

Probador Certificado del ISTQB®

Programa de Estudio de Nivel Avanzado

Jefe de Pruebas

Traducción realizada por el
Spanish Software Testing Qualifications Board
Versión ES 001.08

Basada en el Programa de Estudio
“Certified Tester - Advanced Level Syllabus, Test Manager,
Version 2012”

International Software Testing Qualifications Board



Nota sobre derechos de propiedad intelectual

El presente documento podrá ser copiado parcial o íntegramente siempre y cuando se cite la fuente.
Copyright © International Software Testing Qualifications Board (en adelante denominado ISTQB®).

Subgrupo de Trabajo de Jefes de Pruebas de Nivel Avanzado Rex Black (Presidente), Judy McKay (Vicepresidenta), Graham Bath, Debra Friedenberg, Bernard Homès, Kenji Onishi, Mike Smith, Geoff Thompson, Tsuyoshi Yumoto; 2010-2012.



Historial de Revisiones

Versión	Fecha	Observaciones
ISEB v1.1	04SEP01	ISEB Practitioner Syllabus
ISTQB 1.2E	SEP03	Programa de Estudio de Nivel Avanzado ISTQB de EOQ-SG
V2007	12OCT07	Programa de Estudio de Probador Certificado de Nivel Avanzado, versión 2007
D100626	26JUN10	Incorporación de cambios aceptados en 2009, separación de cada capítulo para los distintos módulos
D101227	27DIC10	Aceptación de cambios de formato y correcciones que no afectan al significado de las frases.
D2011	31OCT11	Cambio para dividir los programas de estudio, objetivos de aprendizaje reelaborados y cambios en el texto para coincidir con los objetivos de aprendizaje. Incorporación de objetivos de aprendizaje.
Alfa 2012	09Feb12	Incorporación de todos los comentarios derivados de los NB recibidos a partir de la entrega de octubre.
Beta 2012	26Mar12	Incorporación de comentarios derivados de los NB recibidos a tiempo de la entrega Alfa
Beta 2012	07ABR12	Versión Beta enviada a GA
Beta 2012	08JUN12	Versión editada entregada a NB
Beta 2012	27JUN12	Incorporación de comentarios EWG y Glosario
RC 2012	15AGO12	Versión candidata de entrega - ediciones NB finales incluidas
GA 2012	19OCT12	Ediciones finales y limpieza para entrega GA

Índice general

Historial de Revisiones.....	3
Índice general.....	4
Agradecimientos	6
0. Introducción a este Programa de Estudio	8
0.1 Objetivo de este documento	8
0.2 Generalidades	8
0.3 Objetivos de aprendizaje sujetos a examen	8
1. Proceso de pruebas - 420 minutos	9
1.1 Introducción	10
1.2 Planificación, monitorización y control de pruebas	10
1.2.1 Planificación de pruebas	10
1.2.2 Monitorización y control de pruebas	11
1.3 Análisis de pruebas.....	12
1.4 Diseño de pruebas.....	14
1.5 Implementación de pruebas.....	15
1.6 Ejecución de pruebas	16
1.7 Evaluación de los criterios de salida y elaboración de informes.....	16
1.8 Actividades de cierre de pruebas.....	17
2. Gestión de pruebas – 750 minutos	19
2.1 Introducción	21
2.2 Gestionar las pruebas dentro de un contexto	21
2.2.1 Comprender a las partes interesadas de las pruebas	21
2.2.2 Actividades adicionales del ciclo de vida de desarrollo de software y productos de trabajo	22
2.2.3 Alineación de las actividades de pruebas y demás actividades del ciclo de vida.....	23
2.2.4 Gestión de pruebas no funcionales	25
2.2.5 Gestión de pruebas basadas en la experiencia.....	26
2.3 Pruebas basadas en riesgos y otros enfoques para la priorización de las pruebas y la asignación del esfuerzo	27
2.3.1 Pruebas basadas en riesgos	27
2.3.2 Técnicas de pruebas basadas en riesgos	32
2.3.3 Otras técnicas para la selección de pruebas	35
2.3.4 Priorización de las pruebas y asignación del esfuerzo en el proceso de pruebas	36
2.4 Documentación de pruebas y otros productos de trabajo.....	37
2.4.1 Política de pruebas.....	37
2.4.2 Estrategia de pruebas	38
2.4.3 Plan maestro de pruebas	40
2.4.4 Plan de nivel de prueba	41
2.4.5 Gestión de riesgos del proyecto	41
2.4.6 Otros productos de pruebas	42
2.5 Estimación de pruebas	42
2.6 Definición y uso de las métricas de pruebas	44
2.7 Valor de negocio de las pruebas	49
2.8 Pruebas distribuidas, externalizadas e internalizadas	50
2.9 Gestión de la aplicación de los estándares del sector	51
3. Revisiones - 180 minutos	53
3.1 Introducción	54
3.2 Revisiones de gestión y auditorías	55
3.3 Revisiones de gestión.....	56
3.4 Métricas para las revisiones	58
3.5 Gestión de revisiones formales.....	59
4. Gestión de defectos – 150 minutos.....	60
4.1 Introducción	61
4.2 El ciclo de vida de los defectos y el ciclo de vida de desarrollo de software	61

4.2.1 Flujo de trabajo de defectos y estados	62
4.2.2 Gestión de informes de defectos no válidos y duplicados	62
4.2.3 Gestión de defectos transversal	63
4.3 Información de los informes de defectos	63
4.4 Evaluar la capacidad del proceso con información del informe de defectos	65
5. Mejora del proceso de pruebas - 135 minutos	66
5.1 Introducción	67
5.2 Proceso de mejora de pruebas	67
5.2.1 Introducción al Proceso de Mejora	67
5.2.2 Tipos de mejora de procesos	68
5.3 Mejora del proceso de pruebas	68
5.4 Mejorar el proceso de pruebas con TMMi	69
5.5 Mejorar el proceso de pruebas con TPI Next	70
5.6 Mejorar el proceso de pruebas con CTP	70
5.7 Mejorar el proceso de pruebas con STEP	71
6. Herramientas y automatización de pruebas - 135 min	72
6.1 Introducción	73
6.2 Selección de herramientas	73
6.2.1 Herramientas de código abierto	73
6.2.2 Herramientas a medida	74
6.2.3 Retorno sobre la inversión (ROI)	74
6.2.4 Proceso de selección	76
6.3 Ciclo de vida de las herramientas	77
6.4 Métricas de las herramientas	78
7. Habilidades de las personas – Composición del equipo – 210 minutos	79
7.1 Introducción	80
7.2 Habilidades individuales	80
7.3 Dinámica del equipo de pruebas	82
7.4 Integrar las pruebas dentro de una organización	83
7.5 Motivación	85
7.6 Comunicación	86
8. Referencias	88
8.1 Estándares	88
8.2 Documentos ISTQB	88
8.3 Marcas comerciales	89
8.4 Referencias bibliográficas	89
8.5 Otras referencias	90
9. Índice terminológico	91

