

TMMi en el contexto Ágil

Versión 1.4

Producido por Fundación TMMi

Editor: Erik van Veenendaal

Aviso de derechos de autor
Distribución ilimitada sujeta a derechos de autor
Derechos de autor © Fundación TMMi, Irlanda.

Este material de la Fundación TMMi se suministra "tal como está".

La Fundación TMMi no garantiza, ni de forma explícita ni implícita, a ningún aspecto incluyendo, pero no limitado a, su validez para el objetivo o comerciabilidad, exclusividad o resultados obtenidos del uso del material.

La Fundación no garantiza de ninguna forma con respecto a la libertad de infracción de la patente, marca registrada o derechos de autor.

El uso de cualquier marca registrada en este documento no pretende de ninguna forma infringir los derechos de autor.

Se garantiza el permiso para reproducir este documento y para realizar trabajo derivado del mismo siempre que sea para uso interno, los derechos de autor y declaraciones de "No garantía" se incluyen en todas las reproducciones y trabajos derivados.

Las solicitudes de permiso para reproducir este documento o preparar trabajo derivado del mismo para uso externo y comercial deben ser realizadas a la Fundación TMMi.

TMMi® es una marca registrada por la Fundación TMMi.

Colaboradores

Asim Ali	(United Arab Emirates)
Katalin Balla	(Hungary)
Clive Bates	(UK)
Jan Jaap Cannegieter	(The Netherlands)
Vahid Garousi	(The Netherlands)
Alon Linetzki	(Israel)
Fran O'Hara	(Ireland)
Jurian van de Laar	(The Netherlands)
Leanne Howard	(Australia)
Poonam Jain	(India)
Tim Moore	(UK)
Alfonsina Morgavi	(Argentina)
Meile Posthuma	(The Netherlands)
Matthias Rasking	(Germany)
Chaobo Shang	(China)
Erik van Veenendaal	(Bonaire – Caribbean Netherlands)
Blaine Webb	(UK)
Karolina Zmitrowicz	(Poland)

Control de Versiones

Esta sección se proporciona a efectos informativos.

Versión	Fecha	Comentario
1.0	30/06/2017	Capítulo de introducción y todas las áreas de proceso del nivel 2
1.1	16/05/2018	Interpretación del contexto ágil para las áreas de proceso del nivel 3
1.2	10/12/2018	Interpretación del contexto ágil para las áreas de proceso del nivel 4
1.3	03/07/2019	Interpretación del contexto ágil para las áreas de proceso del nivel 5
1.4	24/12/2019	Secciones 6.4 y 6.5 añadidas para indicar expresamente que los niveles 4 y 5 aplican en su totalidad.

Contenido

1.1	Objetivos	6
1.2	TMMi y Ágil	6
1.3	Integración del Modelo de Madurez de Pruebas (TMMi)	7
1.4	Ágil	8
1.5	Mejora del proceso de pruebas en un contexto Ágil	9
2.1	Área de Proceso 2.1 Política y Estrategia de pruebas	11
2.2	Área de Proceso 2.2 Planificación de las Pruebas	13
2.3	Área de Proceso 2.3 Monitorización y Control de las Pruebas.....	17
2.4	Área de Proceso 2.4 Diseño y Ejecución de Pruebas	20
2.5	Área de Proceso 2.5 Entorno de Pruebas	24
3.1	Área de Proceso 3.1 Organización de prueba	27
3.2	Área de Proceso 3.2 Programa de Entrenamiento en Pruebas.....	30
3.3	Área de Proceso 3.3 Ciclo de Vida de Prueba e Integración.....	31
3.4	Área de proceso 3.4 Pruebas no funcionales	33
3.5	Área de proceso 3.2 Revisiones por pares	37
4.1	Área de Proceso 4.1 Medición de Pruebas.....	40
4.2	Área de Proceso 4.2 Evaluación de la Calidad de Producto	41
4.3	Área de Proceso 4.3 Revisiones Avanzadas	42
5.1	Área de proceso 5.1 Prevención de defectos	44
5.2	Área de proceso 5.2 Control de calidad	45
5.3	Área de proceso 5.3 Optimización del proceso de pruebas.....	46
6.1	Assessments TMMi	48
6.2	Nivel 2 de TMMi, Gestionado	48
6.3	Nivel 3 de TMMi, Definido	48
6.4	Nivel 4 de TMMi, Medido	49
6.5	Nivel 5 de TMMi, Optimización	49

1 Introducción

1.1 Objetivos

Este documento explica cómo combinar un enfoque Ágil para el desarrollo de software con el modelo de mejora del proceso de pruebas TMMi como una ruta posible para alcanzar sus objetivos de negocio. Explica cómo TMMi se puede utilizar en un entorno Ágil proporcionando alternativas probadas a los métodos de prueba tradicionales y de esta manera implementar prácticas del modelo TMMi que pueden aumentar su agilidad. Este documento describe cómo TMMi puede ayudar a las organizaciones ágiles, independientemente del grado de madurez de la organización en su camino hacia niveles crecientes de agilidad, proporcionando recordatorios de prácticas críticas de pruebas que con frecuencia pierden visibilidad a medida que las organizaciones crecen y las presiones del proyecto aumentan. Se discuten cada área de proceso de TMMi y sus objetivos específicos uno por uno. Se establece cómo estos se relacionan con Ágil y cómo se verán las prácticas. Si no se espera que una práctica se realice en un contexto Ágil, ya que no tiene valor añadido, esto también se indica explícitamente.

Muchas (pequeñas) organizaciones Ágiles tienen éxito y crecen; Pero pueden tener pocos procesos documentados y un programa formal de entrenamiento pobre para las personas involucradas. Para mantener el éxito a medida que la organización crece, requerirá un poco más de disciplina en el proceso. Las organizaciones suelen tener el temor de perder la cultura Ágil que las ha llevado al éxito actual. Este es uno de los desafíos a los que hay que hacer frente al iniciar una iniciativa de mejora del proceso de pruebas TMMi en un entorno Ágil.

Este documento tiene muchos públicos objetivo, siendo los dos grupos principales:

- Organizaciones tradicionales maduras que desean pasar a Ágil manteniendo la madurez de sus procesos.
- Organizaciones ágiles que tienen éxito y están creciendo. Como resultado, necesitan algún tipo de madurez del proceso mientras que al mismo tiempo quieren mantener los beneficios de ser ágiles.

Este documento no está diseñado para ser usado independientemente del modelo TMMi original. Su objetivo es proporcionar orientación a aquellos que realicen la mejora del proceso de pruebas en un entorno Ágil en la interpretación y el significado de los distintos objetivos y prácticas TMMi. Este documento complementa el modelo TMMi original y debe utilizarse como documento complementario.

1.2 TMMi y Ágil

La creencia errónea es que los enfoques TMMi y Ágil están en desacuerdo. Los enfoques ágiles y TMMi no sólo pueden coexistir, sino que cuando se integren con éxito proporcionarán beneficios sustanciales. También hay un reto y es el de mirar las pruebas de forma diferente, estando completamente integradas en el desarrollo Ágil y lo que eso significa en el contexto de un programa de mejora de "pruebas". Tenga en cuenta que la "i" de TMMi se refiere al hecho de que la prueba debería ser una parte integrada del desarrollo de software, y no ser tratado como algo que es totalmente independiente. La literatura y las presentaciones sobre las pruebas en los proyectos Ágiles tienden a centrarse en las pruebas unitarias, la automatización de pruebas y las pruebas exploratorias, ¡pero hay más! El uso del modelo TMMi en un contexto Ágil proporciona recordatorios de prácticas de pruebas críticas que a menudo se "olvidan". Este documento mostrará con ejemplos que los métodos TMMi y Ágil pueden trabajar eficazmente juntos. El desafío es aplicar principios "lean" para potenciar las prácticas Ágiles y facilitar las prácticas TMMi.

Al implementar TMMi se debe tener en cuenta que la intención del modelo TMMi no es "imponer" un conjunto de prácticas en una organización, ni tampoco debe aplicarse como un estándar para el cual se debe "demostrar cumplimiento". Utilizado apropiadamente, el modelo TMMi puede ayudar a localizar las áreas de prueba específicas donde el cambio puede proporcionar valor dado los objetivos del negocio. Esto es cierto independientemente del modelo de ciclo de vida que se está aplicando. Es importante recordar siempre que las prácticas de TMMi son un componente esperado, pero también se puede lograr por lo que se conoce como práctica "alternativa" con respecto a una práctica de TMMi definida. Siempre pensar, ¿cuál es la intención de la práctica, ¿cuál es la razón y cómo agrega valor al negocio? A menudo en una cultura Ágil, la intención ya se logra, pero a través de una práctica alternativa. ¡Normalmente "cualquier" solución es compatible siempre y cuando esté impulsada por las necesidades

del negocio! Cuando utilice TMMi no sea demasiado prescriptivo, esta no era la intención inicial de TMMi. Siempre interprete las metas y prácticas TMMi en el contexto de su situación. En general, al establecer primero las necesidades del proceso dentro de su contexto empresarial específico, se pueden tomar decisiones sobre cómo enfocar e impulsar las prioridades de mejora del proceso.

La mayoría de los conflictos que surgen entre TMMi y Ágil se basan en una visión histórica de TMMi de lo que debe ser una "buena práctica" cuando se implementa, o un malentendido de la lógica de las prácticas ágiles sobre la base de cómo se deben soportar los valores ágiles. Los expertos de TMMi, incluidos los 'Assessors', tendrán que volver a considerar y potencialmente repensar mensajes que podrían ser compartidos inadvertidamente relacionados con lo que debería ser una práctica "conforme a TMMi" cuando se implemente. Cuando los enfoques Ágiles se implementan adecuadamente junto con los procesos TMMi, esto resultará en la implementación efectiva de las prácticas de pruebas, no su eliminación. Tenga en cuenta que además de las prácticas específicas de pruebas dentro del modelo TMMi, también hay prácticas genéricas. El propósito de las prácticas genéricas es apoyar la institucionalización de un área de proceso, lo cual significa asegurar efectivamente que la organización tiene una infraestructura para apoyar el área de proceso cuando llegan nuevas personas o cuando ocurren otros cambios dentro de la organización.

Pasar del desarrollo de software tradicional al Ágil también puede llevar a la iniciativa de recortar los procesos tal como se definen hoy en día. De esta manera, las organizaciones basadas en TMMi se beneficiarán de la forma ágil de pensar. Ha habido una tendencia de las personas a interpretar cosas en el modelo TMMi que no están y, por tanto, crear procesos y productos de trabajo innecesarios sin valor añadido. Volviendo a las raíces y los objetivos de mejora y utilizando el modelo TMMi tal como está previsto, apoyará la alineación de los procesos de valor añadido real con las necesidades y objetivos reales del proceso. Recortar los procesos con una mentalidad ágil dará lugar a procesos que reflejan lo que la gente realmente hace y se asegurará de que sólo se recopilen los datos que se utilizan. La mentalidad Ágil también pondrá el foco de atención en mantener las cosas lo más simple posible, lo que normalmente no es fácil, pero traerá un beneficio para aquellos que practican una implementación TMMi. Las mejoras dentro de Ágil normalmente se llevarán a cabo a través de pequeños equipos capacitados que pueden tomar acción rápida, que es otra forma en la que TMMi puede beneficiarse de ser ágil. Recuerde también que hay una ruptura natural entre el nivel 3 de TMMi y los niveles 4 y 5 de TMMi. Especialmente se recomienda a las organizaciones ágiles que elijan y practiquen de forma crítica las prácticas en los niveles TMMi 4 y 5 que importan y tienen valor añadido. Mientras que TMMi es integral, para ser exitoso las organizaciones deben identificar las prácticas clave de las pruebas y las mejoras que requieren atención.

1.3 Integración del Modelo de Madurez de Pruebas (TMMi)

El marco de TMMi ha sido desarrollado por la Fundación TMMi como una guía y marco de referencia para la mejora del proceso de pruebas, abordando aquellas cuestiones importantes para los gerentes de pruebas, ingenieros de pruebas, desarrolladores y profesionales de la calidad del software. Las pruebas dentro de TMMi se definen en su sentido más amplio para abarcar todas las actividades relacionadas con la calidad de los productos de software.

TMMi utiliza el concepto de niveles de madurez para la evaluación y mejora del proceso. Además, se identifican áreas, metas y prácticas del proceso. La aplicación de los criterios de madurez de TMMi mejorará el proceso de pruebas y ha demostrado tener un impacto positivo en la calidad del producto, la productividad de ingeniería de prueba y el esfuerzo de tiempo de ciclo. TMMi ha sido desarrollado para apoyar a las organizaciones con la evaluación y mejora de sus procesos de prueba.

TMMi tiene una arquitectura escalonada para la mejora de procesos. Contiene etapas o niveles a través de los cuales pasa una organización a medida que su proceso de pruebas evoluciona de uno que es "ad hoc" y no gestionado a uno que es gestionado, definido, medido y optimizado. El logro de cada etapa garantiza que todos los objetivos de esa etapa se hayan logrado y que las mejoras constituyan la base para la siguiente etapa.

La estructura interna de TMMi es rica en prácticas de pruebas que se pueden aprender y aplicar de una manera sistemática para apoyar un proceso de pruebas de calidad que mejora en pasos incrementales. Hay cinco niveles en TMMi que prescriben la jerarquía de madurez y el camino evolutivo para la mejora del proceso. Cada nivel tiene

un conjunto de áreas de proceso que una organización debe implementar para alcanzar la madurez en ese nivel. Las áreas de proceso para cada nivel de madurez de TMMi se muestran en la Figura 1.

Un principio subyacente principal del TMMi es que es un modelo genérico aplicable a varios modelos de ciclo de vida y entornos. La mayoría de los objetivos y prácticas definidas por TMMi han demostrado ser aplicables con modelos secuenciales e iterativos de ciclo de vida, incluyendo Ágil. Sin embargo, en el nivel más bajo del modelo, muchas de las subprácticas y ejemplos proporcionados son (muy) diferentes dependiendo del modelo del ciclo de vida que se está aplicando. Tenga en cuenta que dentro de TMMi, sólo los objetivos son obligatorios, las prácticas no lo son.

TMMi está disponible gratuitamente en el sitio web de la Fundación TMMi. El modelo ha sido traducido en español, francés y chino. TMMi también está disponible en formato de libro publicado.

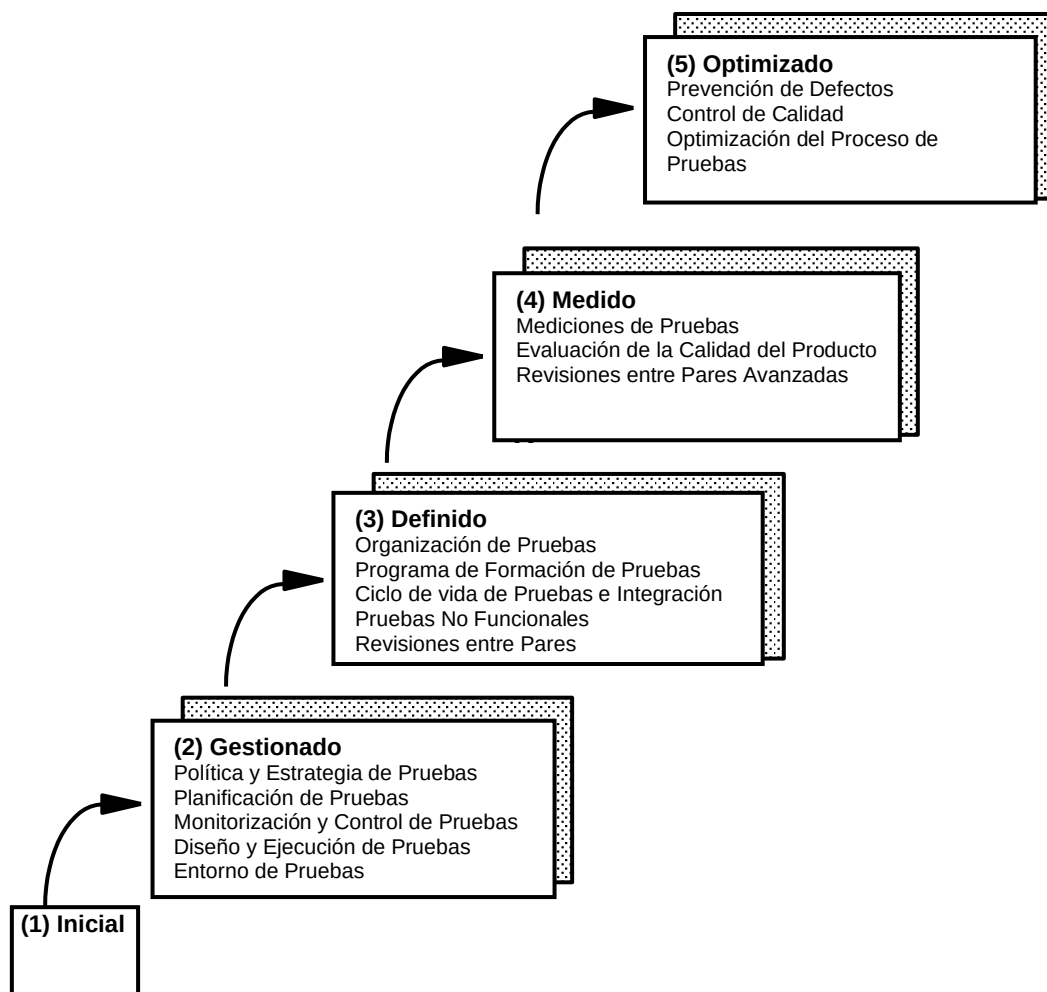


Figura 1: Niveles de madurez y áreas de proceso del modelo TMMi

1.4 Ágil

En el año 2001, un grupo de individuos, que representan las metodologías de desarrollo de software ágil más ampliamente utilizadas, acordaron un conjunto común de valores y principios que se conoció como el Manifiesto para el Desarrollo Ágil de Software o el Manifiesto Ágil. El Manifiesto Ágil contiene cuatro enunciados de valores:

- Individuos e interacciones *sobre* procesos y herramientas
- Desarrollo de Software *sobre* documentación completa
- Colaboración del cliente *sobre* negociación del contrato
- Respondiendo a los cambios *antes* que seguir un plan

El Manifiesto Ágil argumenta que, aunque los conceptos de la derecha tienen valor, los de la izquierda tienen mayor valor. Ágil en sí no es una metodología, pero se han desarrollado diferentes metodologías que proporcionan prácticas para poner estos valores en acción. Tres importantes representantes de metodologías que soportan el concepto Ágil son: “Extreme Programming (XP), Scrum y Kanban”.

1.5 Mejora del proceso de pruebas en un contexto Ágil

Ágil pone mucho énfasis en potenciar los equipos que realizan su propia mejora continua localmente a través de frecuentes retrospectivas. Algunas de estas mejoras pueden estar enfocadas en las pruebas. Estas pueden ser abordadas por un proyecto de mejora basado en TMMi a un nivel superior, pero también pueden ser abordadas por un equipo local que procese los problemas relacionados con las pruebas. Es muy importante que la acción en el contexto de la mejora del proceso de prueba no se interprete y/o se utilice como desposeer de esta propiedad local a los equipos Ágiles.

Algunos de los aspectos principales a tener en cuenta al considerar la influencia de Ágil en el contexto de mejora son:

- Frecuencia del ciclo de mejora.
- Aspectos organizativos.
- Alcance de las mejoras.
- Fuente de mejoras.
- Nivel de documentación (de pruebas).
- Métodos de mejora.

Dentro de los proyectos que utilizan Ágil, las mejoras generalmente tienen lugar en bucles de retroalimentación frecuentes que permiten considerar las mejoras del proceso de prueba con frecuencia (por ejemplo, al final de un sprint cuando se utiliza Scrum). Debido a que el alcance se limita a menudo al ciclo o Sprint anterior, se realizan pequeñas pero frecuentes mejoras que se centran principalmente en la resolución de problemas específicos del proyecto. El enfoque de estas mejoras no suele ser el aprendizaje cruzado de los proyectos y la institucionalización de mejoras.

Si observamos cómo se organiza y se gestiona la mejora de los procesos de prueba, es probable que se concentre menos en un grupo del proceso de prueba a nivel organizativo y se haga más hincapié en la autogestión de los equipos dentro del proyecto. En general, estos equipos tienen el mandato de cambiar el proceso de pruebas dentro del proyecto para satisfacer sus necesidades, dando lugar a procesos altamente adaptados. Sin embargo, algunas organizaciones también realizan reuniones de pruebas semanales para llevar las cosas a un nivel más alto y entre proyectos.

Dado que hay un enfoque más específico de proyecto en la mejora del proceso (de pruebas), es probable que se haga menos hincapié en cuestiones más amplias que afectan a las pruebas en toda la organización. Esto podría significar, por ejemplo, que problemas fundamentales en las pruebas pueden no ser completamente abordados ya que están más allá de este contexto centrado en el proyecto. Un ejemplo típico aquí es el enfoque adoptado para probar ciertos atributos de calidad, como el rendimiento y la fiabilidad. Estos problemas pueden llegar a diferirse de iteración a iteración, ya que a menudo requieren más habilidades y recursos de los que el equipo de proyecto tiene disponibles. Es difícil en estas áreas dar un paso sustancial sin grandes inversiones. Resolver los problemas sólo a nivel de proyecto también podría conducir fácilmente a la suboptimización y a la pérdida de la visión general.

En el contexto Ágil, el rango y el número de ideas alternativas de mejora a considerar puede ser significativamente mayor comparado con los modelos de ciclo de vida no Ágil. Dado que todos los miembros realizan algunas pruebas dentro del proyecto, estas ideas pueden provenir de cualquier miembro del equipo. Esto pone un mayor énfasis en la evaluación y la priorización de las sugerencias de mejora, que puede ser más un esfuerzo de equipo que una tarea asignada a alguien para que mejore el proceso de pruebas. Dado que esto puede requerir el conocimiento específico de pruebas de la persona que mejore el proceso, también puede actuar como un consultor para el equipo a petición.

