à renforcer l'Agile :

1	Stakeholder commitment
2	Degree of involvement
3	Test strategy
4	Test organization
5	Communication
6	Reporting
7	Test process management
8	Estimating and planning
9	Metrics
10	Defect management
11	Testware management
12	Methodology practice
13	Tester professionalism
14	Test case design
15	Test tools
16	Test environment

	High
	Medium
	Low

L'autre raison de fixer des priorités pour les points de contrôle est leur interdépendance : certains points de contrôle doivent être remplis pour permettre un autre point de contrôle. Cette hiérarchisation est formalisée dans les Clusters.

⁸ Voir aussi : *TPI NEXT Business Driven Test Process Improvement, annexe B3.1. Secteurs clés catégorisés.*

2.5 Clusters

Les clusters sont une combinaison de certains secteurs clés et d'un nombre limité de points de contrôle. Partant d'un niveau initial et allant vers un niveau de maturité plus élevé, les étapes d'amélioration sont indiquées par A (la première étape), B (la deuxième étape), etc. Ceci est indiqué dans l'illustration ci-dessous, où le premier groupe (A) des trois niveaux est marqué en vert.

		Individual				Team			Organization		
1	Stakeholder commitment	C	D	D		C	E		C		
2	Degree of involvement	A	A	B	C	A			A	B	
3	Test strategy	A	B	C	D	A	C	C	A	C	
4	Test organization	D	E	F	G	D	F		D	G	
5	Communication	A		C		B	D		C		
6	Reporting	D	F	F		D	E	F	E		
7	Test process management	F	F	G		B	B	E	E	F	
8	Estimating and planning	B		B		A	B		F		
9	Metrics		B			F			G	G	
10	Defect management	C	D	E		A	C	C	D	D	E
11	Testware management	C		C		E	F		D	E	
12	Methodology practice		B			C			A	B	B
13	Tester professionalism	A	B	B	E	A			A	A	C
14	Test case design	A	B	B		A	B	C	C		
15	Test tools	A	A			A			A	C	B
16	Test environment	A	C	C		A			A	A	A

Fixer des priorités

Notez que chaque catégorie (individu, équipe, organisation) a son propre ensemble de groupes et que chaque groupe a un ensemble unique de points de contrôle : Le groupe A au niveau de l'équipe est différent du groupe A au niveau individuel. Les groupes donnent une indication de la priorité : ce qui doit être fait en premier, ce qui vient après. En pratique, les améliorations seront mises en œuvre en combinant des points de contrôle provenant de différents points de vue opérationnels.

3 Stratégie d'amélioration

3.1 Déterminer où vous êtes

Il est toujours conseillé de déterminer la position de départ : comment les tests sont-ils effectués jusqu'à présent, quels sont les goulets d'étranglement connus ou cachés, quelles activités sont assez bien réalisées, quelles sont les possibilités de gains rapides. Cette position de départ peut être assez bien mesurée en utilisant l'outil d'évaluation AGILE TPI, qui illustre la maturité du test à l'aide de tous les points de contrôle. L'utilisation de l'évaluation ne donnera pas seulement un aperçu de la maturité du test. Elle indique également les dépendances avec d'autres capacités et permet d'établir des priorités pour les améliorations à court et à long terme : ce qui doit être fait en premier, ce qui vient ensuite (voir également les Clusters).

3.2 Partir de zéro

Lorsque les activités de test doivent être organisées et mises en place dans des équipes Agiles, où les équipes, les projets et l'organisation sont nouveaux dans le domaine Agile et où l'Agilité est l'approche pour tous les projets, il est conseillé d'utiliser le modèle AGILE TPI avec un large spectre. Cette stratégie combine les points de contrôle et les étapes d'amélioration des trois catégories, ce qui permet d'adopter une approche générique. Commencez par évaluer la situation en utilisant tous les points de contrôle du premier groupe (A)⁹ dans toutes les catégories (P, T et O). Ensuite, déterminez les mesures d'amélioration nécessaires, également sur la base des groupes de toutes les catégories. Élaborez un plan d'amélioration, couvrant les étapes d'amélioration, décliné en activités et actions estimées et planifiées, en tâches et en responsabilités.

3.3 La situation hybride

Dans le cas où les deux méthodes sont utilisées, nous conseillons d'utiliser le modèle TPI NEXT® pour la situation waterfall et le modèle TPI AGILE pour la situation Agile. La comparaison des résultats de l'évaluation des deux modèles permet de prendre des décisions adéquates sur les mesures d'amélioration nécessaires. En outre, il est possible de mettre en œuvre des activités d'amélioration dans les deux environnements, en suivant un plan d'action détaillé basé sur des estimations et une planification réalistes. La combinaison et la comparaison des deux modèles permettent d'apprendre l'un de l'autre, d'utiliser les meilleures pratiques de chacun. Assurez-vous que la direction s'engage à apporter les changements et les améliorations nécessaires en indiquant et en fixant les priorités pour les deux méthodes ou pour l'une d'entre elles.

3.4 Amélioration progressive avec les clusters

L'amélioration des activités de test dans les environnements agiles doit se faire de manière contrôlée et par étapes. C'est là que les Clusters peuvent être utilisés. Partant d'un niveau initial et allant vers un niveau de maturité plus élevé, les étapes d'amélioration sont indiquées par A (la première étape), B (la deuxième étape), etc.

La première étape d'amélioration consiste en l'ensemble des points de contrôle des différents secteurs clés du groupe A. Un plan et une stratégie d'amélioration doivent définir la faisabilité des étapes d'amélioration ; parfois, il est nécessaire de prendre des mesures plus modestes, d'inclure moins de points de contrôle pour améliorer. Les paragraphes suivants montrent l'amélioration par étapes en utilisant les groupes.