Agradecimientos

Este documento ha sido elaborado por un equipo central del Subgrupo de Trabajo de Nivel Avanzado de Jefes de Pruebas Avanzados del International Software Testing Qualifications Board: Rex Black (Presidente), Judy McKay (Vicepresidenta), Graham Bath, Debra Friedenber, Bernard Homès, Paul Jorgensen, Kenji Onishi, Mike Smith, Geoff Thompson, Erik van Veenendaal, Tsuyoshi Yumoto.

El equipo central agradece las sugerencias y aportaciones del equipo de revisión y de los comités nacionales.

Cuando se finalizó el programa de estudio del Nivel Avanzado, formaban parte del Grupo de Trabajo de Nivel Avanzado los siguientes miembros (por orden alfabético):

Graham Bath, Rex Black, Maria Clara Choucair, Debra Friedenber, Bernard Homès (Vicepresidente), Paul Jorgensen, Judy McKay, Jamie Mitchell, Thomas Mueller, Klaus Olsen, Kenji Onishi, Meile Posthuma, Eric Riou du Cosquer, Jan Sabak, Hans Schaefer, Mike Smith (Presidente), Geoff Thompson, Erik van Veenendaal, Tsuyoshi Yumoto.

Las siguientes personas participaron en la revisión, comentarios y votación del presente programa de estudio:

Chris van Bael, Graham Bath, Kimmo Hakala, Rob Hendriks, Marcel Kwakernaak, Rik Marselis, Don Mills, Gary Mogyorodi, Thomas Mueller, Ingvar Nordstrom, Katja Piroué, Miele Posthuma, Nathalie Rooseboom de Vries, Geoff Thompson, Jamil Wahbeh, Hans Weiberg.

Este documento fue hecho público formalmente por la Asamblea General del ISTQB® el 19 de octubre de 2012.

Xavier Gallart del Burgo (Spanish Software Testing Qualifications Board) ha revisado la traducción al español de este programa de estudio.



0. Introducción a este Programa de Estudio

0.1 Objetivo de este documento

Este programa de estudio constituye la base de la Cualificación Internacional de Pruebas de Software de Nivel Avanzado para los Jefes de Pruebas. El ISTQB® facilita el presente programa de estudio en las siguientes condiciones:

1. A los comités nacionales, para que lo traduzcan a su idioma local y para acreditar a los proveedores de formación. Los comités nacionales podrán adaptar el programa de estudio a las necesidades específicas de su idioma y modificar las referencias para adaptarlas a sus publicaciones locales.
2. A los comités examinadores, para que puedan crear preguntas de examen en su idioma local que se adapten a los objetivos de aprendizaje de cada programa de estudio.
3. A los proveedores de formación, para que elaboren los materiales didácticos y determinen las metodologías de enseñanza apropiadas.
4. A los candidatos a la certificación, para que se preparen para el examen (como parte de un curso de formación o de manera independiente).
5. A la comunidad internacional de ingeniería de sistemas y software, para que la actividad de pruebas de software y sistemas avance, y como base para la elaboración de libros y artículos.

El ISTQB® podrá autorizar que otras entidades utilicen este programa de estudio con otros fines, siempre y cuando soliciten y obtengan la correspondiente autorización previa por escrito.

0.2 Generalidades

El Nivel Avanzado consta de tres programas de estudio independientes:

- Jefe de Pruebas
- Analista de Pruebas
- Analista Técnico de Pruebas

El documento Resumen del Nivel Avanzado [ISTQB_AL_OVIEW] incluye la siguiente información:

- Resultados de negocio de cada programa de estudio
- Resumen de cada programa de estudio
- Relaciones entre los programas de estudio
- Descripción de los niveles cognitivos (niveles K)
- Anexos

0.3 Objetivos de aprendizaje sujetos a examen

Los objetivos de aprendizaje respaldan los resultados de negocio y sirven para establecer el examen para lograr la Certificación de Jefe de Pruebas de Nivel Avanzado. Por lo general, todas las partes de este programa de estudio pueden ser objeto de examen en el nivel K1. Es decir, el candidato reconocerá, memorizará y recordará un término o un concepto. Los objetivos de aprendizaje correspondientes a los niveles K2, K3 y K4 se muestran al principio de cada capítulo.

1. Proceso de pruebas - 420 minutos

Palabras clave

criterios de salida, caso de prueba, cierre de pruebas, condición de prueba, control de pruebas, diseño de pruebas, ejecución de pruebas, implementación de pruebas, bitácora de la prueba, planificación de prueba, procedimiento de prueba, guion de prueba, informe resumen de pruebas

Objetivos de aprendizaje del Proceso de pruebas

1.2 Monitorización, planificación y control de pruebas

TM-1.2.1 (K4) Analizar las necesidades de un sistema para planificar las actividades de pruebas y los productos de trabajo que lograrán los objetivos de prueba

1.3 Análisis de pruebas

TM-1.3.1 (K3) Usar la trazabilidad para comprobar la integridad y la coherencia de las condiciones de prueba definidas por lo que respecta a los objetivos de prueba, la estrategia de pruebas y el plan de pruebas

TM-1.3.2 (K2) Explicar los factores que podrían afectar al nivel de detalle en el que pueden especificarse las condiciones de prueba y la ventajas e inconvenientes de especificar condiciones de prueba en un nivel detallado

1.4 Diseño de pruebas

TM-1.4.1 (K3) Usar la trazabilidad para comprobar la integridad y la consistencia de los casos de prueba diseñados por lo que respecta a las condiciones de prueba definidas