En los proyectos que utilizan una metodología Ágil, no espere encontrar el nivel de documentación de pruebas que esperaría de los proyectos que usan un ciclo de vida secuencial. Puede haber un solo "documento de prueba" combinado que cubra los elementos esenciales de una política de pruebas, una estrategia de pruebas e incluso un plan de pruebas a alto nivel. Las personas encargadas de mejorar el proceso de pruebas deben evitar hacer sugerencias de "mejora" que requieran una documentación de pruebas más rigurosa y completa. Uno de los principales principios Ágil es, por supuesto, que la documentación sólo se crea cuando existe una necesidad clara e inequívoca de la misma. No sólo la documentación de pruebas será menos detallada, lo mismo ocurre con la documentación del proceso. A menudo, una estrategia denominada "formalizar la informalidad" se aplica con éxito en entornos Ágiles. Si algo está funcionando bien, no hay necesidad de cambiar esto por el bien de TMMi.

Sin embargo, si se documenta, se puede enseñar y compartir con otros, lo que es a menudo beneficioso. Lo que se entiende aquí con la estrategia de "formalizar la informalidad" es que si hay un proceso que funcione, pero es informal de cierta manera, se puede enseñar y documentar tal como se está realizando. Ser ligero con un proceso significa que no todos los posibles escenarios de uso se abordan en su descripción. Por lo tanto, estos procesos deben ser apoyados por mentores y asistencia en el puesto de trabajo, especialmente durante el período de despliegue inicial. Como resultado, para llevar la madurez del proceso de pruebas a una organización Ágil mientras se mantiene la cultura Ágil, también se necesita más - no menos - formación. Del mismo modo durante una evaluación el enfoque para recopilar evidencia se desplazará hacia hacer más entrevistas en lugar de estudiar artefactos.

Los métodos utilizados para proponer mejoras al proceso de pruebas cuando se utilizan metodologías Ágiles tenderán a centrarse en los métodos analíticos para evaluar las causas raíz de los problemas, tales como los diagramas de causa-efecto. Estos son métodos particularmente útiles para la mentalidad de resolución de problemas que es importante al final de una iteración.

2 Nivel 2 de TMMi - Gestionado

2.1 Área de Proceso 2.1 Política y Estrategia de pruebas

El objetivo del área de proceso de Política y Estrategia de Pruebas es desarrollar y establecer una política de pruebas y una estrategia de pruebas para toda la organización o programa en el cual los niveles de pruebas se definan sin ambigüedades. Para medir el rendimiento de las pruebas, se introducen indicadores de desempeño de las mismas.

2.1.1 SG1 Establecer una Política de Pruebas

Cuando una organización desea mejorar su proceso de pruebas, primero debe definir claramente una política de pruebas. La política de pruebas define los objetivos generales de las pruebas de la organización, metas y puntos de vista estratégicos respecto de las mismas y de los profesionales de pruebas. Es importante que la política de pruebas esté alineada con la política general (de calidad) de la organización. Las mejoras en las pruebas deben basarse en objetivos de negocio claros, que a su vez deberían documentarse en la política (de mejora) de pruebas. Una política de pruebas es necesaria para lograr una visión común de las pruebas y sus objetivos entre todas las partes interesadas dentro de una organización. Esta visión común se requiere para alinear las actividades (del proceso de mejora) de pruebas a través de toda la organización.

Hay que tener en cuenta que el objetivo de las pruebas nunca debe ser un objetivo en si mismo sino que su objetivo es asegurar la calidad del producto

Lo anterior también es cierto en una organización que practica el desarrollo de software Ágil. De hecho, dentro de muchas organizaciones hay mucha discusión sobre el rol cambiante de las pruebas, la independencia de las pruebas, la automatización de las pruebas y los probadores profesionales en el desarrollo de software Ágil. Estos temas y otros son típicamente temas que deben ser abordados en una discusión con la gerencia y otras partes interesadas y documentados en una política de pruebas. Cualquier organización, incluyendo aquellas que practican el desarrollo Ágil, que quiere iniciar un proyecto de mejora de las pruebas, necesita identificar y definir los impulsores de negocio y las necesidades de tal iniciativa. ¿Por qué iniciar un proyecto de mejora? Al dedicar tiempo a captar las verdaderas necesidades del negocio, se puede proporcionar un contexto para decidir dónde enfocar las prioridades de mejora (de las pruebas), por ejemplo, sobre qué área del proceso. Hay que tener en cuenta que una política de pruebas suele ser un documento pequeño, de una página, una página web o un gráfico de pared a nivel organizativo y no un documento a nivel de proyecto

El objetivo específico de TMMi de establecer una política de pruebas, incluyendo sus prácticas específicas, es plenamente aplicable a las organizaciones que aplican el desarrollo de software Ágil. Por supuesto, los elementos de una política de pruebas también se pueden incorporar en una política de desarrollo. No existe un requisito específico de TMMi para que sea un documento separado. La política de desarrollo en una organización que aplica Ágil en el desarrollo de software podría, por ejemplo, identificar Scrum como su marco de gestión, XP como el método principal que se utiliza y la adhesión a los valores Ágiles como principio principal. Otro principio importante que podría mencionarse es que todos en el equipo son responsables de todo; también la calidad del producto es responsabilidad del equipo.

Sin embargo, hay una consideración importante para Ágil en relación con la política de pruebas y especialmente con los objetivos (de mejora) de las pruebas. Aunque puede haber objetivos generales relacionados con la mejora del proceso de pruebas en la organización, esto debe ser equilibrado con los proyectos individuales y los equipos Ágiles, que son responsables de mejorar su propio proceso. El desafío Ágil de mejora del proceso es guiar y enmarcar la mejora en un nivel de organización, sin reducir el sentido individual de propiedad de un equipo Ágil de su propio proceso.

2.1.2 SG2 Establecer una Estrategia de Pruebas

La estrategia de pruebas sigue la política de pruebas y sirve como punto de partida para las actividades de pruebas dentro de los proyectos. Una estrategia de pruebas suele definirse para toda la organización o programa. Una

estrategia de prueba típica se basa en una evaluación del riesgo de producto de alto nivel e incluirá una descripción de los tipos de pruebas, cuadrantes de pruebas y niveles de pruebas que se van a realizar, por ejemplo: pruebas unitarias, de aceptación, regresión y de rendimiento. No basta con indicar, por ejemplo, que dentro de una iteración se realizarán las pruebas unitarias y de aceptación. Necesitamos definir qué se entiende por pruebas unitarias y de aceptación; cómo se llevan a cabo normalmente y cuáles son sus principales objetivos. La experiencia demuestra que cuando se define y se sigue una estrategia de prueba, es probable que se produzcan menos superposiciones entre las diversas actividades de pruebas, lo que conduce a un proceso de pruebas más eficiente. Además, puesto que los objetivos de las pruebas y el enfoque de los distintos tipos y niveles de pruebas están alineados, es probable que queden menos agujeros, lo que conduce a un proceso de pruebas más efectivo y, por lo tanto, a un mayor nivel de calidad del producto. Para establecer esta alineación es muy recomendable cubrir tanto las pruebas Ágiles y como las no Ágiles que se realicen en el mismo proyecto bajo una estrategia de pruebas paraguas. Sin embargo, si por un lado hay proyectos Ágiles y por otros proyectos no Ágiles, normalmente hay dos estrategias de pruebas.

Una estrategia de pruebas es un documento vital dentro de un entorno Ágil. Define a alto nivel las pruebas a realizar en los equipos Ágiles (equipos de iteración); qué tipos de pruebas, cuadrantes de pruebas y niveles de pruebas se ejecutan y su enfoque a alto nivel. El documento describe cómo se organizan las pruebas, por ejemplo, qué pruebas se realizan dentro de los equipos Ágiles y qué pruebas se realizan fuera. Definirá la relación entre los equipos Ágiles y los niveles de pruebas que se realizan fuera de su ámbito, por ejemplo, pruebas de integración de hardware/software, pruebas de integración de sistemas o pruebas beta. Una estrategia de pruebas asegura que todos los involucrados en las pruebas comprendan el panorama general de las pruebas.

El documento de estrategia de pruebas liviano es a menudo una buena solución para la organización Ágil y una salida de los planes de pruebas detallados (del proyecto). Durante la planificación de la liberación se discute y se confirma la estrategia de pruebas disponible en toda la organización o programa, o se crea una estrategia de pruebas derivada específicamente para el proyecto. La estrategia de pruebas confirmada o creada proporciona un marco de pruebas que abarca todas las iteraciones.

El objetivo específico de TMMi de "Establecer una Estrategia de Pruebas", incluyendo sus prácticas específicas, es totalmente aplicable a las organizaciones que aplican el desarrollo de software Ágil. Hay que tener en cuenta que las actividades de "distribución" activas, por ejemplo, para el documento de estrategia de pruebas, pueden ser menos relevantes en ágil. Aquellos que son partes interesadas ya deberían haber participado en discusiones anteriores como parte del enfoque de todo el equipo. Sin embargo, una estrategia de pruebas no es a nivel de equipo sino a un nivel superior. Todo el equipo funciona a nivel de grupo, por lo que un documento de organización o de programa aún necesita distribución a los equipos e incluso a otros grupos de interés, por ejemplo, aquellos que realizan actividades de pruebas fuera del equipo si existe un modelo híbrido.

2.1.3 SG3 Establecer Indicadores de Rendimiento de las Pruebas

Los objetivos del negocio para la mejora de las pruebas, tal como se definen en la política de pruebas, deben traducirse en un conjunto de indicadores clave de rendimiento de las pruebas. La política de pruebas y los indicadores de desempeño que lo acompañan, proporcionan una dirección clara y un medio para comunicar los niveles esperados y alcanzados del rendimiento de las pruebas. Los indicadores de desempeño deben indicar el valor de las pruebas y la mejora del proceso de pruebas para las partes interesadas. Dado que las inversiones en la mejora del proceso necesitan apoyo de gestión a largo plazo, es crucial para medir cuantitativamente los beneficios de un programa de mejora mantenerlos motivados. El fin de este objetivo específico de TMMi es definir un número limitado (por ejemplo, 2 o 3) indicadores de rendimiento de las pruebas. No se trata de configurar e implementar un programa de medición completo, sino más bien de definir un conjunto básico de indicadores que indiquen cómo el valor de las pruebas está cambiando con el tiempo y dentro de diferentes entornos de entrega.

Dentro de Ágil, el enfoque será más de equipo y pensamiento sistémico. Esto puede dar lugar a una ampliación correspondiente de los indicadores para el equipo y el sistema en general, en lugar de limitarse únicamente a los aspectos específicos de las propias pruebas. También los indicadores en TMMi del nivel 2 están relacionados principalmente con los resultados finales de las iteraciones. Algunos ejemplos incluirían defectos escapados, velocidad, grados de satisfacción del cliente, esfuerzo/gasto, porcentaje de automatización de las pruebas, etc. El desafío sería definir la combinación apropiada de indicadores relacionados con el enfoque basado en el equipo y el

pensamiento sistémico, al mismo tiempo que proporciona una buena indicación de los logros del rendimiento de un programa de mejora de pruebas basado en TMMi.

La meta específica de TMMi Establecer Indicadores de Rendimiento de las Pruebas, incluyendo sus prácticas específicas, es totalmente aplicable, aunque los indicadores de desempeño seleccionados y aplicados pueden tener un alcance mayor y ser más amplios que estar relacionados sólo con las pruebas. Por supuesto, este último hará que el análisis y la interpretación de los indicadores de desempeño sean más desafiantes. De hecho, los indicadores de desempeño que se usan no se pueden denominar indicadores de rendimiento de las pruebas, sino más bien un indicador de desempeño del equipo o un indicador de desempeño del sistema. Esto también está bien en el contexto de TMMi, siempre y cuando tenga elementos relacionados con las pruebas y se esté utilizando para evaluar el progreso que se está haciendo en la mejora de las pruebas.

2.2 Área de Proceso 2.2 Planificación de las Pruebas

El propósito de la Planificación de las Pruebas es definir un enfoque de pruebas basado en los riesgos identificados y la estrategia de pruebas definida, y establecer y mantener planes bien fundados para realizar y gestionar las actividades de pruebas.

Hay que tener en cuenta que la clave para planificar exitosamente las pruebas está en el pensamiento inicial ("la actividad"), no en la definición del plan de pruebas asociado ("el documento").

Para los ciclos de vida Ágiles, normalmente se producen dos tipos de planificación, la planificación de la liberación y la planificación de la iteración. El área de proceso de planificación de pruebas en TMMi nivel 2 se centra en las actividades relacionadas con las pruebas en la planificación de la liberación y la iteración. La planificación de la liberación prevé la liberación de un producto al inicio de un proyecto. La planificación de la liberación requiere una backlog definido y puede implicar refinar historias de usuarios más grandes en una colección de historias más pequeñas. La planificación de la liberación proporciona la base para un enfoque de prueba y un plan de prueba que abarca todas las iteraciones. Los planes de lanzamiento son de alto nivel. Una vez finalizada la planificación de la liberación, se inicia la planificación de iteración para la primera iteración. La planificación de las iteraciones se enfoca hacia el final de una sola iteración y se refiere a la acumulación de iteraciones.

2.2.1 SG 1 Realizar la Evaluación de Riesgos

Las pruebas exhaustivas son imposibles, y las selecciones siempre deben hacerse y las prioridades siempre deben establecerse. Por lo tanto, este objetivo de TMMi también es aplicable a los proyectos Ágiles. Para los proyectos Ágiles, se realizará una evaluación del riesgo del producto de alto nivel basada en un documento de visión del producto o un conjunto de historias de usuarios de alto nivel en la planificación de la entrega. Para cada iteración se realizará una sesión de riesgo de producto más detallada basada en las historias de usuarios u otros requisitos para esa iteración como parte de la sesión de planificación de la iteración. El proceso de evaluación del riesgo del producto en un proyecto Ágil tendrá un formato mucho más liviano que los aplicados en proyectos tradicionales siguiendo un modelo secuencial de ciclo de vida. Un ejemplo de una técnica ligera de riesgo de producto que se utiliza es el póker de riesgo [Van Veenendaal].

En la planificación de la liberación, los representantes del negocio que conocen las características de la versión proporcionan una visión general de alto nivel de la funcionalidad que se desarrollará, y todo el equipo, incluido el (los) probador(es), ayudará en la identificación y evaluación del riesgo.

Durante la planificación de la iteración, el equipo Ágil identifica y analiza los riesgos del producto basándose en las historias de usuarios que se implementarán en la próxima iteración. Preferentemente, todos los miembros del equipo Ágil y posiblemente otros interesados, participan en la sesión de análisis de riesgo del producto. El resultado es una lista priorizada de elementos de riesgo del producto que identifican las áreas críticas para las pruebas. Esto a su vez ayudará a determinar la cantidad apropiada de esfuerzo de pruebas a asignar para cubrir cada riesgo con suficientes pruebas y secuenciar estas pruebas de una manera que optimice la eficacia y la eficiencia del trabajo de pruebas a realizar. Las tareas estimadas en el tablero de tareas pueden ser priorizadas en parte según el nivel de riesgo de producto asociado a ellas. Las tareas asociadas con riesgos más altos deben comenzar antes e involucrar

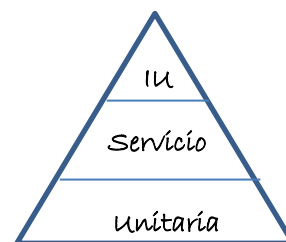
mayor esfuerzo de pruebas. Las tareas asociadas con riesgos más bajos deberían comenzar más tarde e implicar menor esfuerzo de pruebas.

2.2.2 SG2 Establecer el Alcance de las Pruebas

Se define un enfoque de pruebas para mitigar los riesgos del producto identificados y priorizados. Para una iteración específica, los elementos y las características a probar se identifican durante la planificación de la iteración. Esta actividad también se basa en el resultado de la sesión de riesgos de producto. La lista priorizada de elementos a probar se relaciona típicamente con las historias de usuario a probar en esta iteración. Las características normalmente se relacionan, entre otras cosas, con las diversas características de calidad del software a probar. El riesgo de nuevos productos puede aparecer durante la iteración y requieren pruebas adicionales. Cuestiones como los riesgos de nuevos productos que requieren pruebas adicionales normalmente se discuten en las reuniones diarias.

El enfoque de pruebas que se define a nivel de iteración para mitigar los riesgos puede cubrir, por ejemplo, la revisión adicional de las historias de usuarios y los criterios de aceptación, el esfuerzo de pruebas proporcional al nivel de riesgo, la selección de técnica(s) de pruebas apropiadas basadas en el nivel y el tipo de riesgo. El enfoque de la prueba a nivel de entrega será a un nivel mucho más alto y se basará en una estrategia de pruebas definida a nivel de programa o de organización. A menudo un enfoque de pruebas se lleva a cabo o se muestra en el equipo/proyecto wiki.

Un riesgo importante que siempre es evidente en el desarrollo iterativo es el riesgo de la regresión. El enfoque de las pruebas debe definir cómo se maneja el riesgo de la regresión. Normalmente esto se hará creando un conjunto de pruebas de regresión específico, que es preferiblemente automatizado. En este contexto, la pirámide de automatización de pruebas [Cohn] es útil. Muestra cómo maximizar el valor de la automatización de las pruebas de regresión, empezando por las pruebas unitarias en el nivel más bajo de la pirámide y pasando a las pruebas de nivel de servicio.



Las pruebas de interfaz de usuario se encuentran en la parte superior. Las pruebas unitarias son rápidas y confiables. La capa de servicio permite probar la lógica de negocio en la API o en el nivel de servicio, donde la interfaz de usuario (IU) no lo impide. Cuanto más alto sea el nivel, más lenta y más frágiles son las pruebas.

Los criterios de entrada, normalmente parte de un enfoque de pruebas definido (práctica específica 2.3) no son aplicables al desarrollo Ágil. Dentro del desarrollo de software Ágil, las pruebas son una parte integral del proceso del equipo y una actividad casi continua. En consecuencia, no hay necesidad de una lista de verificación específica o puerta de enlace para determinar si las pruebas pueden o no pueden iniciarse. Esto también se aplica a un componente que va dentro de un equipo Ágil desde una etapa de prueba (por ejemplo, pruebas unitarias) a otra (por ejemplo, pruebas de aceptación).

Los criterios de salida de las pruebas (práctica específica 2.4) forman parte de la llamada "Definition of Done" (DoD). Es importante que el DoD tenga criterios específicos relacionados con la prueba, por ejemplo, para la cobertura de las pruebas y la calidad del producto (defectos). La iteración debe resultar en la implementación del conjunto acordado de historias de usuarios y cumplir con los criterios (de pruebas) de salida como se define en el DoD. Normalmente las historias que no cumplen con los criterios de salida se ponen en el backlog y pueden tratarse dentro de la siguiente iteración. Por supuesto, en Ágil no hay criterios de salida para un componente que va de una etapa de pruebas a otra. No sólo existirá una Definición de Hecho en el nivel de iteración, a menudo también hay un DoD en el nivel de liberación que abarca múltiples iteraciones. El DoD en el nivel de entrega de nuevo tendrá normalmente criterios relacionados con la cobertura y la calidad del producto.

La práctica específica 2.5 Definir los criterios de suspensión y reanudación no es aplicable a los ciclos de vida ágiles. Dado que las pruebas son una parte integral del proceso de desarrollo de software Ágil, por supuesto no se tratará como una actividad separada e independiente de otras actividades de la iteración. Cuando hay problemas de

bloqueo que podrían considerarse como amenazas potenciales o reales para el progreso de las pruebas, éstas se discuten en la reunión diaria. En esta reunión el equipo decidirá qué acciones, si las hay, deben ser tomadas para resolver los problemas. Por lo tanto, los criterios formales de suspensión y reanudación no serán requeridos ni definidos, las cuestiones relacionadas con esto se tratan como parte de la rutina Ágil normal. La rutina ágil sirve así como una práctica alternativa para esta práctica específica.

2.2.3 SG3 Establecer la Estimación de las Pruebas

Para los equipos Ágiles, se harán estimaciones detalladas de las pruebas durante la planificación de la iteración. Las estimaciones de alto nivel (de pruebas) se hacen durante la planificación de la liberación y posiblemente también en las sesiones de refinamiento del “backlog”. Todas las estimaciones se hacen, por supuesto, como una estimación de equipo que incluye todo el esfuerzo necesario para entregar cada historia. Es importante garantizar que las actividades de pruebas se tengan en cuenta durante las sesiones de estimación. Esto puede realizarse, haciendo que las actividades de pruebas se identifiquen como tareas separadas y posteriormente se estime cada tarea de pruebas de forma individual o estimando cada historia de usuario en la que se explique y se tenga en cuenta las pruebas a realizar. Un probador, que forma parte de un equipo Ágil, participará en las sesiones de estimación. Planificación de póker o tamaño de camisa son típicas técnicas de estimación que se utilizan en el desarrollo de software Ágil.



El trabajo a realizar para la siguiente iteración se define normalmente por las historias de usuario. Las historias de usuarios necesitan ser de tamaño pequeño para que sean estimables. Estimable es uno de los criterios de la historia de usuario definidos por INVEST [Wake] y es aplicable a las historias de usuarios que formarán parte de una iteración. Durante la planificación de la iteración, los equipos ágiles normalmente identificarán y definirán las tareas relacionadas con las pruebas. Las tareas de pruebas se capturan en un tablero de tareas junto con las otras tareas de desarrollo. Este conjunto de tareas es la base para la estimación de la iteración. El conjunto de tareas definidas para la próxima iteración sirve como una especie de estructura de

trabajo-ruptura (como se utiliza en proyectos secuenciales)

En la planificación de la liberación, las historias de los usuarios o las grandes historias de usuarios (epic's) suelen ser definidas a más alto nivel y aún no detalladas en tareas específicas. Esto, por supuesto, hará que las estimaciones sean más difíciles y menos precisas. Como se ha indicado, los proyectos Ágiles no establecerán una estructura de desglose de trabajo como base para las estimaciones, pero pueden beneficiarse a nivel de liberación de un simple diagrama que visualiza el producto a desarrollar y, por lo tanto, define adecuadamente el alcance del trabajo.