3.4.1 Première étape : Groupe A

SECTEUR CLÉ	NR.	CHECKPOINT
Degré d'implication	2.1.1	Les activités de test telles que l'analyse des risques produit, la planification, la préparation, la conception des cas de test sont effectuées avant l'exécution des tests, dans le but de maintenir ces activités hors du chemin critique des sprints.
Degré d'implication	2.1.2	Les testeurs/développeurs participent à la planification du projet et des sprints afin d'aligner les activités de test sur les autres activités.
Degré d'implication	2.2.1	Le développeur/testeur contribue à l'optimisation de la base de test (au-delà de la revue de testabilité), dans laquelle l'objet testé est décrit.
Degré d'implication	2.3.1	Les leçons tirées des activités de test évaluées sont utilisées pour (la mise en place de) futurs projets.
Stratégie de test	3.1.1	Les risques produit sont identifiés et gérés au cours du projet, par exemple avec du risk poker pour les stories.
Stratégie de test	3.2.1	La stratégie est suivie et, le cas échéant, adaptée pendant le projet.
Stratégie de test	3.3.1	Le processus d'analyse des risques produit et stratégie est évalué lors de la rétrospective ou lors du raffinement des user stories.
Communication	5.1.1	Chaque développeur/testeur participe (pro)activement aux daily stand-ups pour fournir et recevoir des informations pertinentes.
Professionalisme du testeur	13.1.1	Les membres de l'équipe de développement ayant un rôle de test ont suivi une formation sur les tests, en particulier sur l'ingénierie des tests et la conception des cas de test.
Professionalisme du testeur	13.2.1	L'expertise requise - métier et/ou technique - pour et au sein de l'équipe est disponible en cas de besoin.
Outils de test	15.1.1	Un outil de test est utilisé pour effectuer automatiquement les tests de régression
Outils de test	15.1.2	Les outils sont disponibles et peuvent être utilisés à tout moment, voire en permanence, en fonction des besoins.
Outils de test	15.2.1	Une formation à la bonne utilisation des outils de test (mis en œuvre) est dispensée et suivie.
Outils de test	15.3.1	Maintenance, base de connaissances et assistance pour les outils de test utilisés, sont disponibles.
Outils de test	16.1.1	Les cas de test (automatisés) peuvent être exécutés sur tous les environnements pertinents.
Outils de test	16.1.2	La configuration requise de l'environnement de test et les données de test sont clairement définies au début de l'itération.

3.4.2 Etape suivante : Groupe B

SECTEUR CLÉ	NR.	CHECKPOINT
Degré d'implication	2.1.3	Les aspects qualité et tests sont bien pris en compte dans l'analyse et l'atténuation des risques globaux du projet et des sprints par l'équipe et le Product Owner.
Degré d'implication	2.3.2	Des facilités de formation et de coaching sont fournies avant le début des activités de sprint.
Stratégie de test	3.1.2	Les activités et la stratégie de test (y compris le re-test et les tests de régression) sont basées sur les risques produits.
Communication	5.2.1	Les informations pertinentes (comme la qualité du produit et les risques) sont définies et communiquées efficacement à l'équipe et aux parties prenantes.
Professionalisme du testeur	13.1.2	Les membres de l'équipe de développement connaissent et appliquent les principes essentiels d'une méthode de test comme TMap ou ISTQB.
Professionalisme du testeur	13.1.3	Les connaissances et les compétences individuelles sont partagées au sein de l'équipe, par exemple en laissant les autres tester.
Professionalisme du testeur	13.3.1	L'organisation fournit des facilités de formation et de certification en matière de méthodes de test, comme TMap NEXT® ou ISTQB®.
Outils de test	15.3.2	L'utilisation des outils est évaluée en fonction des aspects suivants : délais plus courts, coût réduit, meilleure qualité ou meilleure gestion du processus de test.
Outils de test	16.2.1	Des engagements avec les fournisseurs concernant la disponibilité des environnements sont établis et adaptés.
Outils de test	16.3.1	Une conception logique ou fonctionnelle de l'environnement de test est fournie et maintenue par l'organisation. Elle porte sur les applications, les systèmes et leurs connexions, ainsi que sur l'utilisation de bouchons et de pilotes (mock-ups).

3.4.3 Groupe C

SECTEUR CLÉ	NR.	CHECKPOINT
Engagement des parties prenantes	1.1.1	Le Product Owner connaît et est connu de tous les membres de l'équipe.
Engagement des parties prenantes	1.2.1	Le Product Owner est propriétaire de l'analyse des risques produit (entrée pour la stratégie de test).
Degré d'implication	2.1.4	Les aspects qualité et tests sont bien pris en compte dans l'analyse des impacts (risques) des changements (user stories, solutions fonctionnelles ou techniques) lors des sessions de raffinement du backlog produit.
Stratégie de test	3.1.3	La stratégie est convenue avec l'ensemble de l'équipe et le Product Owner.
Stratégie de test	3.2.2	Les activités de tests d'interface et d'intégration sont planifiées et discutées avec les autres équipes.
Stratégie de test	3.2.3	En fonction des risques, plusieurs techniques de test sont utilisées pour élaborer les cas de test.
Stratégie de test	3.3.2	Les incidents qui surviennent après les tests/en production sont analysés pour adapter la stratégie de test.
Communication	5.1.2	Les informations pertinentes provenant de parties prenantes extérieures à l'équipe sont (pro)activement recueillies, en particulier dans une situation de Scrum of Scrums.
Communication	5.3.1	Les meilleures pratiques et les leçons apprises concernant les (moyens de) communication et leur efficacité sont régulièrement évaluées et des améliorations sont mises en œuvre.
Métriques	9.1.1	"Les informations et données suivantes sont recueillies et analysées : - Temps requis pour la conception et l'exécution des tests (par cas de test) - Nombre de cas de test qui peuvent être préparés, conçus et exécutés par jour/sprint/projet - Le temps consacré aux activités de test par rapport au total des activités de l'équipe par jour/sprint/projet."
Métriques	9.2.1	Les métriques qui sont utilisées tout au long des sprints sont définies, discutées et évaluées lors des rétrospectives.
Métriques	9.3.1	Les moyens et les procédures pour collecter, stocker, fournir et gérer les métriques qui soutiennent les équipes Agile sont documentés.
Gestion des anomalies	10.1.1	Le cycle de vie des anomalies est aussi court que possible ; les anomalies sont résolues au cours du sprint où elles ont été injectées ou, mieux encore, au cours du raffinement et de la planification ou même lors de la création des user stories.
Gestion des anomalies	10.2.1	Le traitement des anomalies est confié à un rôle spécifique (pas seulement aux développeurs, mais parfois au Product Owner ou au Scrum Master).
Professionalisme du testeur	13.3.2	L'organisation met à disposition des moyens pour participer à des groupes d'intérêt spéciaux, assister à des séminaires sur les tests ou lire et appliquer de la littérature sur les tests.
Conception des tests	14.1.1	Un choix réfléchi, basé sur les risques produits et la stratégie, est fait pour documenter les cas de test à un niveau logique en utilisant un modèle ou un outil