1.5 Implementación de pruebas

TM-1.5.1 (K3) Usar riesgos, priorización, entornos de pruebas y dependencias de datos, y restricciones para desarrollar un calendario de ejecución de pruebas que sea completo y coherente por lo que respecta a los objetivos de prueba, la estrategia de pruebas y el plan de pruebas

1.6 Ejecución de pruebas

TM-1.6.1 (K3) Usar la trazabilidad para monitorizar el progreso de las pruebas en cuanto a integridad y coherencia con los objetivos de prueba, la estrategia de pruebas y el plan de pruebas

1.7 Evaluación de los criterios de salida y elaboración de informes

TM-1.7.1 (K2) Explicar la importancia de recabar información exacta y oportuna durante el proceso de pruebas para fomentar la correcta elaboración de informes y evaluaciones en base a los criterios de salida

1.8 Actividades de cierre de pruebas

TM-1.8.1 (K2) Resumir los cuatro grupos de actividades de cierre de pruebas

TM-1.8.2 (K3) Implementar una retrospectiva de proyecto para evaluar los procesos y descubrir áreas de mejora.

1.1 Introducción

El Programa de Estudio de Nivel Fundamentos ISTQB® describe un proceso de pruebas básico en el que se incluyen las siguientes actividades:

- Planificación y control
- Análisis y diseño
- Implementación y ejecución
- Evaluación de los criterios de salida y elaboración de informes
- Actividades de cierre de pruebas

El Programa de Estudio de Nivel Fundamentos establece que, si bien son lógicamente secuenciales, las actividades del proceso pueden solaparse o producirse de manera simultánea. Normalmente estas actividades principales deben adaptarse al contexto del sistema y al proyecto.

En los Programas de Estudio de Nivel Avanzado, algunas de estas actividades se abordan de manera independiente para ofrecer un mayor perfeccionamiento y optimización de los procesos, para ajustarse mejor al ciclo de vida de desarrollo de software, y para facilitar la monitorización y el control efectivo de las pruebas. En este programa las actividades se tienen en cuenta como sigue:

- Planificación, monitorización y control
- Análisis
- Diseño
- Implementación
- Ejecución
- Evaluación de los criterios de salida y elaboración de informes
- Actividades de cierre de pruebas

1.2 Planificación, monitorización y control de pruebas

Esta sección se centra en los procesos de planificación, monitorización y control de las pruebas. Según lo estudiado en el Nivel Fundamentos, estas actividades son competencia del Jefe de Pruebas.

1.2.1 Planificación de pruebas

Para cada nivel de pruebas, la planificación de las pruebas empieza al inicio del proceso de pruebas para ese nivel y continúa durante el proyecto hasta completar las actividades de cierre correspondientes a ese nivel. La planificación implica la identificación de las actividades y los recursos necesarios para cumplir la misión y los objetivos previstos en la estrategia de pruebas. La planificación de pruebas también incluye la identificación de los métodos para recopilar y controlar las métricas que se utilizarán para guiar el proyecto, determinar el cumplimiento del plan y evaluar la consecución de los objetivos. Durante las etapas de planificación se puede determinar métricas útiles, seleccionar herramientas, programar actividades de formación y establecer directrices de documentación.

La estrategia (o estrategias) seleccionada para el proyecto de pruebas permite determinar qué tareas deberían realizarse durante las fases de planificación. Así por ejemplo, si se adopta una estrategia de pruebas basada en riesgos (remítase al Capítulo 2), el análisis de riesgos servirá para guiar el proceso de planificación de pruebas por lo que respecta a las actividades de mitigación necesarias para reducir los riesgos del producto identificados y para ayudar a la planificación de contingencias. Si se identifican varios defectos potenciales de seguridad probables y graves, debería dedicarse una gran cantidad de esfuerzo a desarrollar y ejecutar pruebas de seguridad. Asimismo, si se concluye que en la especificación de diseño normalmente se detectan defectos graves, el proceso de

planificación de las pruebas podría resultar en pruebas estáticas adicionales (revisiones) de la especificación de diseño.

La información sobre riesgos también puede utilizarse para establecer las prioridades de las distintas actividades de pruebas. Así por ejemplo, si el rendimiento de un sistema plantea un riesgo alto, se podrán realizar las pruebas de rendimiento en cuanto el código esté disponible. De la misma forma, si se emplea una estrategia reactiva, podrá garantizarse la planificación para la creación de contratos de pruebas y herramientas para técnicas de pruebas dinámicas, tales como las pruebas exploratorias.

Además, es en la fase de planificación de pruebas cuando el Jefe de Pruebas define claramente el enfoque de las pruebas, incluyendo qué niveles van a utilizarse, las metas y objetivos de cada nivel, y qué técnicas de pruebas se utilizarán en cada nivel de prueba. Así por ejemplo, en las pruebas basadas en riesgos de algunos sistemas de aviónica, una evaluación de riesgos establece el nivel de cobertura del código necesario y, en consecuencia, las técnicas de pruebas que deberían aplicarse.

Pueden existir relaciones complejas entre la base de prueba (por ejemplo, requisitos o riesgos específicos), las condiciones de prueba y las pruebas que las cubren. A menudo se dan relaciones múltiples entre estos productos de trabajo. Para lograr una planificación, una monitorización y un control efectivos de las pruebas, es necesario conocer estas relaciones. Las decisiones sobre herramientas también pueden depender del conocimiento de las relaciones que existen entre los productos de trabajo.

Pueden existir relaciones entre productos de trabajo creados por el equipo de desarrollo y el equipo de pruebas. Por ejemplo, la matriz de trazabilidad puede tener que trazar las relaciones entre los elementos de la especificación de diseño detallado por los diseñadores de sistema, los requisitos de negocio a partir de los analistas de negocio, y los productos de trabajo de las pruebas definidos por el equipo de pruebas. En el caso de que se diseñen y utilicen casos de prueba de bajo nivel, en las fases de planificación puede haberse definido un requisito según el cual los documentos de diseño detallado del equipo de desarrollo deben aprobarse antes de poder iniciar la creación de los casos de prueba. Si se sigue un ciclo de vida Ágil, las sesiones informales de traspaso de información pueden servir para intercambiar información entre los equipos antes de empezar las pruebas.