Aunque un equipo Ágil estimará de una manera relativamente informal, la justificación de las estimaciones debe ser clara, es decir, qué factores se están considerando. Las discusiones basadas en la justificación promueven un mayor nivel de exactitud de las estimaciones. Normalmente, en los proyectos Ágiles la estimación se centra en el tamaño (utilizando puntos de la historia) o en el esfuerzo (utilizando día hombre ideal como unidad de estimación). Los costos normalmente no se abordan como parte de las sesiones de estimación en los proyectos Ágiles.

Este objetivo específico de TMMi y sus prácticas específicas es por lo tanto plenamente aplicable con la excepción de la práctica específica 3.2 Definir el ciclo de vida de las pruebas. Uno de los fundamentos del desarrollo de software iterativo y ágil es trabajar en trozos pequeños. Por lo tanto, las tareas que se han identificado suelen ser lo suficientemente detalladas como para servir de base para la estimación (de las pruebas). De este modo, no es necesario en Ágil definir también un ciclo de vida para las actividades de pruebas que sirva como una base adicional para la estimación.

2.2.4 SG4 Desarrollar un Plan de Pruebas

Reafirmando el comentario de que la planificación de la prueba es sobre el pensamiento inicial ("la actividad") y no sobre la definición del plan de pruebas asociado ("el documento"). En Ágil, la mayoría de las actividades de planificación de las pruebas, tal como se define en este objetivo específico de TMMi, se realizarán durante la planificación de la liberación de la iteración. Sin embargo, el resultado de estas actividades normalmente no se documentará en un plan de prueba, especialmente para la planificación de la iteración, donde podrían reflejarse en el tablero de tareas.

Dado que las pruebas se realizan como parte integral de la planificación de la liberación y de la iteración, las "planificaciones" resultantes también incluirán actividades de pruebas. En lugar de un calendario detallado como el desarrollado con ciclos de vida secuenciales, el calendario dentro de un proyecto Ágil es mucho más parecido a un ordenamiento de historias de usuarios (tareas pendientes) y tareas que refleja las prioridades de la entrega y de la iteración, por ejemplo, basado en la entrega deseada del valor del negocio. Los mapas mentales se utilizan a menudo aquí como técnica de apoyo. El tablero de tareas reflejará las prioridades de la iteración. Por lo tanto, se establece un calendario no explícito (de pruebas); se espera que se definan prioridades de liberación y de iteración claras para las historias de usuarios, las respectivas tareas que se van a realizar, incluidas las tareas de pruebas.

A nivel de proyecto, el personal de pruebas es parte del equipo de construcción; se identifica por adelantado la necesidad de disponer de recursos de pruebas o de múltiples habilidades. A medida que los proyectos cambian o crecen, uno puede fácilmente olvidar la justificación para la selección inicial de un individuo para un equipo/proyecto dado, que a menudo incluye necesidades de habilidades específicas o experiencia específicamente relacionada con el equipo/proyecto. Esto proporciona una buena justificación para registrar las necesidades de habilidades de las personas, proporcionando información de respaldo relacionada con la razón por la que fueron seleccionados para el equipo/proyecto. Una vez definidos los equipos Ágiles, la dotación de personal de pruebas es más o menos fija. Durante la planificación de la iteración, la identificación de los recursos y habilidades (adicionales), por ejemplo, para las pruebas no funcionales, necesarias para realizar las pruebas en una iteración, puede ser discutida si es necesario para asegurar que el equipo tenga suficientes recursos, conocimientos y habilidades para realizar las pruebas requeridas.

El "Scrum Master" debe asegurarse de que el propietario del producto proporcione la entrada a las pruebas según sea necesario, por ejemplo, respondiendo preguntas y manteniendo conversaciones sobre las historias de usuarios. El propietario del producto debe estar lo suficientemente disponible, lo que debería nuevamente ser organizado por adelantado.

Una sesión inicial de riesgo del proyecto debe ser parte de la planificación de la liberación y de la iteración. La identificación (y la gestión) de los riesgos adicionales del proyecto durante la iteración es parte de las reuniones diarias y se documentan normalmente por medio de un registro de impedimentos. Es importante que también se anoten los problemas de las pruebas en el registro de impedimentos. Los impedimentos deben ser discutidos en la reunión diaria, hasta que se resuelvan.

Aunque no se ha desarrollado y documentado ningún plan de pruebas específico y detallado, la práctica específica 4.5 Establecer el plan de pruebas sigue siendo relevante dentro de un contexto Ágil. El resultado de las discusiones que tienen lugar en el contexto de la planificación de las pruebas, sin embargo, es probable que se capturan en una forma liviana, posiblemente un mapa mental.

2.2.5 SG 5 Obtener compromiso con el Plan de Pruebas

Dentro de Ágil, el proceso para desarrollar y establecer un enfoque de pruebas y un plan de pruebas es un ejercicio basado en el equipo, posiblemente conducido por un profesional de pruebas (siendo uno de los miembros del equipo). La calidad del producto es una responsabilidad del equipo. Como tal, siempre que el equipo siga el proceso correcto, el compromiso con el enfoque (de pruebas) y el plan (de pruebas) es un resultado implícito de la planificación de la liberación y de la iteración, ya que es un esfuerzo de equipo. Esto, por supuesto, es una gran diferencia con la forma de trabajar en un entorno tradicional en el que típicamente el probador prepara el plan de pruebas y luego necesita obtener un compromiso explícito.

En los proyectos Ágiles, el equipo Ágil (incluyendo el propietario del producto) debe entender y acordar la lista de riesgos de producto priorizada y las acciones de mitigación de pruebas que se deben realizar. La comprensión y el compromiso pueden lograrse, por ejemplo, mediante una breve presentación seguida de una discusión durante la planificación de liberación o de la iteración explicando los riesgos del producto, el enfoque de las pruebas y su justificación para el equipo.

Durante la sesión de estimación (ver SG 3 Establecer Estimaciones de Pruebas) se estima la carga de trabajo. Los recursos (de pruebas) para un equipo son fijados por la configuración del equipo Ágil. Las historias de usuarios estimadas y seleccionadas para ser desarrolladas toman los recursos disponibles como punto de partida (restricción). Así, conciliar los niveles de trabajo y de recursos no es una actividad significativa. La práctica específica 3.2 Conciliar los niveles de trabajo y recursos es, por lo tanto, una práctica que típicamente no es relevante en un contexto Ágil. Se realizarán reuniones diarias para resolver cualquier problema de recursos durante la iteración y para (re)asignar los recursos apropiados de inmediato, o eliminar una entrega de una iteración que se agregará en una futura planificación de la iteración o de la liberación.

2.3 Área de Proceso 2.3 Monitorización y Control de las Pruebas

El propósito de la Monitorización y Control de las Pruebas es proporcionar una comprensión del progreso de las pruebas y la calidad del producto para que las acciones correctivas apropiadas puedan ser tomadas cuando el progreso de las pruebas se desvíe significativamente del plan de calidad y del producto.

Siguiendo el manifiesto Ágil y sus principios de acompañamiento, hay algunas cosas que son fundamentalmente diferentes a la monitorización y control en los proyectos Ágiles en comparación con los proyectos tradicionales. Aunque la monitorización y el control son elementos esenciales de un proyecto ágil, no implica que se adhiera a un plan rígido, de hecho, es cierto lo contrario ya que tanto el manifiesto como los principios hablan de darle la bienvenida al cambio.

Desde una perspectiva de la monitorización y el control de las pruebas, esto significa que no estamos orientados por el plan, sino que revisamos constantemente nuestro progreso y los resultados de las pruebas y adaptamos nuestro plan y enfoque según sea apropiado. El plan de pruebas, sin embargo, es una entidad viva y necesita continuamente ser revisada y actualizada a medida que se dispone de información nueva o se da retroalimentación. Los proyectos ágiles también necesitan tener en mente el "panorama más amplio" y supervisar y controlar a nivel más alto (versión), así como a nivel de iteración.

Es importante tener en cuenta que como la prueba es un proceso que está totalmente integrado en el proceso general del equipo Ágil, la monitorización y el control de las pruebas también es parte integral de los mecanismos generales de monitoreo y control del equipo Ágil. Como resultado, los probadores no reportan a un gestor de pruebas como a los proyectos tradicionales, sino al equipo.

2.3.1 SG1 Monitorizar el Avance de las Pruebas Respecto al Plan

Los probadores de los equipos Ágiles utilizan varios métodos para supervisar y registrar el progreso de las pruebas, por ejemplo, el avance de las tareas de pruebas y las historias en el tablero de tareas Ágil y los gráficos de trabajos pendientes. Éstos se pueden comunicar al resto del equipo utilizando medios tales como cuadros de mando wiki y correos electrónicos de estilo tablero, así como verbalmente durante las reuniones. Los equipos pueden usar gráficos de trabajo pendiente tanto para realizar el seguimiento del progreso en toda la versión completa como para el seguimiento dentro de cada iteración. Un gráfico de trabajo pendiente mostrará normalmente progreso contra la velocidad esperada y la acumulación de características que se implementarán (rendimiento del equipo). A veces, cuando los recursos del entorno de pruebas son escasos y vitales, por ejemplo, para las pruebas no funcionales, se utilizan gráficos de trabajos pendientes específicos para supervisar el uso de los recursos del entorno de pruebas contra los acordados durante la planificación de la iteración. Otra práctica común es realizar un seguimiento de los problemas del entorno de pruebas a través del tablero de tareas, por ejemplo, mediante el uso de pegatinas marcadas como "entorno de pruebas bloqueado" en las tarjetas de historias o mediante la creación de una columna separada en el tablero donde todas las historias bloqueadas por entornos esperan hasta desbloquearse. Se trata de crear visibilidad del impacto de un entorno de pruebas que está bloqueando el progreso.

Para proporcionar una representación visual instantánea y detallada del estado actual de todo el equipo, incluido el

estado de las pruebas, los equipos pueden utilizar tableros de tareas Ágil. Las tarjetas de historias, las tareas de desarrollo, las tareas de pruebas y otras tareas creadas durante la planificación de la iteración se capturan en el tablero de tareas, a menudo utilizando tarjetas coordinadas por color para determinar el tipo de tarea. Durante la iteración, el progreso se gestiona a través del movimiento de estas tareas a través del tablero de tareas en columnas tales como pendiente de hacer, trabajo en progreso y hecho. Los equipos ágiles pueden utilizar herramientas para mantener sus tarjetas de historial en los tableros de tareas de Ágil, que pueden automatizar los paneles y la visión general del estado. Sin embargo, algunos equipos no crean tareas específicas para las actividades individuales, pero pueden usar la tarjeta de historia y anotar comentarios en esta tarjeta para pruebas, referenciando herramientas Ágiles o wiki donde las pruebas pueden ser documentada.

Las tareas de pruebas en el tablero de tareas normalmente se relacionan con los criterios de aceptación definidos para las historias de usuario. A medida que las secuencias de comandos de automatización de pruebas, las pruebas guionizadas manuales y las pruebas exploratorias para una tarea de pruebas alcanzan el estado de pasó, la tarea se desplaza a la columna finalizada de la tabla de tareas.

La Definición de Hecho sirve como criterio de salida para medir el progreso. El DoD también debe estar relacionado con las actividades de pruebas y muestra todos los criterios que deben satisfacerse antes de que las pruebas de una historia de usuario puedan llamarse 'Hechas'.



Hay que tener en cuenta que los criterios de pruebas relacionados con las tareas de pruebas forman sólo una parte de lo que el equipo acuerda completar como la Definición de Hecho.

El criterio de Definición de Hecho se aplica normalmente en múltiples niveles, por ejemplo, a nivel de iteración y nivel de liberación. Todo el equipo revisa el estado del tablero de tareas regularmente, a menudo durante las reuniones diarias de "stand-up", para asegurarse que las tareas se están moviendo a través del tablero con una velocidad aceptable. Si las tareas (incluidas las tareas de pruebas) no se están moviendo o se están moviendo con demasiada lentitud, desencadena un debate en equipo en el que se analizan los problemas que pueden bloquear el progreso de esas tareas.

La reunión "stand-up" diaria incluye a todos los miembros del equipo Ágil incluyendo a los probadores. En esta reunión, comunican su estado actual y progreso real al resto del equipo. Cualquier problema que pueda bloquear el progreso de las pruebas se comunica durante las reuniones diarias de "stand-up", por lo que todo el equipo es consciente de los problemas y puede actuar en consecuencia. De esta manera también se integra la gestión de riesgos en estas reuniones diarias. Cualquier riesgo del proyecto, incluyendo aquellos para las pruebas, por ejemplo, la falta de disponibilidad en ambientes de pruebas puede ser comunicado y dirigido durante el stand-up diario. (Nota: la monitorización de los riesgos del producto es parte de la monitorización de la calidad del producto y por lo tanto se discute en lo sucesivo con el objetivo específico SP 2 Monitorización de la Calidad del Producto contra Plan y Expectativas.

Las reuniones diarias para la gestión diaria de tareas y las prácticas del equipo Ágil relacionadas con la corrección del curso de tareas son buenas técnicas probadas que cumplen los objetivos de las prácticas específicas de TMMi en el área de proceso de Monitorización y Control de Pruebas.

La revisión de los hitos en Ágil se encuentra en la finalización de una iteración. Los logros de las pruebas, por ejemplo, en relación a la Definición de Hecho, formarán parte de la revisión de iteración. Las demostraciones se organizan con las partes interesadas para discutir el valor comercial y la calidad del producto que se entrega.

Las partes interesadas están representadas por el propietario del producto en la planificación de la iteración, las revisiones de iteración (demos) y las retrospectivas. El propietario del producto está involucrado a través de discusiones sobre el backlog y proporcionará retroalimentación e información para la elaboración de historias de usuarios y el diseño de pruebas a lo largo de la iteración. Otras partes interesadas están involucradas al final de cada iteración durante la revisión de iteración. No se requiere una monitorización específica de la participación de

las partes interesadas, ya que la representación de las partes interesadas por parte del propietario de un producto está incluida en la forma ágil de trabajar. Hay que tener en cuenta que la participación activa del propietario del producto en los proyectos Ágiles es un factor de éxito crucial, pero a veces es bastante difícil de realizar en la práctica. Si este es el caso, esto será reportado y discutido en la reunión diaria y gestionado como un riesgo de proyecto (ver arriba) que afectará a la efectividad y eficiencia del equipo en su conjunto

2.3.2 SG2 Monitorización de la Calidad del Producto Contra el Plan y las Expectativas

Para la monitorización de la calidad del producto se utilizan en gran parte los mismos mecanismos que para la monitorización del progreso (ver SG1 arriba). En Ágil, para la monitorización del riesgo del producto, la atención se centra en una revisión de la lista de riesgos del producto en reuniones regulares en lugar de revisar cualquier documentación detallada de los riesgos. Se discutirán los riesgos del producto recientemente identificados o los riesgos de los cambios en los productos, por ejemplo, como resultado de pruebas exploratorias, y se acordarán las acciones de pruebas requeridas. El estado de los diversos riesgos del producto se muestra típicamente por medio de gráficos en el tablero de instrumentos.

Una buena práctica es que ninguna característica se considera realizada hasta que se ha integrado y probado con éxito en el sistema. En algunos casos, las iteraciones de refinamiento o estabilización ocurren periódicamente para resolver cualquier defecto persistente y otras formas de deuda técnica. Los equipos ágiles usan métricas basadas en defectos similares a las capturadas en metodologías de desarrollo tradicionales, tales como las tasas de pruebas pasadas/falladas, las tasas de descubrimiento de defectos, los defectos encontrados y resueltos, para monitorizar y mejorar la calidad del producto. El número de defectos encontrados y resueltos durante una iteración, así como el número de defectos no resueltos y que posiblemente se conviertan en parte del "backlog" para la próxima iteración, deben ser monitorizados durante las reuniones diarias de "stand-up". Para monitorizar y mejorar la calidad general del producto, muchos equipos Ágiles también usan encuestas de satisfacción del cliente para recibir comentarios sobre si el producto cumple con sus expectativas.

Los criterios de salida de las pruebas, por ejemplo, para la cobertura de las pruebas y la calidad del producto (defectos), son parte de la "Definition of Done" (DoD). La adherencia a los criterios de salida convenidos se supervisa normalmente a través del tablero de tareas, por lo que una historia sólo puede ser indicada como "hecha" si cumple con sus criterios de DoD.

La reunión diaria "stand-up" es el mecanismo utilizado para realizar casi continuamente revisiones de calidad del producto. La revisión de la calidad del producto en Ágil se encuentra en la terminación de una iteración. Las demostraciones se organizan con las partes interesadas para discutir el valor comercial y la calidad del producto que se entrega. La calidad del producto se verifica y valida en función de los criterios de calidad definidos en el DoD.

Siguiendo el área de proceso Planificación de las Pruebas, no se aplican prácticas específicas sobre la monitorización de los criterios de entrada, suspensión y reanudación. Para obtener más información, consulte el objetivo específico 2 Establecer un Enfoque de Pruebas del área de proceso de Planificación de las Pruebas, donde se ha explicado por qué estos criterios normalmente no son aplicables en un entorno Ágil.

2.3.3 Gestionar Acciones Correctivas hasta su Cierre

Los equipos ágiles advierten rápidamente problemas como las desviaciones de las expectativas en un gráfico de trabajos pendientes y/o la falta de avance de las tareas (de pruebas) y las historias en el tablero de tareas Ágil. Estas y otras cuestiones, por ejemplo, cuestiones que bloquean el avance de las pruebas, se comunican durante las reuniones diarias de "stand-up" (véase más arriba), por lo que todo el equipo es consciente de los problemas. Esto desencadena una discusión en equipo donde se analizan los problemas. El resultado podría ser que la línea base necesita una actualización (por ejemplo, la eliminación de una o más historias de usuarios del alcance de la iteración), la Definición de Hecho es quizás demasiado estricta o deben hacerse cambios en la forma de trabajar. El equipo decidirá las acciones correctivas que se deben tomar junto con el propietario del producto. En los casos en que se modifica el "backlog" de la iteración, el propietario del producto presenta un "backlog" de iteración actualizado al equipo y los elementos eliminados de la iteración se llevan a la siguiente iteración y se discuten durante la planificación de la misma.

La gestión de acciones correctivas en los proyectos Ágiles es principalmente responsabilidad del equipo autoorganizado. El equipo puede definir e implementar acciones correctivas apropiadas, o escalar cualquier problema como "impedimento" al Scrum Master. El Scrum Master normalmente tiene la responsabilidad de apoyar al equipo para gestionar los problemas hasta el cierre de los mismos. Los eventos típicos para discutir y gestionar las acciones correctivas son las reuniones diarias y reuniones retrospectivas. Las acciones correctivas que se han acordado pueden ser gestionadas hasta el cierre como "tareas" o "elementos pendientes" a través del backlog o (dentro de la iteración) a través del tablero de tareas.

2.4 Área de Proceso 2.4 Diseño y Ejecución de Pruebas

El propósito del Diseño y Ejecución de Pruebas es mejorar la capacidad del proceso de pruebas durante el diseño y la ejecución de las pruebas estableciendo la especificación del diseño de las pruebas, utilizando técnicas de diseño de pruebas, un proceso estructurado de ejecución de pruebas y la administración de las incidencias de pruebas hasta el cierre.

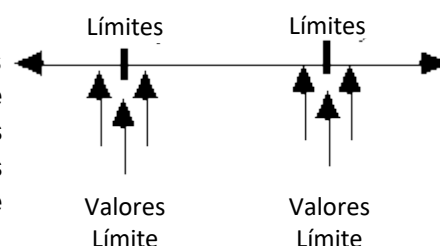
Aunque el objetivo subyacente ("mitigar los riesgos y probar el software") es el mismo para un proyecto Ágil y un proyecto secuencial tradicional, el enfoque adoptado sobre cómo probar es normalmente muy diferente. En Ágil, la flexibilidad y la capacidad de responder al cambio son puntos de partida importantes. Además, el análisis y el diseño de las pruebas, la implementación y la ejecución de las pruebas no son fases posteriores, sino que se realizan en paralelo, superpuestas e iterativamente. El nivel de detalle de la documentación de pruebas establecida es otra diferencia clave. Normalmente, las técnicas basadas en la experiencia y las basadas en los defectos se utilizan con mayor frecuencia en los proyectos Ágiles, aunque las técnicas de diseño de pruebas basadas en especificaciones también pueden seguir siendo aplicables y utilizadas. Con más énfasis en las pruebas unitarias y las pruebas de integración, las técnicas de caja blanca, como tablas de decisión y estados también son mucho más populares. Una última diferencia importante es el nivel del riesgo de la regresión, que requiere más pruebas de regresión en los diferentes niveles de pruebas. Lo ideal es que las pruebas de regresión estén altamente automatizadas. Existen muchas diferencias, pero al final, en el contexto del área de proceso de Diseño y Ejecución, siempre se trata de crear pruebas, mitigar los riesgos de los productos, realizar pruebas y encontrar defectos.

2.4.1 SG1 Realizar el Análisis y Diseño de las Pruebas utilizando Técnicas de Diseño de Pruebas

En Ágil, el análisis y diseño y la ejecución de las pruebas son actividades de apoyo mutuo que normalmente se ejecutan en paralelo a lo largo de una iteración. En proyectos de ciclo de vida secuencial, el análisis de pruebas se lleva a cabo por los probadores revisando la base de las pruebas, por ejemplo, los requisitos y evaluando la capacidad de pruebas de la base de pruebas una vez que se ha creado. En los proyectos Ágiles, los probadores forman parte de un equipo que crea y refina colectivamente historias de usuarios. Se llevan a cabo revisiones informales frecuentes mientras se desarrollan los requisitos, incluyendo para cada historia de usuario los criterios de aceptación. Estos criterios se definen en colaboración entre representantes del negocio, desarrolladores y probadores. Normalmente, la perspectiva única del probador mejorará la historia del usuario al identificar los detalles que faltan y hacerlos verificables. Un probador puede contribuir preguntando a los representantes de negocios preguntas abiertas sobre la historia del usuario y sus criterios de aceptación y proponiendo maneras de probar la historia del usuario. Por lo tanto, el análisis de pruebas no es una actividad explícita por separado, sino una actividad implícita que los probadores desempeñan como parte de su papel en el desarrollo colaborativo de la historia del usuario.