SECTEUR CLÉ	NR.	CHECKPOINT
Conception des tests	14.2.1	Le niveau de couverture de la base de test est clairement indiqué (par exemple dans le document des exigences).
Outils de test	15.3.3	Une politique d'outils est définie pour toutes les activités agiles
Outils de test	16.1.3	Une liste de contrôle prédéfinie est utilisée pour vérifier la (disponibilité et fonctionnalité) de l'environnement.
Outils de test	16.3.2	Une conception technique de l'environnement de test est disponible et formellement acceptée par le test manager ou le spécialiste de l'environnement de test.

3.4.4 Groupe D

SECTEUR CLÉ	NR.	CHECKPOINT
Engagement des parties prenantes	1.1.2	Les tâches et responsabilités (pro)actives du Product Owner sont bien exécutées.
Engagement des parties prenantes	1.1.3	Toutes les parties prenantes concernées par le projet sont définies et connues de toute l'équipe.
Engagement des parties prenantes	1.3.2	Le budget pour les activités (y compris les tests) est accordé par et négociable avec le Product Owner.
Stratégie de test	3.1.4	Lorsque plus d'un niveau de test est exécuté (test unitaire, test fonctionnel, test d'acceptation), les tests sont différenciés entre les niveaux (types de test, couverture et profondeur de test), en fonction des risques analysés.
Organisation des tests	4.1.1	Les tâches et responsabilités en matière de qualité et de tests sont claires et agréées avec le Scrum master et le Product Owner.
Organisation des tests	4.2.1	Avec plus d'un testeur dans l'équipe et dans le cas de plusieurs équipes : les activités de test sont réconciliées et alignées.
Organisation des tests	4.3.1	Les tâches, le rôle, les responsabilités et les attentes du rôle de testeur sont clairement définis en temps utile avant le début des activités de sprint.
Communication	5.2.2	Différents moyens de communication sont envisagés pour transmettre des messages efficaces à l'intérieur et à l'extérieur de l'équipe.
Reporting	6.1.1	Les informations affichées lors du stand-up contiennent des aspects de délais et/ou de coûts, de résultats et (surtout) de risques produit.
Reporting	6.2.1	Le reporting des rétrospectives contient des tendances et des recommandations concernant les délais et le coût des tests.
Reporting	6.2.2	Le reporting des rétrospectives contient des tendances et des recommandations concernant la qualité et les risques produits.
Gestion des anomalies	10.1.2	Lorsque des anomalies sont transférées à une phase ou à un sprint ultérieur, les données suivantes sont stockées : le cas de test correspondant, la personne ayant signalé le défaut, la date, la gravité, une description détaillée (les actions visant à reproduire l'anomalie, le résultat attendu et le résultat observé) et le statut du défaut.

SECTEUR CLÉ	NR.	CHECKPOINT
Gestion des anomalies	10.3.1	Toutes les équipes et tous les projets utilisent le même système (outil) de gestion des anomalies, ou une connexion transparente entre tous les systèmes est assurée.
Méthodologie	12.1.1	Les activités de test sont effectuées conformément à une méthode décrivant notamment les produits de test à livrer et autres exigences sur la manière de travailler.
Méthodologie	12.2.1	L'utilisation pratique des guidelines ou de la méthode de test est évaluée par/au sein de l'équipe.
Méthodologie	12.3.1	La méthode de test décrit, pour toutes les activités, l'objectif, le rôle responsable, les outils et techniques à utiliser et les préconditions.
Professionalisme du testeur	13.1.4	Les compétences en matière de tests sont régulièrement discutées et évaluées par rapport aux attentes avec l'équipe et la direction.
Conception des tests	14.1.2	"Chaque cas test documenté contient les éléments suivants : - Une (description de la) situation initiale - Le processus de changement = les actions de test à réaliser - Le résultat prévu (!) - Les données de test nécessaires."
Conception des tests	14.2.2	Des check-lists ou d'autres moyens tels que des revues sont utilisés pour les caractéristiques qualité (par exemple la facilité d'utilisation) qui ne peuvent pas être couvertes par des cas de test.
Environnement de test	16.3.3	La propriété des environnements de test et des données de test est attribuée à un département distinct.

3.4.5 Groupe E

SECTEUR CLÉ	NR.	CHECKPOINT
Engagement des parties prenantes	1.2.2	Des informations sur la qualité des activités de test et de l'objet du test sont fournies aux parties prenantes.
Organisation des tests	4.1.2	Les tâches et responsabilités en matière de qualité et de tests sont claires pour l'ensemble de l'équipe.
Reporting	6.3.1	Les informations fournies par les équipes sont utilisées pour améliorer la qualité et les tests ainsi que le cycle de vie du développement logiciel.
Gestion du processus de test	7.1.1	Le backlog et la définition of done comprennent les activités de test à effectuer et la qualité requise.
Gestion du processus de test	7.1.2	Le sprint planning et/ou le raffinement du backlog sont effectués avec le Product Owner.
Gestion du processus de test	7.2.1	La définition of done est discutée avec le Product Owner et les autres parties prenantes.
Gestion du processus de test	7.3.1	La (ré)affectation des ressources de test est déléguée à l'équipe ou aux équipes.
Métriques	9.3.2	Les efforts pour fournir des métriques sont mesurés par rapport à leur bénéfice pour les activités Agiles, par exemple la façon dont les métriques contribuent au besoin d'information des parties prenantes.
Gestion des anomalies	10.1.3	Le développeur/testeur contribue, du point de vue des tests, à l'analyse des risques liés aux anomalies.
Gestion des anomalies	10.2.2	Toute l'équipe a accès à la gestion des anomalies (de préférence un outil).
Gestion des anomalies	10.3.2	Les anomalies sont analysées au niveau du projet/programme et des améliorations sont apportées aux pratiques de test afin de prévenir de futures anomalies.
Méthodologie	12.3.2	Un ensemble complet et détaillé de templates fait partie de la méthode de test.
Conception des tests	14.1.3	Dans le cas de test ou le script de test, une référence est faite à la base de test (comme les cas d'utilisation et les user stories), décrivant un comportement spécifique du système.