El plan de pruebas también puede enumerar las prestaciones específicas del software que están dentro de su alcance (tomando como base el análisis de riesgos, si procede), así como identificar de manera explícita aquellas prestaciones que no están dentro de su alcance. En función de los niveles de formalidad y documentación adecuados para el proyecto, cada una de las prestaciones incluida en el alcance podrá estar asociada a una correspondiente especificación de diseño de prueba.

En esta fase también se podrá pedir al Jefe de Pruebas que trabaje con los arquitectos del proyecto para definir las especificaciones del entorno de pruebas inicial, que compruebe la disponibilidad de los recursos necesarios, que garantice que las personas que configuran el entorno están comprometidas a hacerlo, y que sea consciente las restricciones de costes y los plazos de entrega necesarios para finalizar y entregar el entorno de pruebas.

Por último, deberían identificarse todas las dependencias externas y los acuerdos de nivel de servicios (ANS) asociados y, si procede, deberían hacerse los primeros contactos. Algunos ejemplos de dependencias son las peticiones de recursos a grupos externos, a dependencias de otros proyectos (si está trabajando en el marco de un programa), a proveedores externos o socios de desarrollo, al equipo de despliegue y a administradores de bases de datos.

1.2.2 Monitorización y control de pruebas

Para que el Jefe de Pruebas pueda ejercer un control efectivo de las pruebas, este debe establecerse un calendario de pruebas y un marco de monitorización que permitan localizar los productos de trabajo de las pruebas y los recursos en base al plan. Este marco debería incluir el detalle de las

medidas y metas necesarias para relacionar el estado de los productos de trabajo de las pruebas y las actividades con el plan y los objetivos estratégicos.

Para proyectos pequeños y menos complejos, puede ser relativamente sencillo relacionar los productos de trabajo de las pruebas y las actividades con el plan y los objetivos estratégicos, pero en general para lograrlo deben definirse objetivos más detallados, tales como las medidas y metas para alcanzar los objetivos y la cobertura de prueba de la base de prueba.

Es importante relacionar el estado de los productos de trabajo de las pruebas y las actividades con la base de prueba de una manera que sea comprensible y relevante tanto para el proyecto como para las partes interesadas del negocio. La definición de las metas y la medición de los objetivos tomando como base condiciones de prueba y grupos de condiciones de prueba puede ser una forma de conseguirlo relacionando otros productos de trabajo de las pruebas con la base de prueba a través de las condiciones de prueba. Una trazabilidad debidamente configurada, que incluya la capacidad de elaborar informes sobre el estado de trazabilidad, hace que las relaciones complejas existentes entre los productos de trabajo, la base de prueba y los productos de trabajo de las pruebas sean más transparentes y comprensibles.

A veces, las medidas y metas detalladas que las partes interesadas quieren monitorizar no se refieren directamente a la funcionalidad o especificación del sistema, especialmente si se dispone de poca o ninguna documentación formal. Por ejemplo, una parte interesada del negocio puede tener más interés en establecer la cobertura en base a un ciclo de negocio operativo, aunque la especificación esté definida en términos de la funcionalidad del sistema. La intervención de las partes interesadas del negocio en una fase temprana de un proyecto puede ayudar a definir estas medidas y metas, las cuales no solo pueden ayudar a obtener un mejor control durante el proyecto, sino también a impulsar e influenciar las actividades de pruebas a lo largo del proyecto. Así por ejemplo, las medidas y metas de las partes interesadas pueden suponer la estructuración de los productos de trabajo de diseño e implementación de pruebas y/o los calendarios de ejecución de las pruebas de manera que faciliten una correcta monitorización del progreso de las pruebas en base a estas medidas. Estas metas también permiten lograr una mejor trazabilidad en un nivel de prueba específico y ayudan a facilitar trazabilidad de la información en diferentes niveles de prueba.

El control de las pruebas es una actividad constante que supone comparar el progreso real contra el plan previsto e implementar acciones correctivas en caso necesario. El control de pruebas guiará el proceso de pruebas hacia la consecución de su misión, estrategia y objetivos, incluyendo la revisión de las actividades de planificación de pruebas en caso necesario. Las reacciones adecuadas ante los datos de control dependerán del nivel de detalle de la información de planificación.

Remítase al Capítulo 2 para más información sobre el contenido de los documentos de planificación de las pruebas y de las actividades de control de las pruebas.

1.3 Análisis de pruebas

En lugar de considerar el análisis y el diseño de las pruebas juntos según lo indicado en el Programa de Estudio de Nivel Fundamentos, los programas avanzados consideran que se trata de actividades independientes, reconociendo incluso que pueden implementarse como actividades paralelas, integradas o iterativas para facilitar la producción de productos de trabajo de diseño de pruebas.

El análisis de pruebas es la actividad que establece “qué” debe probarse como condición de prueba. Las condiciones de prueba pueden identificarse analizando la base de prueba, los objetivos de prueba y los riesgos del producto. Pueden verse como las medidas y metas detalladas para lograr el éxito (por ejemplo, como parte de los criterios de salida) y deberían ser trazables hasta la base de prueba y los objetivos estratégicos definidos, incluidos los objetivos de prueba y demás criterios de éxito del proyecto o de las partes interesadas. Las condiciones de prueba también deberían ser

trazables hasta los diseños de prueba y demás productos de trabajo de las pruebas a medida que dichos productos de trabajo van creándose.

El análisis de pruebas de un nivel de pruebas dado puede realizarse a partir del momento en que se establezca una base de prueba para ese nivel. Para identificar las condiciones de prueba pueden emplearse técnicas de pruebas formales y otras técnicas analíticas generales (tales como estrategias analíticas basadas en riesgos y estrategias analíticas basadas en requisitos). Las condiciones de prueba pueden especificar o no los valores o las variables, en función del nivel de pruebas, de la información disponible en el momento de la ejecución del análisis y del nivel de detalle seleccionado (es decir, el nivel de granularidad de la documentación).