Basándose en el análisis de las historias de usuarios, se identifican las condiciones de las pruebas. Desde una perspectiva de pruebas, la base de pruebas se analiza para ver lo que podría ser probado - estas son las condiciones de pruebas. Las condiciones de pruebas (a veces denominadas situaciones de pruebas) son básicamente una identificación de "cosas" que necesitan ser probadas/cubiertas [Black, Van Veenendaal].

Siempre que los criterios de aceptación definidos sean detallados y suficientemente claros, pueden asumir el papel de las condiciones de pruebas tradicionales.



Las condiciones de pruebas se traducen posteriormente en pruebas, que son necesarias para proporcionar cobertura de los criterios de aceptación definidos y acordados.

A menudo también es beneficioso realizar análisis de pruebas en un nivel superior en lugar de sólo las historias de los usuarios. Por ejemplo, analizar una característica o épica o una colección de historias para identificar condiciones de pruebas que son más abstractas que las del nivel de la historia del usuario y también abarcan varias historias de usuarios.

Las técnicas de diseño de pruebas basadas en especificaciones suelen ser útiles para derivar condiciones de pruebas de historias de usuarios y criterios de aceptación. Sin embargo, en Ágil la mayoría de las veces estas técnicas de diseño de pruebas se utilizan de forma más implícita que explícita mediante la cual los probadores basados en su experiencia han dominado las técnicas y son capaces de usarlas con flexibilidad en su contexto.

Las condiciones de pruebas se documentarán en un formato liviano, lo contrario a la forma más tradicional de trabajo donde se documentan como parte de un documento de especificación de diseño de pruebas. A veces, especialmente en los proyectos donde las pruebas exploratorias son ampliamente utilizadas, las condiciones de pruebas se identifican por medio de una lluvia de ideas y se documentan como ideas de pruebas (para formar parte de las charlas de pruebas). La definición de las condiciones de pruebas es también la base para establecer la trazabilidad horizontal (véase más adelante) y la gestión de la cobertura del conjunto de pruebas de regresión (automatizadas) mediante las cuales se desarrollan pruebas automatizadas para cubrir condiciones de pruebas específicas

Con el primer principio de pruebas aplicado con Ágil, se identificarán las pruebas que cubren el conjunto de condiciones de pruebas (y posiblemente se automatizarán) antes o al menos en paralelo con el desarrollo del código. Para las pruebas de unidades automatizadas se puede considerar un enfoque como el “Desarrollo Dirigido por Pruebas”. Para niveles de pruebas más altos, el desarrollo impulsado por comportamientos (BDD) y el desarrollo impulsado por pruebas de aceptación (ATDD) son enfoques de desarrollo ágil populares que también admiten las pruebas. Tanto BDD como ATDD también están altamente relacionados con la automatización de pruebas.

Para la mayoría de las pruebas manuales, las pruebas serán identificadas/refinadas a medida que el equipo avanza con la ejecución de las pruebas. Las pruebas a menudo no están documentadas con tanto detalle como en proyectos tradicionales, sino más bien en un formato de ideas de pruebas usando pruebas exploratorias. Para las áreas complejas y críticas un enfoque más tradicional para el diseño de pruebas y el desarrollo de casos de pruebas utilizando técnicas de diseño de pruebas formal puede ser la mejor manera de cubrir los riesgos.

Sin embargo, también con este enfoque la cantidad de documentación será limitada en comparación con las pruebas en un entorno de ciclo de vida secuencial tradicional. La priorización de las pruebas sigue a la priorización de la historia de usuario que se está cubriendo. La priorización de las historias de usuarios se basa en el valor del negocio; la mayor prioridad se relaciona con el valor comercial más alto. Es importante que las pruebas se establezcan para cubrir riesgos funcionales y no funcionales, pero especialmente en Ágil también específicamente para cubrir el riesgo de la regresión.

Se identifican los datos de pruebas específicos necesarios para soportar las condiciones de pruebas y la ejecución de las pruebas. Sin embargo, en Ágil en contraste con los entornos tradicionales, los datos de pruebas necesarios normalmente no se especifican por primera vez como parte de un documento de especificación de pruebas. Más bien, siempre que las herramientas y/o funcionalidades necesarias estén disponibles, los datos de pruebas se crean inmediatamente para permitir un inicio casi inmediato de la ejecución de las pruebas manuales. Sin embargo, para las pruebas automatizadas, normalmente se necesitarán datos que se especifican por adelantado.

Es necesario establecer y mantener la trazabilidad entre los requisitos, las condiciones de pruebas y las pruebas. Los equipos necesitan dejar claro que han cubierto las diversas historias de usuarios y criterios de aceptación como parte de sus pruebas. Si bien una herramienta formal de gestión de requisitos puede no ser necesaria, el equipo necesita apoyo para que sus requerimientos sean organizados y gestionados de una manera que apoye la asignación de identificadores a cada requisito para que los identificadores puedan ser utilizados acorde a los criterios acordados. En muchas organizaciones Ágiles, las historias de usuarios son la base para desarrollar un conjunto de criterios de aceptación relacionados y pruebas posteriores. Una vez establecidas estas pruebas, las pruebas mismas a menudo se convierten en los requisitos acordados detallados. El uso de este enfoque puede ser suficiente para

lograr la intención de la práctica específica de TMMi sobre la gestión de requisitos relacionados con la trazabilidad horizontal.

2.4.2 SG2 Realizar la Implementación de las Pruebas

En la implementación de las pruebas se trata de obtener todo lo que se necesita para iniciar la ejecución de las pruebas. Normalmente, se minimiza el desarrollo de la documentación de pruebas, por ejemplo, procedimientos de pruebas, para dar apoyo a su ejecución. Se desarrollan y priorizan guiones de pruebas (regresión) bastante automatizados. También la preparación de las pruebas de regresión comienza lo antes posible en paralelo con otras actividades de pruebas.

En un entorno Ágil, los procedimientos de pruebas detallados no son una práctica común. Trabajando en un equipo bien informado, las pruebas probablemente se documentarán a un nivel mucho más alto de abstracción. Esto será suficiente para aquellos que ejecutan las pruebas ya que se espera que tengan el nivel requerido de dominio y conocimientos técnicos. Aquellos que ejecutan las pruebas trabajan dentro del equipo, por lo que tendrán una mejor comprensión de qué y cómo se codifica, y cómo la funcionalidad es apta para el propósito. Dado que el nivel de cambio dentro y entre iteraciones es normalmente alto, el desarrollo de un procedimiento de pruebas detallado también crearía mucho trabajo de mantenimiento. Normalmente, muchas pruebas se realizan mediante pruebas exploratorias. En pruebas exploratorias, no se desarrollan procedimientos de pruebas detallados, sino más bien comentarios de pruebas de una página de alto nivel que describen ideas de prueba y orientación para el probador. Sin embargo, los datos de pruebas específicos que se requieren para realizar pruebas, por supuesto, deben crearse por adelantado.

Muchos equipos Ágiles utilizan pruebas automatizadas. Los enfoques típicos que se utilizan son el desarrollo dirigido por pruebas (TDD), el desarrollo dirigido por comportamientos (BDD) y el desarrollo dirigido por pruebas de aceptación (ATDD). Utilizando uno o más de estos enfoques se crearán scripts de pruebas automatizados como parte de la actividad de implementación de las pruebas.

La integración continua es una práctica clave para los proyectos Ágiles. Las pruebas es una actividad paralela y totalmente integrada que debe realizar el equipo. No es una actividad o fase separada independiente. Por lo tanto, la práctica específica SP2.3 Especificar el Procedimiento de Pruebas de Admisión se vuelve irrelevante dentro de un equipo Ágil. No hay necesidad y de hecho no tiene sentido, tener una prueba específica de admisión formal de cómo se especifica y realiza. Sin embargo, en caso de que algunas pruebas, por ejemplo, pruebas de integración de sistemas o pruebas de integración de hardware/software, se realicen fuera del ámbito del equipo Ágil, pueden especificar una prueba de admisión para evaluar la calidad de salida del equipo Ágil y determinar si el producto que se ha entregado está listo para su prueba.

En ambientes ágiles no se creará ningún programa específico de ejecución de pruebas y, por lo tanto, la práctica específica 2.4 Desarrollar el Calendario de Ejecución de Pruebas no es aplicable. Existe un alto nivel de flexibilidad con respecto al orden en que se ejecutarán las pruebas a lo largo de una iteración, aunque ésta será guiada por la prioridad de las distintas historias de usuarios. Las diferentes pruebas a ejecutar se identificarán durante la planificación de la iteración y se administrarán como tareas de pruebas a través del tablero de tareas Ágil.

2.4.3 SG3 Realizar la Ejecución de las Pruebas

Por supuesto, los elementos de software que se están desarrollando en la iteración deben ser probados. Siguiendo el manifiesto Ágil, como con los otros objetivos en esta área de proceso, esto se hará típicamente con mucho menos intensidad de documentación que en un proyecto secuencial tradicional. A menudo no se producen procedimientos de pruebas detallados y registros de pruebas. Las pruebas, especialmente las pruebas de regresión deberían ser automatizadas tanto como sea posible y factible.

En un proyecto Ágil, el software se entrega, como mínimo, diariamente y las pruebas son una parte integral del proceso de desarrollo de Ágil mediante el cual el trabajo se realiza en cooperación mutua para entregar un producto de calidad. No hay un equipo dedicado y separado de desarrollo y pruebas. Por lo tanto, una prueba de entrada específica como se define en la práctica específica 3.1 Realizar pruebas de entrada no es aplicable en un ambiente Ágil puro. Sin embargo, cuando algunas pruebas se realizan fuera del equipo Ágil (como ocurre a menudo), el equipo de pruebas que realiza las pruebas fuera del equipo Ágil realizará normalmente una prueba de entrada. Tenga en

cuenta que se espera que las actividades se completen antes de que empiecen las pruebas de aceptación de una historia de usuario específica dentro del equipo Ágil. Estas actividades son normalmente parte de la Definición de Hecho, por ejemplo, se ha completado y aprobado la cantidad acordada de pruebas unitarias

La ejecución de pruebas se realiza alineada con las prioridades definidas durante la planificación de la iteración. Algunas pruebas se pueden ejecutar utilizando un procedimiento de pruebas documentado, pero normalmente muchas pruebas se ejecutarán utilizando las pruebas exploratorias y basadas en un marco de pruebas. En las pruebas exploratorias, el diseño y la ejecución de las pruebas ocurren al mismo tiempo, regidas por una guía de pruebas preparada. Una guía de pruebas provee objetivos de pruebas y condiciones de pruebas para cubrir durante una sesión de pruebas en tiempo real. Durante las pruebas exploratorias, los resultados de las pruebas más recientes guían las próximas pruebas.

Con el desarrollo iterativo hay una mayor necesidad de organizar y estructurar pruebas de regresión. Esto se hace a veces manualmente, pero preferentemente se hará mediante pruebas de regresión automatizadas y herramientas de apoyo. Las pruebas de regresión en las pruebas unitarias y de integración suelen ser parte de un proceso de integración continua. La integración continua es una práctica importante dentro del desarrollo de software Ágil. Es básicamente un proceso automatizado de construcción y pruebas que se lleva a cabo al menos una vez al día y detecta defectos temprano y rápidamente. La integración continua permite ejecutar pruebas de regresión automatizadas con regularidad y enviar un rápido “feedback” al equipo sobre la calidad del código y la cobertura de código alcanzada.

Las incidencias que se encuentran durante las pruebas pueden ser registradas y reportadas por el equipo. Normalmente, existe un debate en los proyectos Ágiles si todas las incidencias encontradas deben estar registradas. Particularmente en el caso en que el equipo está situado en el mismo lugar, los probadores necesitan hablar con los desarrolladores de las incidencias/defectos que se pueden arreglar de inmediato e incluirse en la siguiente compilación no necesitarán ser registrados, sólo necesitan ser arreglados. En Ágil generalmente no todas las incidencias encontradas serán registrados y gestionados en consecuencia. Como se ha dicho, muchas de ellas son planteadas informalmente por un miembro del equipo (que puede ser un probador) y son corregidos inmediatamente por el desarrollador que inyectó el defecto. Algunos datos se pierden, pero la sobrecarga al mismo tiempo se reduce. Algunos equipos sólo registran incidencias que se escapan a la iteración, algunos los registran si no se pueden arreglar hoy, algunos sólo registran incidencias de alta prioridad. Si no se registran todas las incidencias detectadas, los criterios deben estar disponibles para determinar qué incidencias deben registrarse y cuáles no. Hay que recordar que, la intención es lograr el defecto correctamente corregido, no el registro del mismo. Un ejemplo de tal criterio: "Si un defecto puede ser resuelto antes de la siguiente reunión y la construcción diarias, no necesita ser documentado. Cualquier defecto planteado que no pueda ser cerrado dentro del día, o afecte a una parte interesada fuera del equipo, debe ser capturado y documentado".

A veces las incidencias se registran en el tablero de tareas Ágil, ya sea como una pegatina en una tarea o como una tarea aparte, visualizando que está bloqueando una historia y sus tareas para ser completadas es aceptable usar notas “post-it” en una tabla de tareas para manejar defectos y esto es a menudo el método elegido para los equipos co-localizados. El equipo Ágil trabajará posteriormente sobre las incidencias y las gestionará hasta el cierre (véase el objetivo específico 4 de esta área de proceso en lo sucesivo). Algunos optan por registrar y documentar las incidencias en una herramienta, por ejemplo, la de gestión de defectos o la herramienta de seguimiento de defectos. Dicha herramienta debe utilizarse tan sólo cuando sea posible y no obligar a presentar ningún elemento de los informes de incidencias de pruebas que tengan un valor añadido muy bajo.

Se considera una buena práctica también en los equipos Ágiles registrar datos durante la ejecución de las pruebas para determinar si los ítems probados cumplen con sus criterios de aceptación definidos y de hecho pueden ser etiquetados como “hechos”. Esto también es cierto cuando se aplican técnicas basadas en la experiencia como las pruebas exploratorias. Ejemplos de información que puede ser útil para documentar son la cobertura de las pruebas (cuánto se ha cubierto y cuánto queda por probar), las observaciones durante las pruebas, por ejemplo, el sistema y la historia del usuario bajo pruebas parecen ser estables, cuáles han sido cubiertos y cuáles siguen siendo los más importantes), defectos encontrados y otros problemas y posibles preguntas abiertas.

La información registrada debe ser capturada y/o resumida en algún tipo de herramienta de gestión de estado (por ejemplo, herramientas de gestión de pruebas, herramientas de gestión de tareas, tablero de tareas), de manera que

sea fácil para el equipo y las partes interesadas comprender el estado actual de todas las pruebas que se realizaron. Por supuesto, no es sólo el estado de las tareas de pruebas de forma aislada de las que el equipo necesita un estado, sino que necesita conocer el estado general de la historia del usuario. Nada es completo desde una perspectiva de la historia a menos que todas las actividades relacionadas estén completas, no sólo las pruebas.

2.4.4 SG 4 Gestionar las Incidencias de Pruebas hasta su Cierre

Como se ha indicado anteriormente, en entornos ágiles, a menudo no todas las incidencias encontradas serán registradas. Este objetivo específico sólo se aplica a las incidencias que se registran y por lo tanto necesitan gestionarse hasta el cierre. En principio, administrar incidencias en Ágil es simple. En caso de que la incidencia se registre en el tablero de tareas Ágil, ya sea como una pegatina en una tarea o como una tarea independiente, se visualiza como un defecto que bloquea una historia y sus tareas para ser completadas. El modo de trabajo ágil estándar se aplicará como con cualquier otro impedimento o tarea que esté bloqueando el progreso. Se discutirá durante las reuniones y se asignará para su resolución por el equipo, por lo que desaparecerá de la mesa de tareas.

En caso de que se tome la decisión de aplazar la resolución de la incidencia encontrada para otra iteración, se convierte en simplemente otra opción deseada (o cambio) para el producto. Añádalos al “backlog” y establezca prioridades en consecuencia. Cuando la prioridad es establecida como lo suficientemente alta, será recogida por el equipo en la próxima iteración. Hay que tener en cuenta que la prioridad suele ser definida por el negocio, pero en ocasiones se necesita dar prioridad a las incidencias relacionadas con la llamada deuda técnica. Algunos equipos tienen iteraciones dedicadas a limpiar las incidencias/defectos relacionados con la deuda técnica. Sin embargo, esto no es una práctica recomendada en ágil estándar pues la deuda debe ser pagada sobre una base regular.

Como ejemplo de cómo las incidencias pueden ser gestionadas en un entorno Ágil, primero imagine que el equipo está en medio de una iteración. Al trabajar en una de las historias de usuarios, el equipo determina que no se puede “hacer”, porque uno de los criterios de aceptación no se cumple, por ejemplo, debido a un defecto introducido durante el desarrollo. Los equipos ágiles tienden a arreglar este tipo de defectos tan pronto como sea posible en la iteración actual, de lo contrario realizarían un trabajo no hecho (equivalente a un inventario ligero). Este tipo de incidencias se visualizarán y se administrarán a través del tablero de tareas.

En segundo lugar, supongamos que todas las historias de la iteración se completan de acuerdo con la Definición de Hecho al final de la iteración. Si durante la reunión de revisión, sin embargo, mientras se hace la demo de una historia, hay algo que no se comporta de la manera que el cliente lo desea. Considere que el comportamiento inesperado no se discutió realmente junto con el negocio antes de que comenzara la iteración. En este caso, la funcionalidad de software que se produce está funcionando y lo que probablemente se necesita hacer es traducir la incidencia en una historia adicional y añadirlo al backlog.

En un entorno de ciclo de vida secuencial tradicional, a menudo hay una Junta de Control de Configuración (CCB) específica que cumple, revisa y decide la disposición de la incidencia. Este papel en Ágil es asumido por el equipo potenciado. Por supuesto, el representante comercial, por ejemplo, el propietario del producto tiene un papel importante que desempeñar cuando se trata de decidir sobre la disposición de una incidencia. Manejar defectos es una sobrecarga, que está quitando el enfoque y el esfuerzo de otras cosas. La sobrecarga aumenta cuanto más tratamos de formalizar un proceso para enumerarlos, calificarlos, ordenarlos y, en última instancia, planificarlos. Un proceso de gestión de incidencias complejo como éste es una simple pérdida de tiempo. Es mejor esbozar algunos principios básicos sobre cómo se gestionan y administran las incidencias por los equipos Ágiles y actuar en consecuencia.

2.5 Área de Proceso 2.5 Entorno de Pruebas

El propósito del entorno de pruebas es establecer y mantener un entorno adecuado, incluyendo datos de pruebas, en el que es posible ejecutar las pruebas de forma manejable y repetible.

Con el área de proceso Entorno de pruebas se establece y mantiene un entorno de pruebas adecuado, incluidos los datos de prueba genéricos, con los que es posible ejecutar las pruebas de forma manejable y repetible. Por supuesto también en un proyecto de desarrollo de software que utiliza el ciclo de vida Ágil, un entorno de pruebas adecuado es indispensable. Debido a los tiempos de ciclo cortos de una iteración, el entorno de pruebas necesita ser altamente estable y disponible. Los problemas en el entorno de pruebas siempre tendrán un impacto inmediato en el progreso

y los resultados de la iteración. La gestión adecuada de la configuración y los cambios en el entorno de pruebas y en los datos de pruebas también es de suma importancia.

Los ambientes de pruebas dependen, entre otras cosas, del sistema bajo pruebas, pero normalmente la especificación, implementación y administración del entorno de pruebas se realiza en paralelo al proyecto de desarrollo de software Ágil. La implementación de un entorno de pruebas a menudo toma mucho tiempo y puede ser muy compleja, lo que hace casi imposible realizar esta tarea dentro del tiempo de ciclo limitado de una iteración [Van der Aalst y Davis]. También involucra a diferentes partes interesadas e ingenieros que normalmente forman un equipo Ágil. Las metodologías Ágiles han sido desarrolladas para apoyar el desarrollo de software, no para el desarrollo de la infraestructura o el desarrollo de las pruebas. Es menos adecuado y la forma en que se lleva a cabo el entorno de pruebas o el desarrollo y la gestión de infraestructura sigue siendo en gran medida la misma que en un entorno tradicional, aunque puede haber algunos cambios en el proceso cuando se está practicando Ágil.

2.5.1 Desarrollar los Requerimientos de Entorno

La especificación de los requisitos del entorno de pruebas se realiza tempranamente en el proyecto. La especificación de requisitos se revisa para asegurar su correctitud, idoneidad, factibilidad y representación precisa de un entorno operativo de "vida real". La especificación temprana de requisitos tiene la ventaja de proporcionar más tiempo para adquirir y/o desarrollar el entorno de pruebas requerido y componentes tales como simuladores, stubs o controladores.

En algunos proyectos de desarrollo Ágil se aplica una iteración inicial (iteración 0) cuando se realiza la elicitación y especificación de los requisitos del entorno de pruebas

Se require:



2.5.2 SG2 Implementación del Entorno de Pruebas

A veces (parte de) la implementación se lleva a cabo en la iteración inicial, además de otras actividades, por ejemplo, capacitación, evaluación de riesgos altos del producto y establecer la definición de "hecho". De esta manera se pretende lograr que incluso las primeras iteraciones de desarrollo de software ya tengan un entorno de pruebas (en parte) que funcione. Las acciones o cuestiones técnicas del entorno pueden ser parte del backlog.