3.4.6 Groupe F

SECTEUR CLÉ	NR.	CHECKPOINT
Organisation des tests	4.1.3	Les travaux de qualité et de tests en cours peuvent être traités par n'importe quel membre de l'équipe.
Organisation des tests	4.2.2	L'équipe se sent responsable de la réussite et de l'échec du produit et de la qualité du processus. se sent responsable du succès et de l'échec de la qualité des produits et des processus.
Reporting	6.1.2	Le reporting (fréquence et contenu) destiné au Product Owner correspond à ses exigences de base pour le processus de prise de décision.
Reporting	6.1.3	Le reporting est effectué par écrit lorsque les parties prenantes l'exigent et selon leurs besoins.
Reporting	6.2.3	Les demandes de reporting émanant des parties prenantes sont équilibrées par rapport à l'effort nécessaire pour les fournir.
Gestion du processus de test	7.1.3	Chaque activité de test est visible pour l'équipe, fait l'objet d'un suivi et, le cas échéant, d'ajustements.
Gestion du processus de test	7.2.2	Les déviations anticipées par rapport au backlog sont discutées avec le Product Owner lors des sessions de raffinement du backlog.
Gestion du processus de test	7.2.3	Les ajustements du backlog sont documentés en fonction des exigences des parties prenantes.
Gestion du processus de test	7.3.2	La coordination et la gestion des activités de test sont évaluées régulièrement, en interne par l'équipe et avec les parties prenantes.
Estimation et planification	8.1.2	Pour l'estimation, les activités prises en compte sont : la planification des tests, la définition des cas de test, la préparation des données de test, l'exécution des cas de test, la préservation du testware.
Estimation et planification	8.2.2	Les techniques d'estimation sont évaluées lors de la rétrospective.
Estimation et planification	8.3.1	Des métriques telles que les burndown/burnup charts ou la vélocité sont définies et maintenues pour soutenir les activités d'estimation et de planification des équipes.
Gestion des anomalies	10.2.3	Les anomalies sont analysées lors des rétrospectives pour indiquer la qualité de l'objet de test.
Gestion du testware	11.1.1	La base de test, l'objet de test et tout le testware (scripts, données, plans) sont identifiés par leur nom et leur version.
Gestion du testware	11.2.2	Il existe une procédure et une structure logique pour gérer les données de test, le testware, la base de test et l'objet de test.
Gestion du testware	11.3.1	Des guidelines sont fournis pour la préservation du testware en vue d'une utilisation ultérieure dans les différents projets.
Méthodologie	12.3.3	Les éléments obligatoires, conditionnels et optionnels de la méthode de test sont clairement indiqués.
Conception des tests	14.2.3	L'exactitude et l'efficacité des cas de test sont évaluées et améliorées si nécessaire.
Conception des tests	14.3.1	Techniques de conception des tests et check-lists sont évaluées, ajustées et mises à jour en vue d'une réutilisation ultérieure.

3.4.7 Groupe G

SECTEUR CLÉ	NR.	CHECKPOINT
Organisation des tests	4.1.4	Les rôles assignés, les produits et les services sont clairs pour les parties prenantes en dehors de l'équipe.
Organisation des tests	4.3.2	La qualité et les tests des produits et services sont régulièrement évalués au niveau de l'organisation.
Estimation et planification	8.1.1	Pour l'estimation de l'effort (de test), des techniques simples sont utilisées, comme le planning poker.
Estimation et planification	8.2.1	La besoin d'avoir un planning détaillé, indiquant les dépendances entre les variétés de test et les équipes, est discuté avec les parties prenantes.
Gestion des anomalies	10.2.4	Les tendances des anomalies sont analysées. Pour cela, plus d'informations sont enregistrées sur une anomalie : le sous-système, la priorité, le programme et sa version, la base de test et sa version, la cause racine, l'historique des changements de statut et la solution au problème.
Gestion du testware	11.1.2	Chaque cas de test est facilement traçable par rapport à la base de test, de manière transparente.
Gestion du testware	11.2.1	Les éléments sous gestion du testware sont accessibles pour chaque membre de l'équipe.
Gestion du testware	11.3.2	Une procédure est définie pour le transfert du testware aux projets à venir.
Méthodologie	12.3.4	La méthode de test est évaluée périodiquement et des ajustements sont apportés si nécessaire.

4 Améliorer les capacités de test

4.1 Se concentrer sur les compétences de test

4.1.1 Introduction

Comme le dit le Scrum-guide, les membres de l'équipe peuvent avoir des aptitudes et des compétences spécialisées. Dans ce chapitre, nous nous concentrerons sur ces aptitudes à tester.

4.1.2 Compétences techniques

Le testeur était et reste un spécialiste des tests. Le spécialiste du processus de préparation, de conception, d'exécution et de reporting des résultats des cas de test. À cet égard, il n'y a pas de différence entre l'environnement Agile et les autres environnements. Ainsi, les compétences suivantes s'appliqueront toujours à celui qui est responsable des tests :

- Comprendre la réalisation des tests et l'application des techniques de conception des tests
- Savoir mettre en place des cas et des scripts de test dans un ordre logique.
- Avoir un jugement clair sur les résultats des tests.
- Avoir une compréhension et une expérience des tâches de coordination et de gestion des tests, telles que l'analyse des risques et la gestion des parties prenantes.
- Savoir estimer et planifier les activités de test
- Avoir une vision sur la qualité des produits.
- Avoir la capacité d'expliquer le comment, le quoi et le pourquoi des tests.
- Démontrer une affinité avec l'outillage et l'ingénierie.
- Clarifier les besoins d'améliorations (techniques).

En bref, les compétences techniques requises pour les activités de test dans la situation Agile ne sont pas si différentes des compétences techniques que nous connaissons déjà¹⁰.

4.1.3 Compétences relationnelles

Il ne fait aucun doute que la situation Agile nécessite une communication, à tous les niveaux, entre tous les participants, membres de l'équipe et parties prenantes. La communication exige également une attitude proactive : n'attendez pas que quelqu'un vous demande des informations. Tout ce qui peut vous aider, vous et votre équipe, doit être partagé.