A la hora de decidir sobre el nivel de detalle en el que deben especificarse las condiciones de prueba, deben tenerse en cuenta los siguientes factores, entre otros:

- Nivel de prueba
- Nivel de detalle y calidad de la base de prueba
- Complejidad del sistema/software
- Riesgo del proyecto y del producto
- Relación entre la base de prueba, qué debe probarse y cómo debe probarse
- Ciclo de vida de desarrollo de software empleado
- Herramienta de gestión de pruebas empleada
- Nivel en el que deben especificarse y documentarse el diseño y otros productos de trabajo de las pruebas
- Aptitudes y conocimientos de los analistas de pruebas
- Nivel de madurez del proceso de pruebas y de la propia organización (observe que cuanto más madurez exista mayor puede ser el nivel de detalle requerido o menor el nivel de detalle permitido)
- Disponibilidad de otras partes interesadas del proyecto para consulta

La especificación detallada de las condiciones de prueba tiende a dar lugar a un mayor número de condiciones de prueba. Por ejemplo, usted puede tener una única condición de prueba general, "Comprobación de la prueba", para una aplicación de comercio electrónico. Sin embargo, en un documento detallado de condiciones de prueba, esta condición puede estar dividida en varias condiciones de prueba, con una condición por cada forma de pago soportada, una condición por cada país de destino posible, etc.

Algunas ventajas que supone especificar las condiciones de prueba en detalle son:

- Permite una mayor flexibilidad a la hora de relacionar otros productos de trabajo de las pruebas (tales como casos de prueba) con la base de prueba y los objetivos de prueba, lo que hace que el Jefe de Pruebas pueda disponer de funciones de seguimiento y control mejoradas y más detalladas
- Contribuye a prevenir defectos, según lo descrito en el Nivel Fundamentos, ya que se lleva a cabo en una etapa temprana del proyecto y se refiere a niveles de prueba más altos, a partir del momento en que se establezca la base de prueba y posiblemente antes de disponer de la arquitectura del sistema y del diseño detallado
- Relaciona los productos de trabajo de las pruebas con las partes interesadas en términos que estas pueden comprender (a menudo, los casos de prueba y otros productos de trabajo de las pruebas no significan nada para las partes interesadas del negocio, y las métricas sencillas, como el número de casos de prueba ejecutados, no significan nada para los requisitos de cobertura de las partes interesadas)
- Ayuda a influenciar y orientar no solo otras actividades de pruebas, sino también otras actividades de desarrollo

- Permite optimizar el diseño, la implementación y la ejecución de las pruebas, junto con los productos de trabajo resultantes, mediante una cobertura más eficiente de medidas y metas más detalladas
- Establece la base para una trazabilidad horizontal más clara dentro de un nivel de prueba

Algunos inconvenientes que supone especificar las condiciones de prueba en detalle son:

- Puede llevar mucho tiempo
- La mantenibilidad puede resultar difícil en un entorno cambiante
- El nivel de formalidad tiene que definirse e implementarse en todo el equipo

La especificación en detalle de las condiciones de prueba puede ser especialmente efectiva en las siguientes situaciones:

- Se están empleando métodos de documentación de diseño de pruebas ligeras, tales como listas de comprobación, para adaptarse al ciclo de vida de desarrollo, las limitaciones de coste y/o tiempo y otros factores
- La base de prueba no dispone de ningún requisito formal ni de ningún otro producto de trabajo de desarrollo, o dispone de pocos
- El proyecto es un proyecto a gran escala, complejo o de alto riesgo y requiere un nivel de monitorización y control que no puede obtenerse mediante la mera relación de los casos de prueba con los productos de trabajo de desarrollo

Las condiciones de prueba pueden especificarse con menos detalle cuando la base de prueba puede relacionarse fácilmente y directamente con los productos de trabajo de diseño de pruebas. Es probable que este sea el caso en las siguientes situaciones:

- Pruebas de nivel de componentes
- Proyectos menos complejos en los que existen relaciones jerárquicas sencillas entre lo que debe probarse y cómo debe probarse
- Pruebas de aceptación en las que pueden utilizarse casos de uso para ayudar a definir las pruebas

1.4 Diseño de pruebas

El diseño de pruebas es la actividad que establece “cómo” debe probarse algo. Incluye la identificación de los casos de prueba mediante la elaboración paso por paso de las condiciones o base de prueba mediante el uso de las técnicas de pruebas previstas en la estrategia y/o en el plan de pruebas.

En función de los enfoques adoptados para la monitorización de las pruebas, el control de las pruebas y la trazabilidad, los casos de prueba pueden relacionarse directamente (o indirectamente a través de las condiciones de prueba) con la base de prueba y los objetivos definidos. Estos objetivos incluyen objetivos estratégicos, objetivos de prueba y otros criterios de éxito de las partes interesadas.

El diseño de pruebas para un nivel de pruebas dado puede llevarse a cabo una vez identificadas las condiciones de prueba y cuando se disponga de información suficiente para permitir la creación de casos de prueba de alto o bajo nivel, de conformidad con el enfoque adoptado para el diseño de pruebas. En el caso de las pruebas de nivel más alto, es más probable que el diseño de pruebas constituya una actividad aparte tras un análisis de pruebas previo. En el caso de las pruebas de nivel más bajo, es probable que el análisis y el diseño de pruebas se realicen como una actividad integrada.

Asimismo, es probable que algunas tareas que normalmente tienen lugar durante la implementación de las pruebas se integren dentro del proceso de diseño de pruebas en caso de que se adopte un enfoque iterativo para construir las pruebas a ejecutar; por ejemplo, la creación de datos de prueba. De hecho, este enfoque puede optimizar la cobertura de las condiciones de prueba, creando casos de prueba de bajo o alto nivel en el proceso.

1.5 Implementación de pruebas

La implementación de las pruebas es la actividad durante la cual los Analistas de Pruebas organizan y priorizan las pruebas. En contextos documentados formalmente, la implementación de las pruebas es la actividad en la que los diseños de las pruebas se implementan como casos de prueba, procedimientos de prueba y datos de prueba concretos. Algunas organizaciones conformes con la norma IEEE 829 [IEEE829] definen las entradas y sus resultados esperados asociados en las especificaciones de los casos de prueba y los pasos de las pruebas en las especificaciones del procedimiento de prueba. Lo habitual es que las entradas, los resultados esperados y los pasos de cada prueba se documenten juntos. La implementación de las pruebas también incluye la creación de datos de prueba almacenados (por ejemplo, en ficheros planos o tablas de bases de datos).