Definir el entorno de pruebas (incluidos los datos de pruebas genéricos) podría como con un inicio de ciclo de vida secuencial comenzar por medio de un plan completo, pero a menudo es mejor comenzar con la implementación lo antes posible y tener una versión inicial del entorno de pruebas disponible al inicio de la primera iteración.

De hecho, es difícil ser genérico en ambientes de pruebas. En algunos dominios, los "stubs" y los controladores están siendo rápidamente reemplazados por la virtualización del servicio, permitiendo normalmente una implementación mucho más rápida del entorno de pruebas. En algunos proyectos Ágiles, el aprovisionamiento de entornos de pruebas es altamente automatizado. Los ambientes de pruebas se pueden preparar en minutos u horas, en lugar de en días. También hay mucho más uso hoy en día de la virtualización de máquinas, servidores y servicios. También la nube puede ayudar a proporcionar los entornos requeridos, acelerando de nuevo el acceso.

2.5.3 SG3 Gestionar y Controlar el Entorno de Pruebas

La gestión y el control del entorno de pruebas se llevarán a cabo a lo largo de un proyecto Ágil. A veces, el equipo Ágil necesita configurar y mantener el entorno de pruebas ellos mismos, pero la mayoría de las veces esto lo hace una unidad separada fuera del equipo Ágil. Cuando las actividades son realizadas por el equipo Ágil, esto implica, por supuesto, que el equipo tiene suficientes conocimientos técnicos para poder realizar estas tareas. También significa que las tareas se convierten en parte del proceso ágil, por ejemplo, necesitan ser abordadas en una sesión de planificación, puestas en el tablero de tareas y posiblemente discutidas durante las reuniones diarias de "stand-up" en caso de cualquier problema.

Como conclusión principal para el área de proceso Entorno de Pruebas, las metas y prácticas específicas siguen siendo aplicables y no cambian en esencia. Lo que cambia es su tiempo en el ciclo de vida.

3 TMMi Nivel 3 Definido

3.1 Área de Proceso 3.1 Organización de prueba

El propósito del área de proceso de Organización de pruebas es identificar y organizar un grupo de personas altamente calificadas que se encarguen de las pruebas. Además de las pruebas, el grupo de pruebas también gestiona mejoras en el proceso de prueba de la organización y los activos del proceso de prueba en base a una comprensión exhaustiva de las fortalezas y debilidades del proceso de prueba actual y los activos de este proceso.

3.1.1 SG 1 Establecer una Organización de Prueba

La organización de prueba es un área de proceso a menudo mal entendida. Muchos leen esto como que TMMi requiere un grupo de prueba independiente o incluso un departamento que realice pruebas independientes. Por mucho que sea una posibilidad, también hay otros modelos organizativos que cumplen con los requisitos de TMMi. Además de la organización de prueba independiente que prepara y ejecuta las pruebas, el centro de competencia de pruebas también es una opción. Las tareas y la responsabilidad típicas de un denominado centro de competencia de prueba son establecer, administrar y mejorar los procesos de prueba, y proporcionar recursos a los proyectos con experiencia, conocimiento y habilidades de prueba. Un centro de competencia de prueba comúnmente tiene la propiedad de las metodologías y procesos de prueba. Tenga en cuenta que un probador en el día a día suele ser parte del equipo ágil, pero además de eso es común que un probador pertenezca a una organización de prueba, como el centro de competencia de prueba o el equipo de prueba transversal.

Una organización de prueba requiere liderazgo en áreas relacionadas con pruebas y problemas de calidad. Los miembros del personal de dicho grupo se llaman especialistas en pruebas. La opción del centro de competencia de pruebas se adapta bien a ágil y garantiza que las pruebas se conserven como disciplina y se traten con seriedad. Por supuesto, hay líderes de pensamiento ágiles que no están a favor de mantener disciplinas especiales. Si bien esta puede ser una opción preferida para algunos, implementar TMMi requiere tratar y mantener las pruebas como disciplina. No es necesario decir que esto solo debe hacerse si tiene sentido desde una perspectiva comercial y, por lo tanto, cuando esto no tiene sentido, TMMi probablemente no sea la ruta de mejora preferida.

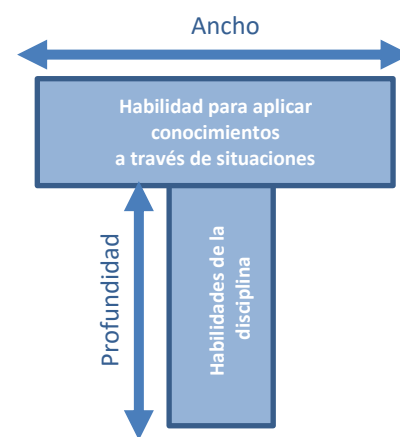
En ágil, la organización de prueba puede adoptar el formato de un equipo de prueba transversal. Un equipo de prueba transversal suele ser un grupo informal de autoevaluación de probadores, que son miembros de diferentes equipos ágiles. Las actividades importantes de un equipo de prueba transversal de prueba pueden ser el intercambio de conocimientos, el aprendizaje inteligente en grupo, la observación de tendencias, etc. La membresía de un equipo de prueba transversal es voluntaria. En la práctica, un equipo de prueba transversal a menudo se reúne cada dos semanas. A un equipo de prueba transversal también se le pueden asignar tareas más formales como establecer trayectorias profesionales de prueba, organizar la capacitación y determinar, planificar e implementar mejoras en el proceso de prueba. Es importante asegurarse de que el equipo de prueba transversal permanezca en gran medida informal y autogestionado. Una restricción importante para que un equipo de prueba transversal tenga éxito es que a sus miembros se les asigna tiempo para dedicarlo a las diversas actividades de equipo de prueba transversal que se realizarán.

Los probadores independientes suelen ser más efectivos para encontrar defectos. Equilibrar la independencia con ágil es un desafío. Algunos equipos ágiles retienen equipos de prueba completamente independientes y separados, y asignan probadores a pedido durante los últimos días de cada sprint. Esto puede preservar la independencia, y estos probadores pueden proporcionar una evaluación objetiva e imparcial del software. Sin embargo, las presiones de tiempo, la falta de comprensión de las nuevas características del producto y los problemas de relación con las partes interesadas y los desarrolladores comerciales a menudo conducen a problemas con este enfoque. Otra opción es tener una organización de prueba independiente y separada donde los probadores sean asignados a equipos ágiles a largo plazo, al comienzo del proyecto, lo que les permitirá mantener su independencia mientras obtienen una buena comprensión del producto y relaciones sólidas con otros miembros del equipo. Además, la organización de prueba independiente puede tener probadores especializados fuera de los equipos ágiles para trabajar en actividades a largo plazo y/o independientes de la iteración, como desarrollar herramientas de prueba automatizadas, realizar pruebas no funcionales, crear y soportar entornos de prueba y datos y llevar a cabo niveles

de prueba que podrían no encajar bien dentro de un sprint (por ejemplo, pruebas de integración del sistema). Tenga en cuenta que también hay posibilidades de practicar pruebas independientes dentro de un equipo ágil. Mientras alguien que no sea el creador de los activos o el código también pruebe el producto, esto sigue siendo independencia. Por lo tanto, los desarrolladores pueden probar el código de los demás a través de revisiones de código y/o pruebas unitarias, o analistas de negocios que contribuyen a las pruebas desde la perspectiva del usuario utilizando, por ejemplo, diferentes personas. Por último, pero no menos importante, la independencia puede organizarse a través de una estructura organizativa y de responsabilidad, pero probablemente sea sobre todo tener la mentalidad crítica correcta.

3.1.2 SG 2 Establecer funciones de prueba para especialistas en pruebas

Las pruebas se consideran una profesión y disciplina valoradas. El equipo ágil necesita conocimientos y habilidades de prueba. El especialista en pruebas proporcionará estos, y también entrenará a otros miembros del equipo mientras realiza actividades de prueba. Las funciones de prueba y las trayectorias profesionales de prueba están definidas y respaldadas por un programa de capacitación de prueba. El conocimiento y las habilidades requeridas para un probador ágil típico son diferentes que en una organización tradicional. Por supuesto, el probador seguirá necesitando conocimientos y habilidades de prueba "tradicionales", pero además también conocimientos y habilidades para la automatización de pruebas y para poder realizar tareas fuera de las pruebas, por ejemplo, secuencias de comandos o ingeniería de requisitos. Dado que un probador ágil está trabajando en un equipo, las habilidades blandas son tan importantes como el crecimiento de las habilidades técnicas.



El probador ágil es un probador llamado en forma de T que se reflejará en la descripción de las funciones de prueba. La organización de prueba está compuesta por personas que tienen las habilidades y la motivación para ser buenos probadores en un contexto ágil. Se asignan a una función de prueba específica. Especialmente en un contexto ágil, es importante, ya que todo el equipo llevará a cabo actividades de prueba, para elaborar otras descripciones de funciones específicas que no sean de prueba con expectativa respecto a las actividades de prueba, los roles y los conocimientos y habilidades de prueba requeridos.

3.1.3 SG 3 Establecer trayectorias profesionales de prueba

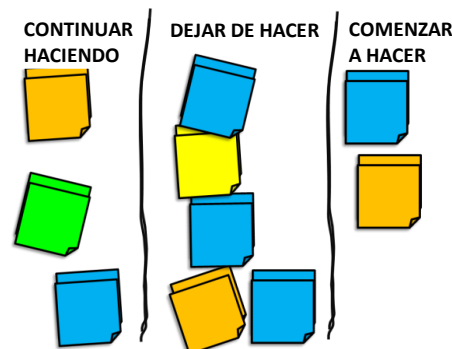
Las trayectorias profesionales de prueba se establecen para permitir que los profesionales de prueba mejoren sus conocimientos, habilidades, estado y recompensas, y así les permitan avanzar en sus carreras. Con base en las trayectorias profesionales de prueba definidas, se desarrollan y mantienen planes de desarrollo profesional de prueba personal para cada miembro de la organización de prueba. Este objetivo y las prácticas específicas que lo acompañan también son totalmente aplicables en un contexto ágil. Con el énfasis ágil en el aspecto de las personas, este es claramente un objetivo específico importante. El equipo ágil necesita probadores que entiendan su trabajo como desafiante, puedan entrenar a otros miembros del equipo y estén motivados. Muchos de estos se abordan con trayectorias profesionales de prueba.

Sin embargo, la definición real de las trayectorias profesionales de prueba generalmente será algo diferente en una organización ágil en comparación con las de una organización que sigue un ciclo de vida secuencial. Mientras que en una organización tradicional las trayectorias profesionales de prueba a menudo se centran en el crecimiento vertical (del probador al gerente de prueba), las trayectorias profesionales de prueba en una organización ágil tienden a centrarse en un crecimiento horizontal (del probador junior al probador senior y quizás al entrenador de prueba). Por lo general, existe una menor necesidad de diversas funciones y roles jerárquicos, como la gestión de pruebas y una mayor necesidad de crecer a lo largo de los roles de propietario del producto o Scrum master. Tenga en cuenta que muchas de las tareas tradicionales de gestión de pruebas siguen siendo relevantes, pero las realizan otros roles (por ejemplo, maestro Scrum master, equipo ágil).

3.1.4 SG 4 Determinar, Planificar e Implementar Mejoras en el Proceso de Pruebas

El proceso de mejora descrito por este objetivo específico de TMMi y sus prácticas específicas están cubiertos en gran medida por la reunión retrospectiva realizada por un equipo ágil. La retrospectiva de iteración es una oportunidad para que el equipo de ágil se inspeccione a sí mismo y defina acciones para mejoras que se implementarán durante la próxima iteración. Una retrospectiva generalmente ocurre después de la revisión de iteración y antes de la siguiente planificación de iteración.

Durante la retrospectiva, el equipo analiza lo que salió bien en la iteración, lo que podría mejorarse y lo que se comprometerá a mejorar en la próxima iteración. Durante cada retrospectiva, el equipo ágil identifica y planifica formas de incrementar la calidad del producto, por ejemplo, mejorando los procesos de trabajo o adaptando la definición de lo hecho, si corresponde. Las retrospectivas pueden dar como resultado decisiones de mejora relacionadas con la prueba centradas en la efectividad de la prueba, la productividad de la prueba y la calidad de las pruebas. También pueden abordar la capacidad de prueba de las aplicaciones, historias de usuarios, características o interfaces del sistema. El análisis de la causa raíz de los defectos puede impulsar las pruebas y las mejoras de desarrollo. Los probadores juegan un papel importante en las retrospectivas. Forman parte del equipo y aportan su perspectiva única. Todos los miembros del equipo, probadores y no probadores pueden proporcionar información sobre las actividades de prueba y no prueba.



Al final de la retrospectiva, el equipo ágil debería haber identificado las mejoras (de prueba) que implementará en la próxima iteración. La implementación de estas mejoras en la próxima iteración es la adaptación a la inspección del propio equipo Ágil. El equipo vota sobre las mejoras más importantes para la próxima iteración, por lo que las mejoras (de prueba) pueden permanecer en el backlog durante más de una iteración, pero eventualmente se implementarán. Es importante que el equipo solo adquiera mejoras que puedan incluir en la próxima iteración, por lo que probablemente sean un máximo de dos o tres por iteración. Los probadores deben tomar posesión de las mejoras de prueba para garantizar que se implementen dentro del equipo. La retrospectiva es un mecanismo importante que permite que un equipo evolucione y mejore continuamente a lo largo de la vida de un proyecto.

Debido a que el alcance a menudo se limita al ciclo o iteración anterior, se realizan mejoras pequeñas pero frecuentes que se centran principalmente en resolver problemas específicos del proyecto. El enfoque de estas mejoras a menudo no está en el aprendizaje entre proyectos y la institucionalización de las mejoras. Al observar cómo se organiza y gestiona la mejora del proceso (prueba), es probable que haya menos enfoque en un grupo de proceso (prueba) a nivel organizacional y más énfasis en la autogestión de los equipos dentro del proyecto. Si bien esto no es algo malo, es importante crear también un mecanismo mediante el cual las mejoras de las pruebas locales que tienen éxito se compartan con el resto de la organización y esas áreas de mejora que están más allá del poder del equipo Ágil también se puedan abordar. Una posible solución podría ser hacer que un representante de la organización de prueba asista a reuniones retrospectivas locales para garantizar que las mejoras se compartan e institucionalicen en toda la organización cuando sea apropiado. El mejorador del proceso de prueba también puede eliminar problemas más amplios o fundamentales que afectan las pruebas en toda la organización.

3.1.5 SG 5 Implemente el proceso de prueba organizacional e incorpore las lecciones aprendidas

También para un proyecto ágil es importante tener acceso y reutilizar el proceso de prueba estándar de la organización y los activos del proceso de prueba tanto como sea necesario y agregue valor. La organización de prueba se asegurará de que se implementen en todos los proyectos. (Consulte el área del proceso Ciclo de vida de prueba e integración para obtener más información sobre el proceso de prueba estándar de la organización y los activos del proceso de prueba y cómo esto se relaciona con los proyectos ágiles).

La implementación del proceso de prueba estándar de la organización y el uso de los activos del proceso de prueba en proyectos ágiles pueden ser monitoreados por los propios equipos autoorganizados, por ejemplo, abordando esto en retrospectivas. Con un representante de la organización de prueba, por ejemplo, un mejorador del proceso de prueba, que está presente en estas reuniones, esta persona puede realizar un seguimiento del estado de

implementación y abordar problemas entre proyectos. Esta persona también puede tomar valiosas lecciones aprendidas y probar las mejoras del equipo ágil a la organización de prueba. Las lecciones aprendidas y la mejora de la prueba pueden presentarse posteriormente como propuestas de mejora del proceso de prueba y administrarse en consecuencia. Por lo tanto, las lecciones aprendidas de las pruebas con el equipo Ágil se incorporan, si corresponde, en el proceso del estándar organizacional (Ágil) (prueba) y los activos del proceso de prueba.

3.2 Área de Proceso 3.2 Programa de Entrenamiento en Pruebas

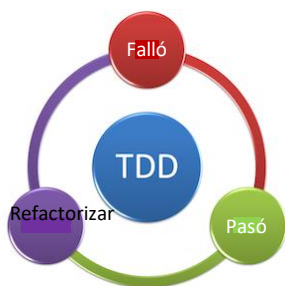
El propósito del área de proceso del Programa de Entrenamiento de Pruebas es desarrollar un programa de capacitación que facilite el desarrollo del conocimiento y las habilidades de las personas para que las tareas y roles de prueba se puedan realizar de manera efectiva y eficiente.

El objetivo principal del Programa de Entrenamiento en Pruebas es proporcionar la capacitación necesaria (en pruebas) a los probadores y otros miembros del equipo. Como resultado de la declaración del manifiesto ágil "individuos e interacciones sobre procesos y herramientas", los procesos en un entorno ágil generalmente se documentarán de forma ligera y no abordarán todos los escenarios posibles. Sin embargo, la capacitación de las personas se convierte en un factor crítico de éxito que necesita atención y es casi como una contramedida para los procesos livianos. Un programa de capacitación de calidad asegura que quienes participan en las pruebas continúen mejorando sus conocimientos y habilidades de prueba, y adquieran el conocimiento de dominio necesario y otros conocimientos y habilidades relacionados con la prueba. Dado que la calidad y las pruebas ahora son responsabilidad del equipo, se espera que la capacitación relacionada con las pruebas no se brinde solo a los profesionales de las pruebas, sino a todo el equipo de Ágil.

3.2.1 SG 1 Establecer una capacidad de entrenamiento en pruebas organizacionales

El proceso comienza con la identificación de las necesidades de capacitación en pruebas estratégicas de la organización. Además de las necesidades tradicionales de capacitación para los probadores, por ejemplo, sobre técnicas de diseño de pruebas y medidas de cobertura, ahora también es necesario adquirir conocimientos y

habilidades sobre los detalles específicos de las pruebas ágiles. Los ejemplos incluyen el manifiesto ágil y sus principios, marcos como Scrum, Kanban y XP, conceptos de prueba ágiles (por ejemplo, Pirámide de prueba y cuadrantes de prueba), desarrollo impulsado por prueba (TDD), desarrollo impulsado por comportamiento (BDD), desarrollo impulsado por prueba de aceptación (ATDD), desarrollo de historias de usuario que incluye el apoyo a la definición de criterios de aceptación y la revisión de su capacidad de prueba, automatización de pruebas y pruebas exploratorias. Además, algunas áreas típicas de conocimiento y habilidades de administración de pruebas, tales como planificación, estimación (por ejemplo, planificación de póker), monitoreo y análisis de riesgos (por ejemplo, riesgo de póker) ahora se convertirán en áreas que deben formar parte del conjunto de conocimientos y habilidades



para la mayoría de los probadores que trabajan en un equipo ágil y no solo para el gerente de pruebas. La identificación de las necesidades de capacitación en pruebas estratégicas no se limita solo a los probadores, es de suma importancia especificar también las necesidades de capacitación relacionadas con las pruebas para otros miembros de equipos ágiles.

Al igual que en cualquier otra organización, las necesidades de capacitación en pruebas se traducirán en un plan de capacitación (liviano), que posiblemente también aborde algunas necesidades específicas de capacitación en pruebas de proyectos. Es necesario revisar el plan de capacitación y establecer compromisos. Tenga en cuenta que, especialmente en Ágil, algunas habilidades se imparten de manera efectiva y eficiente a través de vehículos informales (por ejemplo, capacitación en el trabajo, sesiones de capacitación en el almuerzo, entrenamiento y tutoría), mientras que otras habilidades requieren capacitación formal. El enfoque apropiado para satisfacer las necesidades específicas de capacitación en pruebas debe determinarse por área de conocimiento y habilidades, por ejemplo, siendo la capacitación en pruebas exploratorias en el trabajo un aspecto importante.

3.2.2 SG 2 Proveer la Capacitación necesaria para las Pruebas

En términos generales, las prácticas específicas para SG2 son las mismas en una organización tradicional y en una ágil. Según el plan de capacitación organizacional, se identifican los miembros del equipo que necesitan recibir la capacitación para poder desempeñar su función de prueba de manera efectiva. La capacitación se programa posteriormente, incluidos los recursos necesarios, y se lleva a cabo. Esto parece una declaración obvia, pero algunas organizaciones ágiles luchan por encontrar tiempo para la capacitación. Los equipos ágiles siempre están completamente comprometidos e intentan cumplir con el plazo de finalización de la iteración. Por supuesto, el tiempo para la capacitación debe abordarse en la planificación de la iteración, por lo que el asistente estará menos disponible para la próxima iteración.

Como se indicó anteriormente, los vehículos de capacitación informal a menudo se usan en una organización ágil. Por ejemplo, además de la capacitación formal, la tutoría y el entrenamiento en el trabajo generalmente se fomentan y emplean continuamente. De hecho, la tutoría es responsabilidad de todos y se espera en todos los niveles de una organización. Trabajar en pares es otro vehículo comúnmente utilizado por los equipos de Ágil para mejorar la habilidad y la transferencia de conocimiento. Cuando se usa Scrum, es responsabilidad del Scrum master entrenar al equipo en las prácticas de Scrum.

Se crearán registros de la capacitación de prueba realizada, ya sea informal o formal, y se evaluará la efectividad de la capacitación de prueba. Una capacitación (de prueba) a la que se asiste también se puede evaluar como parte de una reunión retrospectiva en la que se lleva a cabo una discusión sobre si el conocimiento y las habilidades de los asistentes ahora son más adecuados para realizar las tareas de prueba.

3.3 Área de Proceso 3.3 Ciclo de Vida de Prueba e Integración

El propósito del Ciclo de Vida de pruebas e Integración es establecer y mantener un conjunto utilizable de activos de procesos de prueba organizacionales (por ejemplo, un ciclo de vida de prueba estándar) así como estándares de entorno de trabajo e integrar y sincronizar el ciclo de vida de prueba con el de desarrollo. El ciclo de vida integrado asegura la implicación temprana de las pruebas en un proyecto. El propósito del Ciclo de Vida de pruebas e Integración también incluye definir un enfoque de prueba coherente a lo largo de múltiples niveles de prueba, basado en los riesgos identificados y en la estrategia de prueba definida, y proporcionar un plan de prueba general, basado en el ciclo de vida de pruebas definido.