Une communication proactive signifie également avoir un intérêt pour l'information. Vous êtes donc très curieux d'entendre les idées, les opinions et les suggestions : tout ce qui peut vous aider, vous et votre équipe.

En outre, les aspects suivants aideront à devenir le meilleur joueur de l'équipe.

Tests basés sur les risques

Plus que dans la situation en waterfall, un principe directeur du développement Agile est : "le mieux est l'ennemi du bien". Cela implique qu'il existe des risques de ne pas atteindre le périmètre ou les fonctionnalités requises ou souhaitées ou que des anomalies puissent se produire. En tant que testeur, vous devez être prêt à prendre des risques, ou mieux : penser et agir en fonction des risques. Analysez les possibilités de faire des erreurs, en considérant consciemment l'impact possible. Être axé sur le risque n'est pas seulement lié au produit testé ou au projet. Il s'agit de votre attitude dans son ensemble, du fait que vous devez établir des priorités pour les choses que vous faites ou ne faites pas, les personnes avec lesquelles vous travaillez et les décisions que vous prenez.

En pratique :

- Envisagez de sauter certains tests que vous aviez prévu de faire et de délibérer des risques éventuels avec vos coéquipiers. Cela peut accélérer le processus et créer une dette de test, mais cela ne doit jamais mettre en péril la qualité convenue (telle que mentionnée dans la définition of done) ni le principe d'avoir un logiciel qui fonctionne. Insistez toujours sur le fait que la qualité est une approche et une responsabilité de toute l'équipe.

¹⁰ voir également le chapitre 4.13 du livre TPI NEXT, qui porte sur le *professionnalisme du testeur*

- Lors de la séance de planning poker, estimez les "risques" : challengez-vous, vous et votre équipe, de prendre des risques dans les estimations.
- Lorsque vous trouvez des défauts, tenez également compte des risques de non-réparation (possibilité de contournement) ou de l'absence de retest après la réparation.

Prendre ses responsabilités

Alors que dans le waterfall et dans des projets et situations comparables, vous pourriez parfois, en tant que parfait testeur, vous retrancher dans votre rôle d'expert. Dans la situation Agile, vous avez besoin de plus. Faire partie de l'équipe Agile signifie : vous êtes en tant qu'individu responsable des résultats de votre équipe. En même temps, votre équipe est responsable des résultats de ses membres individuels. Cela exige davantage que les compétences plutôt techniques dont nous avons parlé dans le chapitre précédent. Vous devez vous présenter, communiquer clairement, vous affirmer et surtout : être un joueur d'équipe.

En pratique :

- Élargissez votre champ d'action (test) à d'autres disciplines de votre équipe, discutez de vos responsabilités et tâches mutuelles.
- Voyez comment vous pouvez aider vos coéquipiers à obtenir les meilleurs résultats possibles
- Faites preuve d'autocritique et discutez avec vos coéquipiers des possibilités d'amélioration afin de savoir comment ils peuvent vous aider.
- Montrez aux autres, y compris au Scrum master et au Product Owner, comment vous avez effectué les tests, ce que vous avez fait ou n'avez pas fait et pourquoi, par exemple la couverture des cas d'utilisation ou des exigences.

Mettez vos compétences et vos aptitudes au défi

Les résultats de votre équipe seront grandement appréciés si vous réussissez à dépasser les attentes. C'est-à-dire de manière positive. Cela nécessite une attitude enthousiaste : vous voulez apprendre de vos erreurs, vous êtes prêt à prendre des risques, à sortir de votre zone de confort. En même temps, vous avez un esprit ouvert envers vos coéquipiers et les parties prenantes extérieures à l'équipe. Vous partagez vos connaissances avec l'équipe et vous la mettez au défi de partager ses connaissances avec vous. N'ayez pas peur d'échouer ; la meilleure façon d'apprendre est de faire des erreurs et d'en tirer les leçons.

En pratique :

- Utilisez une technique de conception de test que vous n'avez jamais utilisée auparavant
- Initiez votre équipe et réalisez avec elle l'analyse des risques produit

Être organisé

Dans la situation Agile, un plan de test est rarement établi. Les activités et les processus sont à cycle court et contrôlable. Néanmoins, il est sage de planifier, de préparer et de structurer votre travail. Il est utile d'être organisé ; un bureau chaotique n'aide pas vraiment à trouver rapidement les choses dont vous avez besoin. En outre, être responsable de ses propres actions signifie également une bonne gestion de ces activités : structurer, aligner, contrôler, être visible.

En pratique :

- Préparez vos tests le plus tôt possible, par exemple avant le début du codage ou en binôme avec le développeur. Dans la mesure du possible, faites intégrer vos tests dans des tests unitaires. Répétez cela pour chaque user story.
- Préparez vos cas de test de manière à ce qu'ils soient faciles à comprendre pour vos coéquipiers.
- Expliquez à votre équipe comment vous travaillez et demandez-leur leur avis.

Faites confiance et respectez vos coéquipiers

En tant que testeur, vous êtes habitué à être critique, désireux de trouver les erreurs que d'autres personnes ont commises. C'est bien et c'est pourquoi on vous a demandé d'être testeur. Mais ce faisant, vous devez toujours faire preuve de respect envers ceux qui ont fait cette erreur, qui ont causé ce défaut dans l'application. Et faites-leur confiance pour tirer les leçons de ces erreurs et améliorer leurs résultats. De plus, félicitez-les pour leur travail. Ils vous respecteront à leur tour pour l'aide que vous leur apportez afin d'offrir une bonne qualité.

En pratique :

- Faites confiance aux autres coéquipiers pour pouvoir exécuter vos cas tests ou encouragez-les à tester eux-mêmes.
- Si vous trouvez un défaut, signalez-le de manière constructive, par exemple en proposant des solutions ou des contournements possibles.
- Discutez de vos cas de test avec les autres développeurs et demandez-leur leur avis.