La implementación de las pruebas también supone la ejecución de comprobaciones finales para asegurar que el equipo de pruebas está preparado para proceder a la ejecución de las pruebas. Las comprobaciones podrían incluir la entrega del entorno de pruebas requerido, los datos de prueba y el código (posiblemente ejecutando alguna prueba de entorno de pruebas y/o de aceptación de código) y que todos los casos de prueba se hayan escrito, revisado y estén listos para ser ejecutados. Asimismo, también puede incluir la comprobación de criterios de entrada explícitos e implícitos para el nivel de prueba en cuestión (remítase a la Sección 1.7). La implementación de las pruebas también puede incluir el desarrollo de una descripción detallada del entorno de pruebas y los datos de prueba.

El nivel de detalle y la complejidad asociada al trabajo realizado durante la implementación de las pruebas puede verse afectado por el detalle de los productos de trabajo de las pruebas (por ejemplo, casos de prueba y condiciones de prueba). En algunos casos, en particular cuando las pruebas tienen que archivarse para su reutilización a largo plazo en las pruebas de regresión, estas pruebas pueden incluir descripciones detalladas de los pasos a seguir para ejecutar una prueba, con el fin de asegurar una ejecución fiable y consistente independientemente del probador que ejecute la prueba. Si existe normativa aplicable, las pruebas deberían poder demostrar que cumplen los estándares (remítase a la sección 2.9).

Durante la implementación de las pruebas, el orden en el que deben ejecutarse las pruebas manuales y automatizadas debería incluirse en un calendario de ejecución de pruebas. Los Jefes de Pruebas deberían comprobar detenidamente las limitaciones, incluidos los riesgos y las prioridades, que podrían hacer que las pruebas se ejecutaran en un orden específico o en un equipo particular. Deben conocerse y comprobarse las dependencias del entorno de pruebas o datos de prueba.

La implementación temprana de las pruebas puede tener algunos inconvenientes. Si se sigue un ciclo de vida Ágil, por ejemplo, el código puede cambiar drásticamente de una iteración a otra, haciendo que gran parte del trabajo de implementación pase a estar obsoleto. Incluso si el ciclo de vida no es tan proclive a los cambios como el de Ágil, cualquier ciclo de vida iterativo o incremental puede dar lugar a cambios importantes entre iteraciones, haciendo que las pruebas programadas pasen a no ser fiables o a estar sujetas a importantes necesidades de mantenimiento. Lo mismo aplica a los ciclos de vida secuenciales mal gestionados en los que los requisitos cambian con frecuencia, incluso en etapas tardías del proyecto. Antes de embarcarse en un gran esfuerzo de implementación de pruebas, es recomendable conocer el ciclo de vida de desarrollo de software y la previsibilidad de las características del software que estarán disponibles durante las pruebas.

La implementación temprana de las pruebas puede tener algunas ventajas. Por ejemplo, las pruebas concretas ofrecen ejemplos elaborados de cómo debería comportarse el software, si están escritas de conformidad con la base de prueba. Es probable que para los expertos de dominio del negocio sea más sencillo verificar pruebas concretas que verificar normas de negocio abstractas, por lo que pueden identificar más debilidades en las especificaciones del software. Estas pruebas verificadas pueden constituir ejemplos ilustrativos del comportamiento requerido para los diseñadores y desarrolladores de software.

1.6 Ejecución de pruebas

La ejecución de las pruebas empieza una vez entregado el objeto de prueba y una vez satisfechos los criterios de entrada para la ejecución de las pruebas. Las pruebas deberían estar diseñadas, o como mínimo definidas, antes de la ejecución de las pruebas. Debería disponerse de herramientas, en particular para la gestión de las pruebas, el seguimiento de los defectos y (si procede) la automatización de la ejecución de las pruebas. El seguimiento de los resultados de las pruebas, incluyendo el seguimiento de las métricas, debería estar operativo y todos los miembros del equipo deberían entender los datos objeto de dicho seguimiento. Los estándares para el registro de pruebas y la notificación de defectos deben estar disponibles y ser públicos. Asegurándose de que estos elementos funcionan antes de ejecutar las pruebas, la ejecución puede llevarse a cabo de una forma eficiente.

Las pruebas deberían ejecutarse conforme a los casos de prueba, si bien el Jefe de Pruebas debería considerar dejar cierto margen para que el probador pueda cubrir otros escenarios de prueba y comportamientos interesantes que puedan observarse durante las pruebas. En el caso de aplicar una estrategia de pruebas que sea, como mínimo en parte, reactiva, debe reservarse cierto tiempo para sesiones de prueba empleando técnicas basadas en la experiencia y en los defectos. Evidentemente, cualquier fallo detectado durante estas pruebas no programadas debe incluir una descripción de las variaciones del caso de prueba escrito que son necesarias para reproducir el fallo. Las pruebas automatizadas seguirán sus instrucciones definidas sin desviación.

La función principal del Jefe de Pruebas durante la ejecución de las pruebas es supervisar el progreso según el plan de pruebas y, si es preciso, emprender y llevar a cabo acciones de control para orientar las pruebas hacia una conclusión satisfactoria en términos de misión, objetivos y estrategia. Para ello, el Jefe de Pruebas puede recurrir a la trazabilidad desde los resultados de las pruebas hasta las condiciones de prueba, la base de prueba y, en última instancia, los objetivos de prueba, y también desde los objetivos de prueba hasta los resultados de prueba. Este proceso se describe en detalle en la Sección 2.6.

1.7 Evaluación de los criterios de salida y elaboración de informes

La documentación y la elaboración de informes para la monitorización del progreso y el control de las pruebas se describen más en profundidad en la Sección 2.6.

Desde el punto de vista del proceso de pruebas, es importante garantizar que se dispone de procesos efectivos para obtener la información necesaria para evaluar los criterios de salida y elaborar los informes.

La definición de los requisitos de información y los métodos de recopilación forman parte de la planificación, el seguimiento y el control de las pruebas. Durante el análisis, el diseño, la implementación y la ejecución de las pruebas, el Jefe de Pruebas debe asegurarse de que los miembros del equipo de pruebas responsable de esas actividades están comunicando la información necesaria de una manera precisa y oportuna con el fin de facilitar una efectiva evaluación y elaboración de informes.