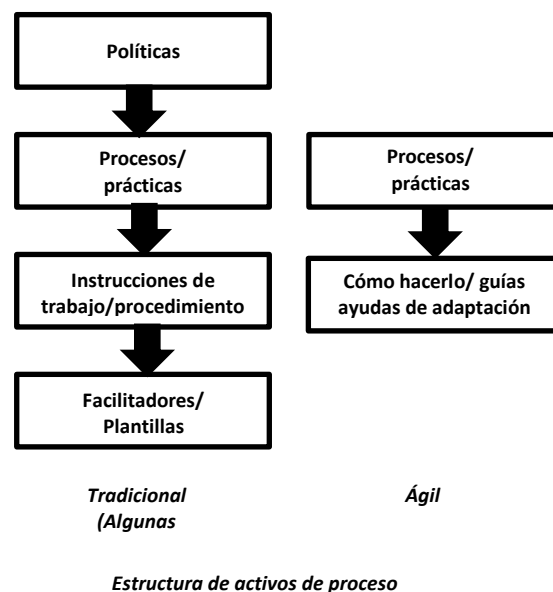
3.3.1 SG 1 Establecer los activos del proceso de pruebas organizacional

Una responsabilidad importante de la organización de pruebas es definir, documentar y mantener un proceso de pruebas estándar, alineado con la política y los objetivos de pruebas de la organización. Esto incluye una descripción de los distintos tipos y niveles de prueba a ejecutar. Si bien TMMi no exige procesos de pruebas definidos y documentados de forma detallada a menudo, plantillas bien enfocadas, elaboradas con ejemplos, son muy útiles para establecer una forma común de trabajo. Los tres pilares de calidad y pruebas ágiles de Bob Galen, son una buena referencia donde se identifican estándares como listas de comprobación, plantillas y repositorios como uno de los tres pilares de las pruebas de software ágiles [Galen]. Una plantilla identifica las actividades requeridas para recopilar información y documentar los resultados implícitos dentro de su estructura. Un proceso no necesita ser una secuencia estricta de actividades. Cuando existen dependencias, pueden capturarse mediante notas en la plantilla o campos en un formulario que no están disponibles hasta que se han completado otros campos previos. Las plantillas tienen un valor práctico que transmite la intención real de un proceso. Las plantillas también evitan las ambigüedades que se encuentran comúnmente en los procesos documentados de forma muy detallada. Como organización ágil, si tiene procesos efectivos, es posible que solo necesite documentarlos de una manera ligera y tal vez agregar algunas cosas. Algunas pautas clave que a menudo son utilizadas en una organización ágil a la hora de establecer procesos, son:

- Los aspectos obligatorios del proceso son agrupados fuera de las guías
- Ningún proceso tiene más de dos páginas (objetivo, regla liviana)

- Los procesos no contienen información de "cómo hacerlo" o información de herramientas a menos que uno haya decidido ordenar esto en todos los proyectos, independientemente de su tamaño y escala.
- Las guías contienen opciones de adaptación / planificación e información de "cómo hacerlo". Tenga en cuenta que para la información de "cómo hacerlo" se puede hacer referencia a un libro o documento existente.

No es infrecuente en muchas organizaciones grandes, que existan cuatro niveles de activos del proceso, como políticas, procesos/prácticas, instrucciones/procedimientos de trabajo y ayudas/plantillas. Esto no es debido a ningún requisito de TMMi, sino por la forma elegida a la hora de implementar sus activos de proceso. Si bien la elección de los activos del proceso depende de cada organización, la mayoría de las organizaciones ágiles descubrieron que solo son suficientes dos niveles de activos del proceso. Esto se logra mediante la consolidación de una declaración de la política con la descripción del proceso asociado que encapsula "lo que se debe hacer" para llevar a cabo el proceso. El segundo nivel contiene pautas de "cómo hacerlo" para ejecutar el proceso y adaptarlo. Este nivel puede verse como una ayuda para adaptar el proceso, otorgando autonomía al equipo para decidir cómo trabajar exactamente dentro del marco definido y generalmente incluye plantillas de soporte. En la mayoría de las organizaciones ágiles, los procedimientos paso a paso se reemplazan por guías de herramientas y formación/tutoría. Como se indicó anteriormente,



una plantilla, como por ejemplo una plantilla de pruebas, una plantilla de hoja de sesión de prueba o una plantilla de plan de pruebas, puede servir como guía de las actividades de proceso requeridas que están implicadas en la plantilla. Esta es una técnica habitual a la hora de desarrollar procesos ágiles efectivos [McMahon].

El conjunto de procesos de pruebas estándar de la organización puede adaptarse a los proyectos para crear sus procesos específicos definidos. En el contexto de PE 1.3 Establecer criterios y pautas de personalización, la mayoría de las organizaciones actuales utilizan el denominado enfoque de adaptación a medida. Este enfoque requiere que las personas se esfuercen por explicar por qué no se necesita algo. Esto supone gran cantidad de esfuerzo en los proyectos más simples, lo cual no es un enfoque ágil. Cuando haces la adaptación, ¿hasta dónde puedes llegar? ¿Hay un mínimo definido? Sin embargo, si se comienza en un conjunto mínimo donde el conjunto mínimo es lo que todos deben hacer, se elimina el riesgo de adaptar un "debe hacerse". El enfoque de adaptación es mucho más coherente con los modelos ágiles y cumple totalmente con TMMi. La adaptación es un mecanismo extremadamente poderoso para las organizaciones ágiles que desean mantener su agilidad o para organizaciones altamente maduras (de prueba) que buscan aumentar su agilidad. Un conjunto de criterios simples podría ayudar a hacer las preguntas correctas durante la adaptación. Sin criterios de ayuda se tiende a adaptar en exceso.

Para respaldar la implementación del proceso de prueba estándar, se creará y mantendrá una biblioteca de activos del proceso de pruebas, a menudo en una wiki, en la que los probadores pueden encontrar, por ejemplo, plantillas, mejores prácticas y listas de verificación, que los ayudarán en su trabajo diario. La creación de esta biblioteca de activos de proceso de pruebas también tiene sentido en un contexto ágil, siempre y cuando los equipos obtengan un valor agregado al usar los activos proporcionados durante sus actividades de prueba. De forma similar, este razonamiento también aplica a la base de datos de procesos de pruebas en la que los datos de pruebas, por ejemplo, estimaciones, cobertura, medidas de calidad, se recopilan y se ponen a disposición de los equipos. Es importante asegurarse de que todo lo que se recopila y se pone a disposición para permitir el aprendizaje entre equipos y el intercambio de experiencias tiene un valor agregado para los equipos.

Por supuesto, también en un contexto ágil, los estándares del entorno de trabajo se utilizan para guiar la creación de entornos de trabajo de proyectos.

3.3.2 SG 2 Integrar el ciclo de vida de los modelos de pruebas con los modelos de desarrollo

El modelo de ciclo de vida de prueba estándar define las actividades principales y los entregables para los distintos niveles de prueba. El modelo de ciclo de vida de prueba estándar está alineado con los modelos de ciclo de vida de desarrollo para integrar las actividades de prueba en términos de fases, hitos, entregables y actividades. La integración del ciclo de vida se realiza de tal manera que se garantiza la participación temprana de las pruebas en los proyectos. Todo esto parece que no hay nada que hacer aquí en caso de que se haya adoptado un ciclo de vida ágil; Ya está todo resuelto. En Ágil, el desarrollo y las pruebas deberían estar totalmente integrados dentro del ciclo de vida, y las pruebas deberían realizarse lo antes posible.

Por mucho que esto sea cierto en muchas organizaciones que usan un modelo Ágil, también hay organizaciones que luchan con este aspecto. Su iteración se asemeja a un mini ciclo de vida del modelo V para el desarrollo donde las pruebas no comienzan hasta el final de la iteración, a menudo se realizan bajo presión de tiempo y, a veces, incluso se interrumpen. Por supuesto, esto no es una implementación correcta de una integración de desarrollo ágil con el ciclo de vida de pruebas. Sin embargo, muestra que para algunas organizaciones que son "ágiles", este objetivo específico es un área de mejora. Es importante que se cree una comprensión común al definir cómo las pruebas pueden involucrarse y realizarse al principio de la iteración.

Este objetivo específico también es aplicable a una organización que realiza desarrollo de software ágil. Si se realiza de manera adecuada, el desarrollo y las pruebas deben integrarse completamente y existe un entendimiento común sobre cómo se hace esto. En tales casos, no es necesario que suceda nada adicional. En otras circunstancias, como se describió anteriormente, este objetivo específico es un área de mejora muy importante y debería conducir a la realización de actividades de desarrollo y pruebas mucho mejor alineadas dentro en una iteración.

3.3.3 SG 3 Establecer un Plan Maestro de Prueba

En el nivel 3 de TMMi, las pruebas se planifican en un plan maestro que aborda la coordinación de las tareas de prueba, las responsabilidades y el enfoque de prueba en todos los niveles de prueba. Esto evita redundancias innecesarias u omisiones de pruebas entre los distintos niveles de prueba y puede aumentar significativamente la eficiencia y la calidad del proceso de prueba general. El plan maestro de pruebas describe cómo la estrategia de pruebas se aplica en un proyecto en particular, incluidos los niveles concretos que se llevarán a cabo y la relación entre esos niveles. Una planificación (de pruebas) de nivel 2 de TMMi ya se aborda en la versión y en el nivel de iteración como parte del área de proceso de Planificación de prueba. Un equipo ágil generalmente realiza múltiples niveles de prueba que deben abordarse adecuadamente como parte de la planificación de la versión e iteración.

Sin embargo, algunos proyectos que funcionan de acuerdo con un modelo de ciclo de vida ágil han adoptado un enfoque algo híbrido, en el que, además de las pruebas dentro de un equipo ágil, algunas pruebas se organizan fuera del equipo. En tales circunstancias, un plan de prueba maestro definirá y administrará la relación de los equipos ágiles con aquellos niveles de prueba que se realizan fuera de su alcance, por ejemplo, pruebas no funcionales específicas (seguridad, rendimiento), pruebas de integración de hardware/software, pruebas de integración de sistemas o prueba beta. Todas estas prácticas específicas relacionadas con este objetivo específico también serán aplicables. Especialmente en proyectos más grandes y a medida que las organizaciones crecen, documentar un plan general de prueba global, así como las decisiones relacionadas, de manera consistente comienza a tener un mayor valor. Esto es especialmente cierto cuando una organización está utilizando marcos ágiles escalados.

En caso de que todas las pruebas se realicen dentro de los equipos Ágiles, un plan de prueba maestro específico no tiene valor agregado y, por lo tanto, es probable, que este caso este objetivo de TMMi no sea relevante. Aquí se espera que la planificación a nivel de versión e iteración cubra todos los aspectos de prueba necesarios.

3.4 Área de proceso 3.4 Pruebas no funcionales

El propósito del área del proceso de Pruebas no funcionales es mejorar las capacidades del proceso de pruebas para incluir pruebas no funcionales durante la planificación, el diseño y la ejecución de la prueba.

Aunque las pruebas no funcionales son importantes, es independiente del ciclo de vida que se aplica. El producto que se está desarrollando determinará qué aspectos no funcionales son relevantes para las pruebas. Por supuesto, esto es cierto tanto para los modelos de ciclo de vida secuencial como para los modelos de desarrollo ágil. Por lo tanto, las pruebas no funcionales también son aplicables al desarrollo de software ágil.



Sin embargo, dependiendo de la naturaleza de los aspectos no funcionales y del enfoque de pruebas utilizado, algunas pruebas no funcionales, por ejemplo, rendimiento y fiabilidad, cuando se prueban de manera tradicional, dichas pruebas no pueden probarse en una iteración corta. Como estas pruebas no funcionales pueden, dependiendo de los métodos y técnicas que se utilicen, durar semanas para configurar, organizar y ejecutar, su realización es menos aplicable al ritmo de las iteraciones. Por lo tanto, a menudo se necesita un enfoque diferente para probar atributos no funcionales. Este es uno de los cambios fundamentales para mejorar con modelos ágiles, ya que los atributos de calidad ya se prueban en fases tempranas a lo largo de las iteraciones, y no se dejan hasta el final como con un modelo tradicional de ciclo de vida.

No todos los componentes/funciones pueden estar disponibles desde la iteración uno para hacer una prueba no funcional completa, sin embargo, los resultados de esta prueba pueden usarse para dar indicaciones tempranas de problemas de calidad.

Las historias de los usuarios deben abordar elementos tanto funcionales como no funcionales para asegurar que se desarrolle el producto adecuado para usuarios y clientes. Las características de calidad ISO 25010 pueden ayudar a estructurar los requisitos y los probadores deben considerar estos elementos no funcionales [ISO25010].

3.4.1 SG 1 Realizar una evaluación de riesgo de producto no funcional

La(s) sesión(es) de riesgo del producto para proyectos Ágiles, tal y como se identifica y describe en el objetivo específico SG 1 Realizar Evaluación de Riesgos como parte del área de proceso de Planificación de Pruebas, ahora se ampliará explícitamente para incluir también aspectos y riesgos no funcionales. A nivel de versión, la evaluación de riesgos, ahora también para pruebas no funcionales, se puede realizar en función de la visión del producto. A nivel de iteración, esto se realiza utilizando las historias de usuario que definen los requisitos no funcionales como entrada principal.

Preferentemente, todos los miembros del equipo, incluido el propietario del producto, y posiblemente otras partes interesadas deberían participar en las sesiones de riesgo del producto. Para algunas áreas no funcionales, se pueden necesitar especialistas para colaborar. Las sesiones de riesgo del producto generalmente darán como resultado una lista documentada de riesgos de producto prioritarios (no) funcionales. Como se indicó en el área del proceso de Planificación de pruebas, en los proyectos ágiles, tanto el proceso utilizado como la documentación resultante serán mucho más livianos en comparación con un proyecto tradicional que sigue un modelo de ciclo de vida secuencial.

3.4.2 SG 2 Establecer un enfoque de prueba no funcional

Se define un enfoque de prueba para las características de calidad no funcionales más relevantes para mitigar los riesgos de productos no funcionales identificados y priorizados. Para una iteración específica, las características no funcionales que se probarán se identifican durante la planificación de la iteración. La lista priorizada de características no funcionales que se probarán generalmente se relaciona con las historias de usuarios que se probarán en esta iteración. Los nuevos riesgos de productos no funcionales pueden hacerse evidentes durante la iteración que requiere pruebas adicionales. Los problemas como los riesgos de nuevos productos no funcionales que requieren pruebas adicionales generalmente se tratan en las reuniones diarias.

El enfoque de prueba no funcional que se define a nivel de iteración para mitigar los riesgos no funcionales generalmente cubrirá la identificación de los métodos de prueba no funcionales y las técnicas de prueba adecuadas en función del nivel y el tipo de riesgos no funcionales. Por lo general, también abordará el uso de herramientas de soporte y el enfoque para la automatización de pruebas para pruebas no funcionales. El enfoque de prueba para pruebas no funcionales a nivel de versión estará en un nivel mucho más alto y debería basarse en la estrategia de prueba definida a nivel de programa u organización, y también el plan maestro de prueba definido, si existe. (Consulte SG 3 Establecer un plan maestro de prueba dentro del área de proceso Ciclo de vida e integración de la prueba para conocer las razones para establecer o no establecer un plan maestro de prueba). Tanto el enfoque de pruebas no funcional en el nivel de versión como el de iteración son parte del enfoque de prueba general y se mantendrán o mostrarán en la wiki del equipo/proyecto.

Los criterios de salida no funcionales son parte de la denominada definición de hecho (DoD). Es importante que el DoD tenga criterios específicos relacionados con las pruebas no funcionales, por ejemplo, el tiempo medio entre fallas (MTBF) o "las páginas web de "front-end" han sido probadas para la lista de los 10 principales riesgos de OWASP". La iteración debe dar como resultado la implementación del conjunto acordado de historias de usuarios no funcionales (incluidos sus criterios de aceptación) y cumplir con los criterios de salida no funcionales (de prueba) tal como se definen en el DoD. Tenga en cuenta que los atributos no funcionales también pueden ser parte de los criterios de aceptación para historias de usuario funcionales y necesariamente deben especificarse como historias de usuario no funcionales separadas. No solo existirá una definición de hecho a nivel de iteración, a menudo también hay un DoD a nivel de versión que abarca múltiples iteraciones. También el DoD a nivel de liberación puede tener criterios relacionados no funcionales.

3.4.3 SG 3 Realizar análisis y diseño de pruebas no funcionales

Este objetivo específico sigue en gran medida las mismas prácticas, pero ahora desde una perspectiva no funcional, como ocurre con el objetivo específico SG1 Realizar análisis y diseño de prueba utilizando técnicas de diseño de prueba del área de proceso Diseño y ejecución de prueba. Durante el análisis y el diseño de las pruebas, el enfoque de pruebas para las pruebas no funcionales se traduce en condiciones y pruebas tangibles. En modelos ágiles, el análisis y diseño de pruebas, y la ejecución de pruebas son actividades de apoyo mutuo que generalmente se ejecutan en paralelo a lo largo de una iteración. Esto también es cierto para la mayoría de las pruebas no funcionales. El análisis de pruebas no funcionales no es, por lo tanto, una actividad explícitamente separada, sino más bien una actividad implícita que los probadores realizan como parte de su rol dentro del desarrollo colaborativo de historias de usuarios.

Basado en el análisis de las historias de usuarios no funcionales, se identifican las condiciones de prueba no funcionales. Las condiciones de prueba son básicamente una identificación de "cosas" que necesitan ser probadas/cubiertas. Siempre que los criterios de aceptación definidos sean detallados y lo suficientemente claros, generalmente asumirán el papel de las condiciones de prueba tradicionales. Los criterios de aceptación se traducen posteriormente en pruebas no funcionales. Con las pruebas no funcionales, a menudo es beneficioso realizar análisis de pruebas a un nivel superior en lugar de solo historias de usuarios. Por ejemplo, analizando una característica, una épica o una colección de historias para identificar condiciones de pruebas no funcionales que son más abstractos que los identificados a nivel de historia del usuario. Con el principio de 'prueba-primero' aplicado con modelos ágiles, las pruebas no funcionales que cubren el conjunto de condiciones de prueba no funcionales se identificarán (y posiblemente se automatizarán) antes, o al menos en paralelo, con el desarrollo del código.

Para la mayoría de las pruebas no funcionales manuales, las pruebas se identificarán/refinarán a medida que el equipo avance con la ejecución de las pruebas no funcionales. Las pruebas a menudo no se documentan con tanto detalle como en los proyectos tradicionales, sino más bien en un formato de ideas de prueba cuando se usan pruebas exploratorias. Para áreas no funcionales complejas y críticas, es preferible para cubrir riesgos, un enfoque más tradicional durante el diseño de pruebas y el desarrollo de casos de prueba utilizando técnicas formales de diseño de pruebas no funcionales. Sin embargo, también con este enfoque, la cantidad de documentación será limitada en comparación con las pruebas no funcionales en un entorno tradicional de ciclo de vida secuencial. La priorización de las pruebas no funcionales generalmente sigue la priorización de la historia del usuario que están cubriendo. Sin embargo, la priorización también puede estar impulsada por el tiempo necesario para preparar y realizar una determinada prueba no funcional.

Se identifican los datos de prueba específicos necesarios para respaldar la ejecución de pruebas no funcionales. En modelos ágiles, los datos de prueba necesarios generalmente no se especifican en primer lugar en un documento de especificación de prueba. Cuando se especifica, a menudo se registra como parte de la historia del usuario. Sin embargo, siempre que las herramientas y/o funcionalidades necesarias estén disponibles, los datos de prueba necesarios para soportar las pruebas no funcionales normalmente se crean al instante para permitir un inicio rápido de la ejecución de pruebas manuales no funcionales. Por el contrario, para las pruebas no funcionales automatizadas, los datos generalmente deben especificarse por adelantado.

La trazabilidad entre los requisitos no funcionales, las condiciones de prueba y las pruebas deben establecerse y mantenerse. Los equipos deben dejar claro que han cubierto las diversas historias de usuarios no funcionales y los criterios de aceptación como parte de sus pruebas. En muchas organizaciones ágiles, las historias de los usuarios son la base para desarrollar un conjunto de criterios de aceptación relacionados y pruebas posteriores. Con este enfoque se puede lograr la trazabilidad horizontal desde los requisitos hasta la prueba.

3.4.4 SG 4 Realizar la implementación de prueba no funcional

La implementación de la prueba consiste en preparar todo para iniciar la ejecución de las pruebas. Por lo general, se minimiza el desarrollo de documentación de pruebas, por ejemplo, procedimientos de pruebas, para soportar la ejecución de las pruebas. Más bien, los scripts de pruebas automatizados (regresión) se desarrollan y priorizan.

La implementación de la prueba no funcional seguirá muchas de las prácticas ya descritas en el objetivo específico SG 2 Realizar la implementación de la prueba como parte del área de proceso Diseño y ejecución de la prueba.

Lo que debe hacerse específicamente y cómo se realiza la implementación de las pruebas no funcionales depende en gran medida del enfoque definido, las técnicas utilizadas y qué características no funcionales deben probarse y en qué nivel. Por ejemplo, las pruebas exploratorias, que requieren menos preparación, pueden ser adecuadas para las pruebas de usabilidad, pero a menudo son menos adecuadas para pruebas extensas de confiabilidad y rendimiento.

Para algunas características de calidad no funcionales, la disponibilidad de datos de prueba es esencial y debe crearse durante la implementación de la prueba. Esto es en gran medida lo mismo que con los proyectos tradicionales, aparte de que tal vez se documenten de manera breve, lo que permite la reutilización para las pruebas de regresión.