4.1.4 Le testeur en tant que coach

Au fil des ans et de l'évolution de leur pratique quotidienne, les testeurs sont devenus le centre des opérations dans le cycle de vie des applications. D'une part, ils sont motivés par la nécessité de faire appel aux autres disciplines : architectes, concepteurs, développeurs, métier (management et utilisateurs), opérations, maintenance et fournisseurs. Exemple : le testeur a besoin d'une référence pour ses cas de test : quel est le résultat attendu ou le résultat de cette fonction spécifique, quelle action est nécessaire pour exécuter cette fonction. D'autre part, les autres disciplines et parties dépendent des résultats des tests et sont désireuses d'en connaître l'issue : (quand) pouvons-nous mettre en production, quelle est la qualité des produits (intermédiaires).

À ce poste et sur la base de ses vastes connaissances et de son expérience, le testeur peut agir en tant que coach pour d'autres disciplines. Ce sont surtout les équipes agiles et les parties prenantes extérieures qui peuvent bénéficier de ces connaissances, provenant des testeurs dans leur rôle d'ingénieurs de test, de test managers, de chefs d'équipe de test, etc. dans les différentes phases du cycle de vie des applications. Les paragraphes suivants présentent des exemples de connaissances et d'expérience que le "testeur" peut apporter aux équipes Agiles.

Compétences pour initier un projet :

- Comment mettre en place et gérer un projet.
- Connaissance de la budgétisation ou des estimations.
- Développer et utiliser des modèles.
- Analyser et atténuer les risques produit :
 - Qui impliquer dans une évaluation des risques.
 - Comment identifier les risques.
 - Comment déterminer la gravité des risques (classification).
 - Activités d'atténuation.
- Analyser et atténuer les risques projet.

Compétences en matière de préparation :

- Connaissance de la fonctionnalité requise à travers la partie essentielle d'un cas test : le résultat souhaité ou le résultat attendu d'une fonction, d'un processus ou d'une activité.
- Rendre la conception (tant fonctionnelle que technique) elle-même plus lisible et plus analysable pour les relecteurs.
- Traduire les souhaits des utilisateurs et éviter les surprises lors des test d'acceptation utilisateur.

Compétences pour soutenir le développement/l'implémentation :

- Gérer les défauts et leurs corrections :
 - Fournir une analyse des causes racines.
 - Solutions possibles et réalisables.
 - Solutions de contournement.
 - Évaluation des risques liés aux défauts.
 - Plan et calendrier : corriger immédiatement ou plus tard.
- Un retest est-il nécessaire oui/non !
- Et aussi :
 - Aptitude à apporter les mauvaises nouvelles.
 - Diplomatie.

Compétences pour soutenir l'Acceptation :

- Les tests ne se limitent pas à la recherche d'erreurs et de défauts. Les tests donnent confiance dans le produit.
- Assistance aux utilisateurs :
 - Partage d'expérience / de résultats.

- Ce qui est bon, ce qui est acceptable.
- Soutenir la conception de cas de test.
- Fournir des informations sur la couverture et l'atténuation des risques (métier).

Compétences pour soutenir les opérations :

- Soutenir les utilisateurs.
- Rédiger le manuel d'utilisation.
- Test utilisé comme une formation/formation utilisée comme un test.
- Expliquer la logique des programmes, des fonctions, des systèmes.
- Test "en production" :
 - Soutenir les corrections de bogues.
 - Des corrections rapides.
 - Re-test intelligent.

5 Annexe A : Secteurs clés et leur signification

Cette annexe décrit les secteurs clés tels qu'ils sont utilisés et expliqués en détail dans le livre "Business Driven Test Process Improvement TPI NEXT®". Notez que dans les situations Agiles, l'essence du secteur clé peut être la même mais la terminologie peut être différente.

#	Secteur clé	Groupe	Importance
K01	Engagement des parties prenantes	SR	Les parties prenantes sont engagées et proactives et créent de bonnes conditions pour une communication et une coopération efficaces : L'engagement des personnes impliquées dans les tests est une condition importante pour le bon déroulement du processus de test. Les parties prenantes peuvent être la direction du projet ou la direction hiérarchique, mais aussi les utilisateurs (finaux), la comptabilité interne et le marketing. La partie prenante principale fournit le délai, le budget et les ressources nécessaires (quantitativement et qualitativement) pour réaliser un bon test.
K02	Degré d'implication	SR	L'implication étroite des tests dans le projet permet d'améliorer la qualité du produit dès le début, et contribue à maintenir les activités de test en dehors du chemin critique du projet : La préparation et la coordination des différents tests peuvent être effectuées et la période pendant laquelle le projet de test se trouve sur le chemin critique du projet peut être maintenue aussi courte que possible. L'implication précoce de l'équipe de test dans le cycle de vie du développement logiciel permet de trouver les défauts le plus rapidement et le plus facilement possible et peut-être même d'éviter des erreurs. À un stade précoce, l'équipe de test soutient l'analyse des risques du projet, la revue des exigences et des conceptions.
K03	Stratégie de test	SR	La stratégie de test guide le processus de test vers une répartition optimale des efforts et des ressources : La stratégie de test définit la répartition de l'effort de test et la couverture sur les parties à tester ou les aspects de l'objet de test. La stratégie de test vise à détecter les défauts les plus importants le plus tôt possible et au moindre coût. La qualification "le plus important" est déterminée par les objectifs du test, tels que les processus métier, les facteurs critiques de succès ou les risques perçus du produit.
K04	Organisation des tests	SR	Une organisation de test répond aux besoins des projets en matière de ressources de test, de produits de test et de services de test : Les tests sont l'œuvre de ressources humaines. Chaque participant aura ses propres aptitudes, tâches, responsabilités, autorité, compétences et intérêts, et cela doit être clair pour toutes les personnes concernées. L'organisation des tests prend en compte tous ces aspects et relations afin d'assurer le bon déroulement de la communication et des processus.
K05	Communication	SR	Une communication claire assure une compréhension commune et un alignement des attentes entre toutes les parties concernées : Dans un processus de test, une communication claire et efficace entre les personnes impliquées est nécessaire pour créer les conditions adéquates, coordonner la stratégie de test, ajuster les activités, négocier les priorités, etc. La communication se fait dans différentes directions, depuis, vers et au sein de l'équipe de test. La communication, bien menée, encourage l'interactivité et le retour d'information.