La frecuencia y el nivel de detalle necesario para la elaboración de informes dependerán del proyecto y de la organización. Este extremo debería negociarse durante la fase de planificación de las pruebas y debería incluir la consulta con las partes interesadas relevantes del proyecto.

1.8 Actividades de cierre de pruebas

Una vez establecido que la ejecución de pruebas ha finalizado, deben registrarse las salidas clave y bien comunicarse a la persona relevante o bien archivarse. En general, estas actividades se denominan actividades de cierre de pruebas. Las actividades de cierre de pruebas pueden clasificarse en cuatro grandes grupos:

1. Comprobar la completación de las pruebas - asegurar que todo el trabajo de pruebas se ha concluido. Así por ejemplo, todas las pruebas previstas deberían ejecutarse u obviarse deliberadamente, y todos los defectos conocidos deberían arreglarse y probarse su confirmación, diferirse a una versión futura o aceptarse como restricciones permanentes.
2. Entregar los artefactos de prueba - remitir los productos de trabajo de valor a aquellos que los necesiten. Así por ejemplo, los defectos conocidos diferidos o aceptados deben comunicarse a las personas que usen y soporten el uso del sistema. Las pruebas y los entornos de pruebas deben facilitarse a los responsables de las pruebas de mantenimiento. Los conjuntos de pruebas de regresión (tanto manuales como automatizadas) deberían documentarse y entregarse al equipo de mantenimiento.
3. Lecciones aprendidas - celebrar o participar en reuniones retrospectivas con vistas a documentar las lecciones importantes (tanto del proyecto de pruebas como durante todo el ciclo de vida de desarrollo de software). En estas reuniones se establecen planes para garantizar que las buenas prácticas pueden repetirse y que las malas prácticas no se repiten o, de no poder resolverse algún problema, que aparezca en los planes del proyecto. Algunas áreas que deben tenerse en cuenta son las siguientes:
 - a. ¿Ha sido la representación de los usuarios suficientemente amplia y transversal durante las sesiones de análisis de riesgos? Por ejemplo, gracias al descubrimiento tardío de clústeres de defectos imprevistos, el equipo puede haber descubierto que, en proyectos futuros, debería haber una mayor participación transversal de representantes de usuarios en las sesiones de análisis de riesgos de la calidad.
 - b. ¿Han sido exactas las estimaciones? Por ejemplo, es posible que las estimaciones se hayan malinterpretado significativamente y, en consecuencia, las actividades de estimación futuras deberán tener este extremo en cuenta, junto con los motivos subyacentes, tales como: ¿han sido ineficientes las pruebas o son las estimaciones efectivamente inferiores a lo que deberían haber sido?
 - c. ¿Cuáles son las tendencias y los resultados del análisis de causa-efecto de los defectos? Por ejemplo, evaluar si las peticiones de cambio tardías han afectado a la calidad del análisis y desarrollo, buscar tendencias que indican malas prácticas, como por ejemplo, obviar un nivel de prueba que hubiera permitido localizar defectos antes y de una manera más rentable, obteniendo un ahorro de tiempo evidente. Comprobar si las tendencias de los defectos pueden asociarse a áreas tales como las nuevas tecnologías, los cambios de personal o la falta de aptitudes.
 - d. ¿Existen oportunidades potenciales para la mejora del proceso?
 - e. ¿Ha habido alguna variación imprevista del plan que debería preverse en planificaciones futuras?
4. Archivar resultados, registros, informes y demás documentos y productos de trabajo en el sistema de gestión de la configuración. Por ejemplo, el plan de pruebas y el plan de proyecto debería almacenarse en un archivo de planificación, con una vinculación clara al sistema y a la versión en la que se han utilizado.

Estas tareas son importantes, si bien se omiten con frecuencia, y deberían incluirse explícitamente como parte del plan de pruebas.

Es habitual omitir una o más de estas tareas, por lo general debido a la reasignación o al despido prematuro de los miembros del equipo del proyecto, a presiones de calendario o recursos en proyectos posteriores o al agotamiento del equipo. En los proyectos realizados bajo contrato, tales como los desarrollos personalizados, el contrato debería especificar las tareas necesarias.



2. Gestión de pruebas – 750 minutos

Palabras clave

plan de pruebas de nivel, plan maestro de pruebas, riesgo de producto, riesgo del proyecto, riesgo de la calidad, riesgo, análisis de riesgos, evaluación de riesgos, identificación de riesgos, nivel de riesgo, gestión de riesgos, mitigación de riesgos, pruebas basadas en riesgos, enfoque de pruebas, condición de prueba, control de pruebas, director de pruebas, estimación de pruebas, líder de pruebas, nivel de prueba, gestión de pruebas, monitorización de pruebas, plan de pruebas, política de pruebas, estrategia de pruebas, Delphi de banda ancha

Objetivos de aprendizaje de Gestión de pruebas

2.2 Gestión de pruebas en el contexto

- TM-2.2.1 (K4) Analizar las partes interesadas, las circunstancias y las necesidades de un proyecto o programa de software, incluyendo el modelo de ciclo de vida de desarrollo de software, e identificar las mejores actividades de pruebas
- TM-2.2.2 (K2) Entender cómo las actividades del ciclo de vida de desarrollo de software y los productos de trabajo afectan a las pruebas, y cómo las pruebas afectan a las actividades del ciclo de vida de desarrollo de software y a los productos de trabajo
- TM-2.2.3 (K2) Explicar las formas de gestionar los problemas de gestión de pruebas asociados a las pruebas basadas en la experiencia y a las pruebas no funcionales

2.3 Pruebas basadas en riesgos y otros enfoques para la priorización de las pruebas y la asignación del esfuerzo

- TM-2.3.1 (K2) Explicar los diferentes modos en que las pruebas basadas en riesgos responden a los riesgos
- TM-2.3.2 (K2) Explicar, mediante ejemplos, las distintas técnicas para analizar los riesgos del producto
- TM-2.3.3 (K4) Analizar, identificar y evaluar los riesgos de la calidad del producto, resumiendo los riesgos y su nivel de riesgo evaluado tomando como base las perspectivas de las partes interesadas clave del proyecto
- TM-2.3.4 (K2) Describir cómo pueden mitigarse y gestionarse los riesgos de la calidad del producto, en función de su nivel de riesgo evaluado, a lo largo del ciclo de vida y del proceso de pruebas
- TM-2.3.5 (K2) Poner ejemplos sobre las distintas alternativas para la selección de pruebas, la priorización de las pruebas y la asignación del esfuerzo