Por supuesto, la implementación y preparación de la prueba comenzará lo antes posible en paralelo a otras actividades (de prueba). No es una fase separada, sino un conjunto de actividades (enumeradas en el tablero de tareas) que deben realizarse para permitir una ejecución de prueba eficiente y efectiva.

3.4.5 SG 5 Realizar ejecución de prueba no funcional

Al igual que con el objetivo anterior en esta área del proceso, nos referiremos en gran medida al objetivo específico relacionado SG 3 Realizar la ejecución de la prueba parte del área del proceso de Diseño y ejecución de la prueba tratada anteriormente en este documento. Las prácticas para la ejecución de pruebas no funcionales, informar incidentes de pruebas y escribir registros de prueba en un entorno ágil son básicamente las mismas que para las pruebas funcionales en un entorno ágil. Por lo general, se realizará con mucha menos intensidad de documentación que en un proyecto tradicional. A menudo no se producen procedimientos de prueba detallados ni registros de prueba.

La ejecución de las pruebas no funcionales se realiza de acuerdo con las prioridades definidas durante la planificación de la iteración. Algunas pruebas pueden ejecutarse utilizando un procedimiento de prueba documentado, pero generalmente también se ejecutarán muchas pruebas no funcionales utilizando pruebas exploratorias y basadas en sesiones como marco de trabajo. Con el desarrollo iterativo hay una mayor necesidad de organizar y estructurar las pruebas de regresión. Esto a veces se hace manualmente, pero preferiblemente se hará usando pruebas de regresión automatizadas y herramientas de soporte. Las pruebas de regresión, por

supuesto, también se aplica a los aspectos no funcionales del sistema que se han identificado como importantes para las pruebas.

El equipo puede registrar e informar los incidentes no funcionales que se encuentran durante las pruebas. Por lo general, existe un debate dentro de los proyectos ágiles sobre si todos los incidentes encontrados deberían registrarse. Algunos equipos solo registran incidentes externos a las iteraciones, algunos lo registran si no se puede solucionar en el momento, otros solo registran incidentes de alta prioridad. Si no se registran todos los incidentes encontrados, los criterios deben estar disponibles para determinar qué incidentes deben registrarse y cuáles no. A veces, los incidentes se registran en el panel de tareas de Ágiles, ya sea como una pegatina en una tarea o como una tarea separada, visualizando que está bloqueando una historia y que sus tareas no se completen. Algunos optan por registrar y documentar los incidentes encontrados utilizando una herramienta, por ejemplo, herramienta de gestión o seguimiento de defectos. Dicha herramienta se debe utilizar de la manera más eficiente posible y no forzar la presentación de ningún elemento de los informes de incidentes de prueba no funcionales que tengan un valor agregado escaso o nulo.

También se considera una buena práctica en los equipos ágiles registrar datos durante la ejecución de las pruebas no funcionales para determinar si los elementos evaluados cumplen con los criterios de aceptación definidos y, de hecho, pueden etiquetarse como "hecho". La información registrada debe ser capturada y/o resumida en alguna forma de herramienta de gestión de estado (por ejemplo, herramientas de gestión de prueba, herramientas de gestión de tareas y tablero de tareas), de manera que sea fácil para el equipo y las partes interesadas comprender el estado actual de todas las pruebas que se realizaron

3.5 Área de proceso 3.2 Revisiones por pares

El propósito del área de proceso de Revisión por pares es verificar que los productos de trabajo cumplan con los requisitos especificados y eliminar los defectos de los productos de trabajo seleccionados de manera temprana y eficiente. Un efecto colateral importante es desarrollar una mejor comprensión de los productos de trabajo y de los defectos que podrían prevenirse.

3.5.1 SG 1 Establecer el alcance de revisión por pares

Por lo general, los equipos ágiles no hacen revisiones formales por pares en el sentido de que generalmente las personas no tienen tiempo asignado para proporcionar comentarios sobre un producto. Sin embargo, logran la intención de las Revisiones por pares al hacer revisiones continuas, por pares, menos formales durante todo el desarrollo. Esta es una práctica común en muchas organizaciones ágiles. Sin embargo, también debe existir una disciplina al realizar estas actividades. Este objetivo de TMMi cubre las prácticas para establecer un enfoque de revisión por pares dentro de un proyecto. El enfoque de revisión define cómo, dónde y cuándo deben realizarse las actividades de revisión y si esas actividades son formales o informales. El establecimiento de un enfoque de revisión por pares también es aplicable a los proyectos ágiles, sin embargo, por lo general, las técnicas de revisión aplicadas y la forma en que se organizan las revisiones es muy diferente.

Ejemplos de revisiones por pares que generalmente se realizan en proyectos ágiles:

- Tener sesiones de perfeccionamiento / preparación sobre las especificaciones (por ejemplo, historias de usuarios) con el equipo y las partes interesadas del negocio de forma regular a lo largo de una iteración;
- reuniones diarias con otros miembros del equipo para discutir abiertamente los productos de trabajo, por ejemplo, código o pruebas, desarrollar y proporcionar comentarios.
- demostración de productos de manera temprana y a menudo con los clientes, al menos al final de una iteración o durante la revisión de iteración;

Las especificaciones pobres son a menudo una de las principales razones del fracaso del proyecto. Los problemas de especificación pueden ser el resultado de la falta de conocimiento de los usuarios sobre sus verdaderas necesidades, la ausencia de una visión global del sistema, características redundantes o contradictorias y otros problemas de comunicación. En el desarrollo ágil, las historias de los usuarios se escriben para capturar los requisitos desde la perspectiva de los representantes comerciales, desarrolladores y evaluadores. En el desarrollo secuencial, esta visión compartida de una característica se logra a través de revisiones formales después de que se

escriben los requisitos. En el desarrollo ágil, esta visión compartida se logra a través de revisiones informales frecuentes mientras se establecen los requisitos. Estas sesiones de revisión informales a menudo se denominan refinamiento del backlog de trabajo o sesiones de preparación del “backlog” de trabajo. Durante las reuniones de refinamiento, el representante comercial y el equipo de desarrollo (y las partes interesadas, si están disponibles) utilizan técnicas de revisión para encontrar el nivel de detalle necesario para la implementación y para aclarar los problemas abiertos.

Las revisiones casi continuas del equipo de los productos de trabajo que se están realizando es parte del enfoque de todo el equipo. Se realizan con la intención de identificar defectos tempranos y también para identificar oportunidades de mejora. Los métodos y técnicas ágiles como XP o emparejamiento también incluyen revisiones por pares como práctica central para crear bucles de retroalimentación para los equipos. Por supuesto, en caso de alta complejidad o riesgo, el equipo puede optar por aplicar una técnica de revisión semiformal o formal, por ejemplo, inspección. En estos casos, existe una razón clara para gastar el esfuerzo en una revisión más formal y aplicar una forma de trabajo más disciplinada. Una parte de los criterios del enfoque de revisión puede definirse cuando se utiliza una técnica de revisión más formal.

Si bien la ingeniería de requisitos enfatiza la validación de requisitos a través de métodos como revisiones informales, tutoriales, inspecciones o lecturas basadas en perspectivas, los métodos ágiles se esfuerzan por validar los requisitos a través de comentarios frecuentes y tempranos para implementar rápidamente incrementos valiosos del producto. La necesidad de una validación formal temprana se reduce al mostrar resultados rápidos en forma de incrementos de producto integrados. Si el incremento no cumple completamente con los requisitos de las partes interesadas, el delta se vuelve a colocar en el backlog del producto en forma de nuevos requisitos y se prioriza con todos los demás elementos del backlog. Una demostración de lo que realmente se construyó durante una iteración es simplemente una forma muy eficiente de impulsar una conversación basada en validación de algo concreto. Nada proporciona más visión a la conversación que ver realmente cómo algo funciona. La demostración es una actividad que se realiza durante la revisión de iteración. Las características se demuestran y se discuten con las partes interesadas, se hacen las adaptaciones necesarias para el backlog del producto o el plan de versión para reflejar los nuevos aprendizajes de la discusión.

3.5.2 SG 2 Realizar las revisiones entre pares

Por supuesto, como cualquier enfoque, no solo debe existir como un enfoque definido y acordado, sino que debe cumplirse. El equipo debe dedicar una cantidad considerable de tiempo a las sesiones de refinamiento del backlog durante una iteración, aplicar revisiones formales informales (y posiblemente) como parte de su rutina diaria, tener demostraciones de las partes interesadas de forma regular. Se esperan demostraciones de partes interesadas/clientes al menos al final de cada iteración.

Los criterios de entrada generalmente solo se aplican con revisiones formales y, por lo tanto, es probable que sean menos o nada relevantes dentro de los proyectos ágiles. Sin embargo, el uso de criterios de salida es aplicable y tiene valor agregado. INVEST [Wake] es un ejemplo de un conjunto de criterios de salida a los que comúnmente se hace referencia en el proyecto ágil cuando se revisan y actualizan historias de usuarios. INVEST es un acrónimo que abarca los siguientes conceptos que forman una buena historia de usuario.

Por supuesto, existen otros criterios de salida relacionados con la calidad y que se pueden usar de manera beneficiosa.

Es importante que el probador, como parte del equipo ágil, participe en sesiones de revisión. Esto es especialmente cierto para las sesiones en las que se tratan y revisan los productos de trabajo que se utilizan como base para realizar pruebas durante la iteración, por ejemplo, en las sesiones de

refinamiento del “backlog”. Se recomienda tener al menos un desarrollador y un probador presentes al refinar el trabajo del “backlog”, para asegurar que haya puntos de vista del sistema alternativos. Por lo general, la perspectiva única del evaluador mejorará las historias del usuario mediante la identificación de detalles faltantes o requisitos no



funcionales. Un probador puede apoyar especialmente en la identificación y definición de criterios de aceptación para una determinada historia de usuario. Un probador contribuye preguntando a los representantes comerciales con preguntas de "¿y si?" sobre la historia del usuario, proponiendo formas de probar la historia del usuario y confirmando los criterios de aceptación.

La Práctica específica 2.3 Analizar los datos de la revisión por pares es especialmente relevante con las revisiones formales. Con las revisiones formales se emplea mucho esfuerzo, para garantizar que el esfuerzo se utilice de manera eficiente y efectiva, los datos de revisión por pares se recopilan y se comunica al equipo para aprender y ajustar el proceso de revisión. Como se indicó, las revisiones formales, por ejemplo, la inspección, son menos comunes en los proyectos ágiles. Mientras que la recopilación de datos es una parte esencial de las revisiones formales, es mucho menos común con la revisión informal. Como resultado, se puede argumentar que la recopilación de datos detallados de revisión por pares, análisis y comunicación es menos o incluso no relevante en un contexto ágil. En el caso de decidir recopilar datos cuando se realizan revisiones por pares, es necesario hacerse las siguientes preguntas:

- ¿Quién va a usar esta información que se está recolectando?
- ¿Cómo se relacionan estos datos con nuestros objetivos comerciales?

Si nadie va a utilizar los datos de manera significativa, no desperdicie recursos valiosos para recopilarlos. Como conclusión, para la mayoría de los proyectos ágiles. Práctica específica 2.3. Se considerará que los datos de análisis por pares no son relevantes. Tenga en cuenta que algunos datos de revisión valiosos básicos aún se recopilarán y utilizarán en el contexto de las prácticas genéricas, por ejemplo, PG 2.8 Supervisar y controlar el proceso.

4 TMMi Nivel 4 Medido

Tal y como se indica al comienzo de este documento, es importante recordar que hay una brecha natural entre el nivel 3 y los niveles 4 y 5 de TMMi. Existe controversia sobre el valor que aporta ir a niveles 4 y 5 de TMMi. Especialmente en organizaciones ágiles se recomienda seleccionar de forma estricta prácticas de los niveles 4 y 5 que aporte valor añadido. Mientras que TMMi se presupone que lleva al éxito, las organizaciones deben identificar las prácticas y mejoras en prueba que sean claves. Es interesante el “post” publicado por Jeff Sutherland, cofundador de Scrum, acerca del valor de utilizar Scrum junto con prácticas de niveles alto de madurez [Sutherland *et al*]. A continuación, se proporciona información con respecto a los niveles 4 y 5 y el valor que aportan estos niveles superiores y sus prácticas, utilizándolas de forma menos formal en un contexto ágil.

En primer lugar, en contraste con lo comentado en el párrafo anterior, se puede ser selectivo utilizando las prácticas de los niveles 4 y 5 con técnicas ágiles para abordar objetivos de negocio críticos, al tratar de alcanzar los niveles 2 y 3. No se debería esperar, es preferible ser flexible y no estancarse en las prácticas de los niveles 2 y 3. Es recomendable utilizar el modelo en base a los “drivers” del negocio y utilizar áreas de proceso, objetivos y prácticas que aporten valor. Por lo tanto, es preferible utilizar TMMi de forma constante y no tratar de ser muy estricto.

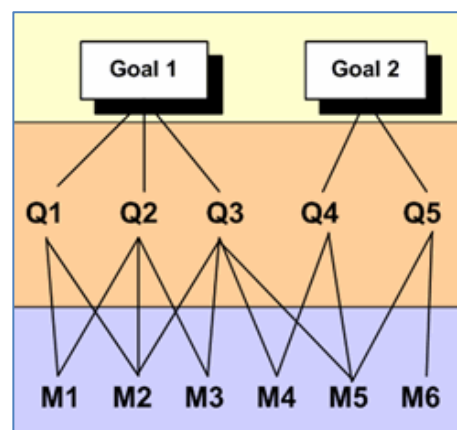
4.1 Área de Proceso 4.1 Medición de Pruebas

El propósito de la Medición de Pruebas es identificar, recolectar, analizar y aplicar mediciones para dar soporte a la organización en la evaluación objetiva de la efectividad y eficiencia del proceso de pruebas, la productividad de su equipo de pruebas, la calidad del producto resultante y los resultados de la mejora de pruebas. Por ello, la organización de pruebas desarrollará y mantendrá una capacidad de medición de pruebas que sirva para dar soporte a las necesidades de información de la gestión.

En un contexto ágil se favorece el compromiso en la medición de forma precisa del trabajo en los proyectos. No obstante, en este contexto, no se tiende a proporcionar mucho enfoque en dar un paso hacia atrás y asegurar que los indicadores correctos son utilizados a lo largo de la organización para asegurar que están alineados con los objetivos de la organización derivados de los objetivos de negocio. Estas son fortalezas la puede proporcionar el área de proceso de Medición de Pruebas de TMMi a organizaciones ágiles.

4.1.1 SG 1 Alinear la Medición de Pruebas con las Actividades de Análisis

En demasiadas ocasiones los programas de métricas parecen estar enfocados en la recolectar cuánta más información posible, mientras que la realidad es que debería estar enfocado en recolectar solo aquello que realmente sea necesario. Comenzando con la definición clara de los objetivos del programa de métricas y revisándolos de forma periódica, “nos debe permitir ser exitosos en nuestros proyectos de pruebas”. A continuación, es necesario compartir con los interesados las métricas de pruebas que realmente darán soporte a esos objetivos. Con este “brainstorming”, se definirán un conjunto limitado de indicadores clave.



Goal-Question-Metric (GQM) [Van Solingen and Berghout] es un método meramente práctico que contempla este planteamiento. Se necesita un contexto específico para las métricas más relevantes en base a los objetivos de negocio. Las métricas estándares de la organización normalmente son insuficientes para la mejora del proceso (de pruebas). El propósito de las mediciones es guiar a las decisiones. Es preferible utilizar pequeños equipos empoderados para generar métricas útiles y posteriormente refinarlas en pequeños ciclos.

Con un programa de métricas de pruebas focalizado en el valor añadido (al negocio), el área de proceso de Medición de Pruebas de TMMi está totalmente alineado con los principios ágiles de simplicidad. Es necesario asegurarse de que existe una estrategia de métricas en la organización. Es importante señalar que se puede comenzar un

programa de métricas a partir de unos cuantos objetivos y mediciones clave y expandirlo a medida que las necesidades de negocio evolucionan.

Hay que tener en cuenta que en entornos ágiles el foco estará basado en el pensamiento de equipo y sistemas. Esto puede resultar que las métricas pueden expandirse al equipo y sistema en su conjunto mas que estar enfocadas en las pruebas. Esto puede derivar en mayores retos en la fase de análisis de indicadores.

Por supuesto, las métricas que serán definidos y para las cuales se recogerán datos, dependerá de los objetivos de medición. A continuación se muestran algunos ejemplos de métricas de pruebas habituales en entornos ágiles:

- Tiempo de ciclo de defecto: Los equipos ágiles deberían esforzarse para corregir defectos tan rápido como sea posible. De hecho, uno de los principales objetivos de los entornos ágiles colaborativos es solucionar defectos de forma rápida para que el software pueda ser entregado antes.
- Numero de defectos diferidos a "versiones" futuras: Uno de los objetivos de los equipos ágiles es producir software que funcione en cada iteración. Este indicador permite medir los defectos que no han sido corregidos durante una iteración o sprint. Algunas organizaciones denominan a este indicador, la deuda técnica.
- Numero de defectos encontrados en una iteración.
- Número de defectos encontrados después del despliegue en producción o en UAT.
- Numero de defectos detectados por personas externas al equipo.
- Número de solicitudes de soporte por parte de los clientes.
- Porcentaje de pruebas automáticas.
- Porcentaje de cobertura de código.

4.1.2 SG 2 Proporcionar Resultados de Mediciones de Pruebas

Una vez que las métricas más importantes han sido definidas en base a las necesidades de negocio, las practicas específicas de este objetivo específico aplican de la misma forma que lo hacen en organizaciones tradicionales. La única diferencia es que éstas se implantarán de forma diferente y liviana.

Los datos de prueba necesarios se recolectan y revisan para asegurar su completitud e integridad. Posteriormente los datos se analizan tal y como está establecido y los resultados se comunican con todos los interesados

4.2 Área de Proceso 4.2 Evaluación de la Calidad de Producto

El propósito de la Evaluación de la Calidad de Producto es desarrollar un entendimiento cuantitativo de la calidad de los productos y dar soporte al cumplimiento de los objetivos de calidad de los productos en los proyectos.

4.2.1 SG 1 Los Objetivos medibles del Proyecto para la Calidad del Producto y sus Prioridades son Establecidos

En esencia no hay diferencia real en contextos ágiles y tradicionales a la hora de identificar y establecer un conjunto de objetivos medibles y priorizados para la calidad del producto. El objetivo global es contribuir a la satisfacción de las necesidades y deseos de los clientes y usuarios finales en cuanto a calidad de producto. Existen muchas opciones para realizar esto en un contexto ágil. En algunos casos se utiliza el "Definition of Done" (DoD) para definir objetivos medibles de calidad de producto, en otros casos se utilizan características, épicas o historias de usuario para definirlos por lo que también utilizarán criterios de aceptación para hacerlos medibles. La opción (forma de trabajar) depende de factores como la naturaleza del objetivo de calidad del producto. Referirse a [ISO25010] para una visión de atributos de calidad.

El "Definition of Done" (DoD) es un "checklist" comprensible de actividades necesarias del equipo y productos de trabajo para asegurar que solo las historias de usuario realmente hechas son entregadas, no solo en términos de funcionalidad sino también de calidad. La ventaja de establecer objetivos de calidad de producto en el DoD es que los mantiene visibles al equipo en todo momento. Obviamente los objetivos de calidad deberán ser globalmente aplicados en el DoD. Si los objetivos de calidad no son globalmente aplicados, probablemente la mejor alternativa será definirlos como funcionalidad o épica.

En algunos casos se especifican un conjunto de atributos de calidad relevantes a nivel de “versión” a alto nivel. Estos serán gradualmente elaborados en las sesiones de refinamiento como una funcionalidad, épica o historia de usuario. Posteriormente, al comienzo de una iteración, el equipo considerará si uno o más de los objetivos de calidad de producto definidos, formarán parte del objetivo para esta iteración específica.

4.2.2 SG 2 PA El progreso actual hacia la consecución de los objetivos de calidad de producto es cuantificado y Gestionado

Una vez que los objetivos de calidad de producto han sido establecidos, deben ser monitorizados, controlados y ajustados cuando sea necesario. Como la calidad de producto ha sido definida como parte del DoD y/o como funcionalidades específicas, épicas o historias de usuario con criterios de aceptación medibles, los mismos métodos y técnicas puede utilizarse para hacer el seguimiento y gestionar el progreso hacia la consecución de los objetivos de calidad de producto tal y como se describen en el área de proceso 2.3 Monitorización y Control de Pruebas del nivel 2. El progreso será medido diariamente y tratado en las reuniones diarias (“daily meetings”), donde se reporta el progreso, calidad y riesgo. En resumen, la Evaluación de la Calidad de Producto normalmente también es soportada por el área de proceso de Medición de Pruebas que es donde se define la infraestructura de métricas.

4.3 Área de Proceso 4.3 Revisiones Avanzadas

El propósito de las Revisiones Avanzadas a partir de las prácticas del área de proceso de Revisiones entre pares de TMMi nivel 3, es la medición de la calidad de producto en fases tempranas del ciclo de vida y mejorar la estrategia y alcance de pruebas alineando las revisiones entre pares (pruebas estáticas) con las pruebas dinámicas.

4.3.1 SG 1 Coordinar el alcance de las “Revisiones entre pares” con el Alcance de Pruebas

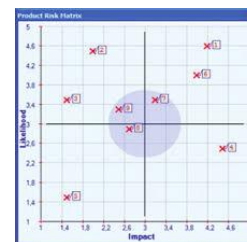
Mientras que en proyectos tradicionales las pruebas estáticas, (ej. revisiones), normalmente no se traza con las pruebas dinámicas realizadas por un equipo de pruebas independiente, en el contexto ágil, esto se espera que ocurra. En un entorno ágil, el alcance para asegurar la calidad de producto debería ser un en el que todas las actividades de pruebas, tanto dinámicas como estáticas se organizan en único alcance coherente y coordinado. Es responsabilidad del equipo ágil integrar las pruebas estáticas y dinámicas hasta obtener la combinación óptima. Se espera que el equipo ágil tome el liderazgo de establecer un alcance integrado coherente y efectivo. Este alcance para pruebas estáticas (revisiones entre pares), deberá estar alineado y coordinado con el alcance de las pruebas dinámicas.