#	Secteur clé	Groupe	Importance
K06	Reporting	SR	Les rapports fournissent aux parties prenantes des informations pour soutenir le processus de prise de décision et le suivi économique projet : Le reporting, en tant que communication monodirectionnelle, vise à fournir des conseils objectifs et un aperçu de la qualité de l'objet et du processus de test. Le reporting fournit une vision économique du projet de test et soutient le processus de prise de décision. Cela fait du reporting l'un des produits de test les plus importants.
K07	Gestion du processus de test	TM	La gestion du processus de test optimise l'exécution de la mission de test dans les délais et coûts alloués et avec les résultats requis : La gestion du processus de test consiste à garder les ressources, les parties prenantes, l'infrastructure, les techniques et le cycle de planification, préparation, actions et vérifications, sous contrôle et à les relier entre eux. Une bonne gestion des tests garantit un processus de test efficace et efficient.
K08	Estimation et planification	TM	L'utilisation correcte de techniques d'estimation et de planification appropriées rend le processus de test prévisible et fiable : L'estimation et la planification des tests indiquent quelles activités doivent être réalisées, quelles sont les ressources (humaines) nécessaires et quand elles sont nécessaires. L'utilisation correcte des techniques d'estimation et de planification rend le processus de test prévisible et fiable.
K09	Métriques	TM	Les métriques permettent d'assurer l'objectivité en quantifiant les observations : Les métriques sont des observations quantifiées et objectives des caractéristiques d'un produit ou d'un processus. Elles sont utilisées pour contrôler le processus de test, pour justifier les conseils donnés par le test et pour comparer les systèmes et/ou les processus. En particulier pour améliorer le processus de test, les métriques sont importantes pour évaluer les conséquences de certaines actions d'amélioration, en comparant les données avant et après l'exécution de l'action d'amélioration.
K10	Gestion des anomalies	TM	La gestion des anomalies traite des anomalies à la fois au niveau individuel et à un niveau plus large où les causes racines sont analysées et des axes d'amélioration proposés : Bien que l'administration et la gestion des anomalies concernent l'ensemble du projet de développement logiciel et non pas spécifiquement l'équipe de test, les testeurs sont les personnes les plus impliquées. Une bonne administration permet de suivre le cycle de vie d'une anomalie et de fournir diverses informations (statistiques), rapports et conseils sur la qualité.
K11	Gestion du testware	TM	La gestion du testware assure la cohérence entre les artefacts de test, et entre les artefacts de test et les documents de conception qui s'y rapportent : Les produits d'entrée (tels que la conception, l'objet de test) ainsi que les produits de sortie des tests (tels que les cas de test, les scripts, les plans de test, les rapports) doivent être maintenables et réutilisables, et doivent donc être gérés. En outre, la gestion du testware assure la cohérence entre les artefacts de test, d'une part, et entre les artefacts de test et les documents de conception correspondants, d'autre part.

#	Secteur clé	Groupe	Importance
K12	Méthodologie	TP	Une méthode de test décrite oriente et soutient les projets de test : Pour chaque activité ou processus de test, une certaine méthode de travail est utilisée, comprenant des actions, des procédures, des règles, des techniques, etc. Cette méthode de test doit être suffisamment générique pour être applicable dans chaque situation, mais contient suffisamment de détails pour orienter et soutenir la pratique quotidienne des tests.
K13	Professionalisme du testeur	TP	Le professionnalisme des testeurs comprend la bonne combinaison des différentes aptitudes, compétences, disciplines, fonctions et connaissances nécessaires pour effectuer les activités de test aux niveaux attendus : Les tests exigent un large éventail de compétences : connaissance et compréhension du métier et de l'organisation, connaissance de l'objet du test, savoir-faire informatique général et compétences techniques. En outre, les testeurs doivent posséder des compétences spécifiques, allant de l'expertise en matière de méthodes à la capacité de communiquer efficacement avec les parties prenantes (compétence sociale). Le professionnalisme des testeurs fait référence à la bonne combinaison de ces aptitudes et compétences qui sont nécessaires pour réaliser les activités de test aux niveaux attendus.
K14	Conception des tests	TP	La conception des (cas de) tests dirige l'exécution du test vers une recherche d'anomalies conforme à la stratégie de test définie : Les cas de test sont dérivés des informations amont (la base de test) d'une manière standardisée. Cette approche donne une information sur la couverture et de la profondeur des tests et facilite la (re)conception et la (re)exécution des cas de test.
K15	Outils de test	TP	Les outils de test permettent ou accélèrent des activités de test spécifiques : Les outils de test sont des aides automatisées ou non pour le processus de test. Les outils de test excellent dans l'exécution systématique de travaux répétitifs, indépendamment de la complexité et du volume, dans des situations où les activités manuelles prendraient trop de temps, seraient infaisables ou sujettes à erreur. Les outils de test permettent ou accélèrent des activités de test spécifiques, ce qui se traduit par un délai d'exécution plus court, une plus grande profondeur de test, une meilleure motivation des testeurs, etc.
K16	Environnement de test	TP	L'environnement de test est explicitement conçu, mis en œuvre et entretenu en fonction des objectifs du niveau de test : L'environnement de test comprend principalement les éléments suivants : • Environnement client • Réseau • Stockage • Serveur • Middleware • Données de test L'environnement de test doit être composé et mis en place de manière à refléter les objectifs du test. L'environnement de test a une grande influence sur la qualité, le délai et le coût du processus de test. Les aspects importants de l'environnement de test sont les responsabilités, la gestion, le respect des délais et une disponibilité suffisante, la représentativité et la flexibilité.

6 Annexe B : Guidelines pour l'outil AGILE TPI

Quick Start

Pour que toutes les macros puissent fonctionner (comme la mise à jour automatique de la matrice de maturité), veuillez cliquer sur la case {Options} située derrière l'avertissement de sécurité et choisir l'utilisation des macros.

Cliquez sur le premier onglet [Informations] pour entrer des données spécifiques sur l'évaluation, l'analyse ou le projet d'amélioration en question. N'oubliez pas d'enregistrer le fichier sous un nouveau nom.