4.3.2 SG 2 Medición de la Calidad de Producto en Fases Tempranas con las Revisiones entre pares

En el nivel 4 de TMMi, la organización ha definido e implantado un conjunto de objetivos cuantificados para los productos de software. Las revisiones entre pares juegan un papel esencial en la consecución de estos objetivos. Mientras que en el nivel 3 de TMMi, las revisiones entre pares son utilizadas sobre todo para detectar defectos, en nivel 4 el énfasis está en la medición de la calidad de los productos/documentos para controlar la calidad de los productos en fases tempranas. Las prácticas de revisiones son mejoradas para incluir prácticas como el muestreo, aplicación de criterios de salida, y la prescripción de reglas. Esto es lo que Tom Gilb denomina inspección ágil, y tiene que ver con pasar de la limpieza al muestreo, medición, motivación y prevención de defectos [Gilb]. Esta es una práctica que se utilizará en una organización con alta madurez. Otro ejemplo serio utilizar las reglas INVEST [Wake] para las historias de usuario como el “Definition of Ready” (DoR) y medir la calidad de las historias de usuario revisándolas frente a estas reglas. Tener un “Definition of Ready” (DoR) significa que las historias deberían ser inmediatamente procesables.

4.3.3 SG 3 Ajustar el Alcance de Pruebas en Base a los Resultados de la revisión en fases tempranas del Ciclo de Vida

Cuando los resultados de las revisiones y las pruebas dinámicas están coordinados, los resultados de las revisiones tempranas y los datos puede influenciar en los riesgos de producto y alcance de pruebas. Tal y como se ha mencionado en el objetivo específico 1 de esta área de proceso, en el contexto ágil la calidad es un esfuerzo del equipo y los resultados de las actividades de verificación y validación son tratadas en las reuniones del equipo. Por lo tanto, el propósito de utilizar los resultados de las pruebas tempranas para influenciar actividades de pruebas posteriores, encaja a la perfección en los equipos y proyectos ágiles.



5 TMMi Nivel 5 Optimización

Los objetivos y prácticas en TMMi nivel 5 se derivan de enfoques de mejora de procesos y gestión de calidad de alta madurez. En este alto nivel de madurez, los objetivos y las prácticas son casi independientes del ciclo de vida del software que se está aplicando. Por supuesto, una organización basada en Lean Ágil abordará las cosas de manera diferente a las organizaciones tradicionales basadas en la jerarquía, pero las prácticas esenciales en torno al control de calidad, la optimización de procesos y la prevención de defectos permanecerán prácticamente sin cambios. La razón de esto es que estas prácticas se consideran prácticas de gestión de la calidad, que normalmente se realizan en gran medida a un nivel superior al nivel de desarrollo de productos operativos. Por tanto, estas prácticas no están directamente relacionadas con las actividades principales de desarrollo y/o prueba de software dentro de un proyecto o flujo de entrega de productos.

5.1 Área de proceso 5.1 Prevención de defectos

El propósito de la Prevención de defectos es identificar y analizar las causas comunes de los defectos a lo largo del ciclo de vida del desarrollo y definir acciones para evitar que se produzcan defectos similares en el futuro.

Por lo general, las organizaciones ágiles ya se centran en la prevención de defectos, que es una base importante sobre la que se pueden construir los objetivos y las prácticas específicas de esta área de proceso. Las organizaciones ágiles eficientes, se centran en la prevención de defectos en lugar de solo el descubrimiento de defectos. Una cultura de prevención de defectos incluye separar los defectos del “trabajo en proceso” de los defectos que escapan más allá de la iteración en la que se desarrollan las historias de usuario.

La retrospectiva ágil es una práctica para que los equipos reflexionen, aprendan y mejoren continuamente en lo que hacen. Aunque las retrospectivas se utilizan con mayor frecuencia para explorar la forma actual de trabajo, también se pueden utilizar para investigar problemas de calidad o para acordar acciones que pueden mejorar la calidad del software que se entrega.

El área de proceso de Prevención de Defectos aborda las prácticas para identificar y analizar las causas comunes de defectos en los equipos, productos y flujos de valor, y definir acciones específicas para eliminar las causas comunes de esos tipos de defectos en el futuro. Todos los defectos, ya sean encontrados durante el desarrollo, las pruebas o en producción, están dentro del alcance del área de proceso. Dado que la prevención de defectos en este nivel necesita datos de medición, la prevención de defectos se basa en las prácticas de medición de nivel 4 de TMMi y los datos de medición disponibles con respecto al desarrollo, las pruebas y la calidad del producto.

Básicamente, existen dos tipos distintos de actividades de prevención de defectos en las organizaciones ágiles: nivel de equipo y entre equipos. Las actividades de prevención de defectos relacionadas con el equipo son responsabilidad del equipo y ya se han abordado en el nivel 3 de TMMi, las actividades de prevención de defectos entre equipos las realiza un grupo de procesos de prueba o un equipo de pruebas y estas son las que se abordan específicamente por esta área de proceso.

5.1.1 SG 1 Determinar causas comunes de defectos

Las mejoras ágiles, incluidas las mejoras del proceso de prueba, generalmente se llevan a cabo en base a frecuentes ciclos de retroalimentación. Debido a que el alcance a menudo se limita a la iteración anterior, se realizan mejoras pequeñas pero frecuentes que se enfocan principalmente en resolver problemas locales específicos. El enfoque de estas mejoras a menudo no está en el aprendizaje entre equipos y la institucionalización de las mejoras. Resolver problemas solo a nivel de equipo también podría conducir fácilmente a una suboptimización y a perder el contacto con el panorama general. Aquí es donde esta área de proceso de TMMi agrega valor, basándose en las prácticas ágiles existentes, la prevención de defectos ahora también se aplica a nivel organizacional (en equipos, productos y flujos de valor) y generalmente administrada por un grupo de procesos (de prueba) o un equipo de pruebas (consulte párrafo 3.1.1).

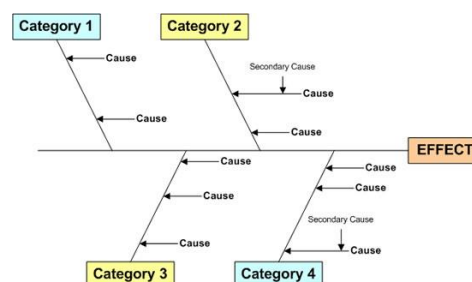
La selección de los defectos que se analizarán debe basarse en varios factores, incluido el riesgo y el valor comercial. Se debe prestar atención a aquellas áreas donde la prevención de defectos tenga el mayor valor agregado

(generalmente en términos de reducción de costo o riesgo) y/o donde los defectos son más críticos. Aunque el área del proceso de prevención de defectos se usa comúnmente para defectos, también se puede usar para problemas por ejemplo relacionados con la velocidad.

Aunque un grupo de procesos de prueba puede coordinar las actividades de prevención de defectos, esto debe ser realizado por un equipo multidisciplinario, por ejemplo, con representantes de ingeniería de requisitos, ingeniería de sistemas y/o desarrollo de software, ya que las acciones de mejora a menudo afectarán a otras disciplinas. Por supuesto, esto ya es común en los equipos y métodos de entrega ágiles. Sin embargo, hay que asegurarse de que estén involucradas todas las disciplinas requeridas, según las acciones de mejora.

Los métodos analíticos para evaluar las causas fundamentales de los defectos, como los diagramas de causa-efecto, los diagramas de espina de pescado o los 5 por qué, se utilizan normalmente para la prevención de defectos y las prácticas ágiles ya habituales.

Las tres prácticas específicas (1.1 Definir los parámetros de selección de defectos, 1.2 Seleccionar defectos para el análisis y 1.3 Analizar las causas raíz y comunes de los defectos seleccionados) según lo definido por el modelo TMMi para este objetivo específico también son aplicables en una organización ágil.



5.1.2 SG 2 Priorizar y definir acciones para eliminar sistemáticamente las causas comunes de defectos

No existen diferencias reales en la realización de prácticas específicas de este objetivo específico, ya sea en un ciclo de vida ágil o secuencial. Por supuesto, la solución real propuesta y la propuesta de mejora posterior serán diferentes, ya que deben adaptarse a la forma de trabajo ágil, por ejemplo, creando elementos adicionales al backlog en lugar de propuestas de acción formalizadas con aprobaciones. Se toman las acciones apropiadas para minimizar la probabilidad de que el problema se repita. A menudo, esto significa combinar la mejora del proceso con tutorías y/o capacitación adicional para los miembros del equipo y otro personal. Es necesario abordar los problemas tanto de las personas como de los procesos de una manera integrada, en lugar de tratar de separarlos artificialmente.

Además, las organizaciones ágiles se basan en equipos autogestionados. Esto implica que los equipos pueden elegir qué mejoras implementar y/o rechazar. Como tal, durante la priorización y definición de acciones, se debe involucrar una representación adecuada de los equipos para negociar si estas mejoras realmente les aportan un valor agregado.

Además, la forma en que se implementan las mejoras será diferente, será más como una sugerencia para los equipos autogestionados. Puede encontrar más información sobre la implementación de mejoras de prueba en el área de proceso 'Optimización del proceso de prueba', descrito en el capítulo 5.3.

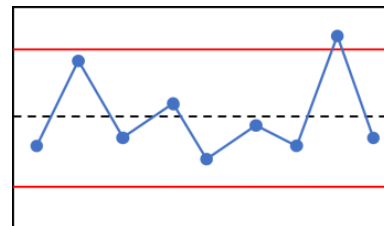
5.2 Área de proceso 5.2 Control de calidad

El objetivo del Control de calidad es gestionar y controlar estadísticamente el proceso de prueba. En este nivel, el rendimiento del proceso de prueba es completamente predecible y se estabiliza dentro de límites aceptables. Las pruebas a nivel de proyecto se realizan utilizando métodos estadísticos basados en muestras representativas para predecir la calidad del producto y hacer que las pruebas sean más eficientes.

5.2.1 SG 1 Establecer un proceso de prueba estadísticamente controlado

Los principios y elementos esenciales para el control de calidad son los mismos independientemente del método de ciclo de vida que se aplique. El control de la calidad del proceso implica el establecimiento de objetivos para el desempeño del proceso de prueba estándar. En base en las mediciones tomadas en el desempeño del proceso de prueba por parte de los equipos, se lleva a cabo el análisis y se realizan ajustes para mantener el desempeño del proceso de prueba dentro de límites aceptables.

Cuando el rendimiento del proceso de prueba se estabiliza dentro de límites aceptables, el proceso de prueba definido, las mediciones asociadas y los límites aceptables para las mediciones se establecen como línea base y se utilizan para controlar estadísticamente el rendimiento del proceso de prueba. La capacidad del proceso de prueba estándar de la organización, es decir, el rendimiento del proceso de prueba que un equipo puede esperar alcanzar, ahora se comprende y se conoce completamente.



Como resultado, las desviaciones de estas expectativas se pueden trabajar en un equipo de manera temprana y consistente para asegurar que se desempeñen dentro de los límites aceptables.

Es importante seleccionar para el control estadístico subprocesos que sean críticos para el negocio y hayan experimentado problemas en el pasado. La experiencia indica que la mayoría de los subprocesos que cumplen mejor con estos criterios no están aislados en equipos individuales, sino que afectan a múltiples equipos, proyectos y flujos de entrega de productos. Cuando se utilizan métodos y técnicas ágiles, el trabajo se divide en iteraciones cortas, donde el trabajo completado en cada iteración implica realizar múltiples actividades de desarrollo y prueba. Por lo tanto, como resultado, el desarrollo ágil contempla de manera más efectiva que el desarrollo tradicional, la monitorización de los subprocesos para el control estadístico, ya que los ciclos son mucho más cortos, lo que genera una retroalimentación casi instantánea sobre el rendimiento del proceso de prueba. La retroalimentación rápida suele ser muy valiosa para los equipos y un importante principio ágil.

5.2.2 SG 2 Las pruebas se realizan utilizando métodos estadísticos

El control de calidad del producto se basa en los perfiles operativos [Musa] y los modelos de uso del producto en su entorno previsto para hacer inferencias estadísticamente válidas que dan como resultado una muestra representativa de pruebas. Este enfoque, especialmente útil a nivel de sistema, utiliza métodos de prueba estadísticos para predecir la calidad del producto basándose en esta muestra representativa. En otras palabras, al probar un subconjunto de todos los usos posibles representados por el perfil de uso operativo, los resultados de la prueba pueden servir como base para las conclusiones sobre el rendimiento general del producto.

Este es básicamente un enfoque que encaja bien con Ágiles, porque es sencillo por naturaleza y está impulsado por los comentarios del uso operativo y las sesiones de revisión sobre el producto. Sin embargo, sí requiere que una organización ágil tenga el nivel adecuado de madurez para poder aplicar este enfoque. También se basa en gran medida y depende de las mediciones y los datos recopilados y disponibles.

Una organización que utiliza métodos estadísticos en las pruebas puede cuantificar los niveles de confianza y la confiabilidad (consulte el marco TMMi para obtener más información sobre estos términos). Tanto el nivel de confianza como la confiabilidad se utilizan normalmente como "Definition of Done" (DoD) cuando se aplican pruebas estadísticas.

5.3 Área de proceso 5.3 Optimización del proceso de pruebas

El propósito de la optimización del proceso de prueba es mejorar continuamente los procesos de prueba existentes utilizados en la organización e identificar nuevas tecnologías de prueba (por ejemplo, herramientas de prueba o métodos de prueba) que pueden ser apropiadas y hacer la transición en la organización de manera ordenada. La optimización del proceso de prueba también admite la reutilización de los activos de prueba en toda la organización. Las mejoras respaldan la calidad del producto de la organización y los objetivos de desempeño del proceso de prueba derivados de los objetivos comerciales de la organización.

5.2.3 SG 1 Seleccionar las mejoras al proceso de pruebas

Las propuestas de mejora de las pruebas se recopilan y analizan. Las propuestas de mejora de la prueba provienen de fuentes internas y, por lo general, entre equipos, proyectos y productos cruzados. Las fuentes internas incluyen problemas e ideas provenientes de actividades de prevención de defectos, revisiones de equipos, retrospectivas, capítulos y grupos. Después de un análisis de costos y beneficios, propuestas de mejora de pruebas que resulten interesantes se pilotan para evaluar el impacto antes de que se implementen en toda la organización. Finalmente, se hace una selección de mejoras de prueba para su implementación en toda la organización.

5.2.3 SG 2 Se evalúan nuevas tecnologías para determinar el impacto en el proceso de prueba

Como parte de la optimización del proceso de prueba, el grupo de proceso de pruebas organizado centralmente (o equipo de pruebas) busca de manera proactiva en el mercado nuevas tecnologías, como herramientas, métodos, técnicas o innovaciones técnicas, que puedan mejorar la capacidad de prueba de los equipos ágiles. Al mantener un conocimiento de las innovaciones tecnológicas relacionadas con las pruebas y evaluarlas y experimentar sistemáticamente con ellas, la organización, en colaboración con los representantes del equipo, selecciona las tecnologías de prueba adecuadas para mejorar la calidad de sus productos y la productividad de sus actividades de prueba. Los pilotos se realizan dentro de los equipos para evaluar tecnologías de prueba nuevas y no probadas antes de que se incorporen a la práctica estándar.

5.2.3 SG 3 Desplegar las mejoras de pruebas

Las mejoras de prueba y las nuevas tecnologías de prueba apropiadas se implementan en los equipos ágiles para mejorar las pruebas. Sus beneficios se miden y la información sobre las nuevas innovaciones se difunde entre las organizaciones. Por supuesto, existe una diferencia entre la implementación en una organización tradicional y una organización ágil. En una organización tradicional, la implementación suele dar como resultado nuevas prácticas que deben ser aplicadas por todos los involucrados. En una organización ágil el equipo tiene más autonomía y se autogestiona. Esto también implica que el proceso de implementación es diferente. Las mejoras de prueba y las nuevas tecnologías de prueba apropiadas se incorporan en los activos del proceso de prueba de la organización según corresponda, se organizan y ofrecen capacitación y soporte. Sin embargo, al final, el equipo ágil debe decidir si la mejora de la prueba se convertirá o no en parte de su forma de trabajar.

5.2.3 SG 4 Establecer la reutilización de activos de prueba de alta calidad

En el nivel 3 de TMMi, es posible que ya se lleve a cabo cierta reutilización de software de prueba en equipos, proyectos y flujos de productos; sin embargo, la reutilización de los activos de prueba se convierte en un objetivo importante en el nivel 5 de TMMi. El equipo no debe reinventar la rueda, sino aprovechar la experiencia y los activos ya disponibles y, como tal, ahorrar tiempo y esfuerzo. Se identifican los activos de prueba de alta calidad que probablemente puedan reutilizarse en toda la organización, por ejemplo, en reuniones retrospectivas, sesiones de lecciones aprendidas, discusiones dentro del grupo de pruebas o evaluaciones de pruebas. Posteriormente, los activos de prueba reutilizables se seleccionan y se agregan al repositorio de prueba central en un formato que se puede modificar para su reutilización en proyectos y entrega de productos futuros.

6 Resumen de relevancia de Objetivos y Practicas Específicas TMMi

6.1 Assessments TMMi

El objetivo de este capítulo es dar soporte a los 'Assessors' y 'Lead Assessors' en cuanto a la relevancia de los objetivos y prácticas específicas de TMMi en el contexto ágil. En los capítulos anteriores se han identificado un número limitado de prácticas específicas que son menos relevantes en un entorno ágil. Estas prácticas se listan a continuación. Durante un Assessment TMMi, el 'Lead Assessor', o 'Assessor', tendrá que evaluar de forma explícita estos elementos para indicar su relevancia en un contexto ágil. En caso de que se determine que un elemento es realmente menos relevante o incluso no relevante esto debe ser documentado de forma razonada tanto en el proceso de la organización como en el informe del Assessment.

6.2 Nivel 2 de TMMi, Gestionado

Área de Proceso 2.2 Planificación de Pruebas

SG2 Establecer el Alcance de las Pruebas	Aplica
- SP2.3 Definir criterios de entrada	Poco relevante
- SP2.5 Definir los criterios de suspensión y de reanudación	Poco relevante
SG3 Establecer Estimaciones de Pruebas	Aplica
- S3.2 Definir del ciclo de vida de pruebas	Poco relevante
SG5 Obtener el compromiso con el Plan de Pruebas	Aplica
- SP5.2 Reconciliar el trabajo y los recursos	Poco relevante

Área de Proceso 2.3 Monitorización y Control de Pruebas

SG2 Monitorizar la Calidad de Producto contra el Plan y las Expectativas	Aplica
- SP2.3 Monitorizar los criterios de entrada	Poco relevante
- SP2.5 Monitorizar los criterios de suspensión y reanudación	Poco relevante

Área de Proceso 2.4 Diseño y Ejecución de Pruebas

SG2 Realizar la Implementación de Pruebas	Aplica
- SP2.3 Especificar las pruebas de humo	Poco relevante
- SP2.4 Desarrollar el calendario de ejecución	Poco relevante
SG3 Realizar la Ejecución de Pruebas	Aplica
- SP3.1 Ejecutar las Pruebas de Humo	Poco relevante

6.3 Nivel 3 de TMMi, Definido

Área de Proceso 3.5, Revisiones por Pares

SG2 Realizar Revisiones por Pares	Aplica
- SP2.3 Analizar los datos de las revisiones por pares	Poco relevante

6.4 Nivel 4 de TMMi, Medido

Todas las áreas de proceso del nivel 4 así como los objetivos y prácticas específicas aplican. No existen prácticas específicas que no sean relevantes.

6.5 Nivel 5 de TMMi, Optimización

Todas las áreas de proceso del nivel 5 así como los objetivos y prácticas específicas aplican. No existen prácticas específicas que no sean relevantes.

Referencias

- [Black, Van Veenendaal] Rex Black, Erik van Veenendaal and Dorothy Graham (2012), *Foundations of Software Testing ISTQB Certification* (3rd edition), Cengage
- [Cohn] M. Cohn (2009), *Succeeding with Ágil: Software Development using Scrum*, Addison-Wesley
- [Galen] R. Galen (2015), *Three Pillars of Ágil Quality and Testing: Achieving Balanced Results in your Journey Towards Ágil Quality*, RGCG, LLC
- [Gilb] T. Gilb (2004), *Ágil Specification Quality Control*, International Council on Systems Engineering (INCOSE)
- [ISO25010] ISO/IEC 25010 (2011), *Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) System and Software Quality Models*, International Organization for Standardization
- [Sutherland et al] C.R. Jakobsen, K. Johnson and J. Sutherland (2007), *Scrum and CMMI level 5: The Magic Portion for Code Warriors* in: IEEE Ágil 2007 Conference, Washington, D.C.
- [McMahon] Paul. E. McMahon (2011), *Integrating CMMI with Ágil Development – Case Studies and Proven Techniques for Faster Performance Improvement*, Addison Wesley
- [Musa] J. Musa (1998), *Software Reliability Engineering Testing*, McGraw-Hill Education
- [Van der Aalst and Davis] L. van der Aalst and Cecile Davis (2013), *TMap NEXT in scrum – Effective testing in ágil projects*, Sogeti Nederland B.V.
- [Van Solingen and Berghout] R. van Solingen and E. Berghout (1999), *The Goal/Question/Metric method*, McGrawHill
- [van Veenendaal] E. van Veenendaal (2014), *PRISMA: Product Risk Assessment for Ágil Projects*, in: *Testing Experience*, Issue 04/12, December 2012
- [Wake] B. Wake (2003), *INVEST in Good Stories, and SMART Tasks*, xp123.com/articles/invest-in-good-stories-and-smart-tasks