Selon le point de vue que vous choisissez (individu, équipe ou niveau organisationnel), utilisez l'une des feuilles de travail ou utilisez la feuille de travail [All checkpoints]. Tous les onglets sont automatiquement synchronisés. Les checkpoints peuvent être évalués par oui [y,Y] ou non [n,N]. Il est conseillé de remplir la colonne "Notes" pour les affiner ultérieurement ou lorsque vous souhaitez utiliser l'outil pour les évaluations suivantes.

Une fois que tous les checkpoints ont été évalués, l'outil produit une vue d'ensemble des résultats dans l'onglet [Matrice de maturité du test].

Les checkpoints non remplis peuvent être utilisés comme référence pour des activités d'amélioration. La définition des améliorations appropriées est également prise en charge par les Clusters : commencez par les checkpoints du Cluster A, suivi par ceux du Cluster B, etc.

Les guidelines suivants détaillent les différentes étapes, les activités et les onglets de l'outil.

Référence détaillée

GENERALITES

Lors de l'ouverture du fichier, déverrouillez et activez les macros. Cela garantira que les résultats seront automatiquement traités dans l'onglet de matrice de maturité.

Après avoir activé les macros, enregistrez le fichier sous un nom approprié (voir aussi Quick start).

ONGLET INFORMATIONS

Colonne	Guideline
C	Cliquez sur les cellules dans les lignes sous C pour remplir les informations appropriées. Les informations contenues dans les cellules B4 [Projet] et B8 {nom de l'entreprise} seront ajoutées à la matrice de maturité En cliquant sur les cellules C15 et C16, une case permet de choisir les dates appropriées.

ONGLET « ALL CHECKPOINTS »

Colonne	Guideline
A*	N° de séquence Cette colonne fait référence aux numéros des checkpoints, indiqués par [Seq.Nr.]. Le numéro de séquence combine : Position 1 : Référence au secteur clé Position 2 : Indique le niveau opérationnel : P = Niveau Professionnel ou individuel T = Niveau Team O = Niveau Organisation ou projet Position 3 : le numéro de séquence du checkpoint <i>Exemple : 2.P.1 représente le premier checkpoint du secteur clé Degré d'implication (secteur clé 2) pour le niveau professionnel.</i>
B*	Cluster Les lettres indiquées dans cette colonne font référence à l'un des sept clusters (A, B, C, D, E, F ou G). <i>Exemple : Le E de la ligne 12 signifie que le checkpoint fait partie du cluster E.</i>
C*	Checkpoint Dans cette colonne, les checkpoints sont définis. Notez que dans certains checkpoints, plusieurs contrôles peuvent exister (par exemple, le mot "et" est utilisé). Tous les contrôles

sont applicables pour l'analyse et l'évaluation, sauf lorsque le mot "Exemple" est mentionné.

D

Rempli ?

Il s'agit de la colonne permettant de saisir les résultats d'une ou plusieurs évaluations. En cliquant sur la cellule, vous pouvez choisir les valeurs possibles. La valeur par défaut est "NA", ce qui donne une cellule non colorée dans la [matrice de maturité du test]. Les options sont les suivantes :

- "y" ou "Y" lorsque le checkpoint est vérifié et rempli. Cela donne une cellule de couleur verte dans la [matrice de maturité du test].
- "n" ou "N" lorsque le checkpoint a été vérifié mais n'a pas été rempli. Le résultat est une cellule rouge dans la [Matrice de maturité du test].
- "na" ou "NA" lorsque le checkpoint n'est pas applicable dans le contexte. Le résultat est une cellule gris clair dans la [Matrice de maturité du test]

E

Notes

Cette zone est nécessaire à l'enrichissement des résultats de votre analyse. Vous pouvez y expliquer pourquoi un certain checkpoint est, ou n'est pas, rempli ou non applicable. Les explications possibles peuvent être qu'aucune preuve n'a été trouvée dans les documents et/ou lors des entretiens.

Il est conseillé d'ajouter des informations et des notes à tous les checkpoints, sauf à ceux qui n'ont pas été contrôlés. Cela vous aidera à étayer vos rapports et votre présentation à la direction. Cela permet également de rendre le document transférable à d'autres analystes et/ou pour de futures évaluations.

* Note : Les colonnes A, B et C sont fixes. Elles ne peuvent pas être modifiées.

ONGLET MATRICE DE MATURITÉ

Colonne	Guideline
Général	Cette feuille présente les résultats de l'analyse et les valeurs des checkpoints telles qu'elles ont été saisies dans la colonne D de l'onglet [All checkpoints]. Elle contient la matrice de maturité, les résultats par catégorie et au total (chiffres absolus), et le graphique en toile d'araignée pour les résultats par secteur clé (pourcentages).
A -AK	Lignes 8-23 Ces cellules contiennent tous les checkpoints et leurs valeurs. Les checkpoints sont marqués par une lettre correspondant à leur groupe. Les couleurs ont la signification suivante : gris : le checkpoint n'a pas été analysé et/ou évalué ; en raison de la valeur "NA". rouge : le checkpoint a été analysé <u>mais n'a pas été rempli</u> ; en raison de la valeur "n" ou "N". vert : le checkpoint a été analysé et a été rempli ; en raison de la valeur "y" ou "Y". Cliquez sur l'une des cellules (checkpoints) pour voir leur texte (ligne 4) et/ou leur commentaire (ligne 5) repris des notes de l'onglet [All checkpoints]. Lignes 4 et 5 La ligne 4 montre le texte d'un checkpoint lorsque ce dernier est sélectionné et cliqué dans la matrice de maturité. La ligne 5 montre les notes qui ont été ajoutées à ce checkpoint dans l'onglet [All checkpoints].
AN-AU	Lignes 1-9 Le graphique montre les résultats des valeurs de tous les checkpoints en chiffres absolus. Tous les checkpoints remplis sont représentés par une couleur bleue, les checkpoints non remplis ou non analysés restent gris. Barres de gauche à droite : 1ère barre : score de tous les (109) checkpoints 2ème barre : score de tous les checkpoints de la catégorie Professionnel 3ème barre : score de tous les checkpoints de la catégorie Team 4ème barre : score de tous les checkpoints de la catégorie Organisation Lignes 11-23 Le graphique en toile d'araignée montre les résultats des valeurs des checkpoints dans [All checkpoints] par secteur clé, en pourcentage du total. Les résultats seront présentés sous forme de ligne noire sur la toile.