2.4 Documentación de pruebas y otros productos de trabajo

- TM-2.4.1 (K4) Analizar los modelos de políticas de pruebas y estrategias de pruebas, y crear planes maestros de pruebas, planes de nivel de prueba y otros productos de trabajo de las pruebas que estén completos y que sean coherentes con estos documentos
- TM-2.4.2 (K4) En un proyecto dado, analizar los riesgos del proyecto y seleccionar las opciones apropiadas para la gestión de riesgos (es decir, mitigación, contingencia, transferencia y/o aceptación)
- TM-2.4.3 (K2) Describir, mediante ejemplos, cómo afectan las estrategias de pruebas a las actividades de pruebas
- TM-2.4.4 (K3) Definir normas y plantillas de documentación para los productos de trabajo de las pruebas que se ajusten a las necesidades de la organización, el ciclo de vida y el proyecto, adaptando las plantillas disponibles en organismos de normalización cuando proceda

2.5 Estimación de pruebas

- TM-2.5.1 (K3) En un proyecto dado, crear una estimación para todas las actividades del proceso de pruebas, empleando todas las técnicas de estimación aplicables
- TM-2.5.2 (K2) Entender y poner ejemplos de factores que pueden influenciar las estimaciones de las pruebas

2.6 Definición y uso de métricas de prueba

- TM-2.6.1 (K2) Describir y comparar las métricas relativas a las pruebas típicas
- TM-2.6.2 (K2) Comparar las distintas dimensiones de la monitorización del progreso de las pruebas
- TM-2.6.3 (K4) Analizar y notificar los resultados de las pruebas en términos del riesgo residual, el estado de los defectos, el estado de ejecución de las pruebas, el estado de la cobertura de prueba, y la confianza para ofrecer información y recomendaciones que permitan a las partes interesadas del proyecto adoptar decisiones sobre las entregas

2.7 Valor de negocio de las pruebas

- TM-2.7.1 (K2) Presentar ejemplos (mediciones) de cada una de las cuatro categorías, determinando los "costes de calidad"
- TM-2.7.2 (K3) Estimar el valor de las pruebas tomando como base los costes de calidad, además de otras consideraciones cuantitativas y cualitativas, y comunicar el valor estimado a las partes interesadas de las pruebas

2.8 Pruebas distribuidas, externalizadas e internalizadas

- TM-2.8.1 (K2) Conocer los factores necesarios para utilizar con éxito estrategias distribuidas, externalizadas e internalizadas para la contratación del equipo de pruebas

2.9 Gestión de la aplicación de las normas del sector

- TM-2.9.1 (K2) Resumir las fuentes y los usos de las normas de pruebas de software

2.1 Introducción

En el Nivel Avanzado, el profesional de pruebas ya ha empezado a especializarse en su carrera profesional. Este capítulo se centra en las áreas de conocimiento que los profesionales de pruebas necesitan a medida que pasan a ocupar puestos de Líder de Pruebas, Jefe de Pruebas y Director de Pruebas. En este programa de estudios, nos referimos conjuntamente a estos profesionales como Jefes de Pruebas, quedando entendido que distintas organizaciones tendrán distintas definiciones para los cargos y niveles de responsabilidad de las personas que ocupan dichos puestos.

2.2 Gestionar las pruebas dentro de un contexto

Una de las principales responsabilidades de un jefe es asegurar y utilizar recursos (personas, software, hardware, infraestructura, etc.) para llevar a cabo procesos que añadan valor. En el caso de los jefes de software y TI, los procesos a menudo forman parte de un proyecto o programa destinado a entregar software o un sistema para su uso interno o externo. En el caso de los Jefes de Pruebas, los procesos son los referidos a las pruebas, en particular a las actividades del proceso de pruebas fundamentales que se describen en el Programa de Estudio de Nivel Fundamentos y en el Capítulo 1 de este programa de estudio. Dado que los procesos de pruebas añaden valor al éxito general del proyecto o programa (o evitando que se produzca un tipo de fallo más grave), el Jefe de Pruebas debe planificar y controlar los procesos de pruebas en consecuencia. En otras palabras, el Jefe de Pruebas debe organizar los procesos de pruebas adecuadamente, incluyendo las actividades y los productos de trabajo asociados, en función de las necesidades y circunstancias de las demás partes interesadas, sus actividades (por ejemplo, el ciclo de desarrollo de software en el que tienen lugar las pruebas), y sus productos de trabajo (por ejemplo, especificaciones de requisitos).

2.2.1 Comprender a las partes interesadas de las pruebas

Una persona es una parte interesada de las pruebas si tiene un interés en las actividades de pruebas, en los productos de trabajo de las pruebas o en la calidad del sistema o entregable final. El interés de la parte interesada puede suponer la participación directa o indirecta en las actividades de pruebas, la recepción directa o indirecta de los productos de trabajo de las pruebas, o el efecto directo o indirecto en la calidad de los entregables producidos por el proyecto o el programa.

Si bien las partes interesadas de las pruebas pueden variar, en función del proyecto, el producto, la organización y otros factores, pueden incluir las siguientes funciones:

- Desarrolladores, líderes de desarrollo y jefes de desarrollo. Estas partes interesadas implementan el software objeto de las pruebas, reciben los resultados de las pruebas y a menudo tienen que tomar medidas en función de esos resultados (por ejemplo, corregir los defectos detectados)
- Arquitectos de bases de datos, arquitectos de sistema y diseñadores. Estas partes interesadas diseñan el software, reciben los resultados de las pruebas y a menudo tienen que tomar medidas sobre esos resultados
- Marketing y analistas de negocio. Estas partes interesadas determinan las características, y el nivel de calidad inherente a esas características, que debe presentar el software. A menudo también participan en la definición de la cobertura de pruebas necesaria, en la revisión de los resultados de las pruebas y en la toma de decisiones en base a los resultados de las pruebas
- Jefes senior, jefes de producto y patrocinadores de proyecto. Estas partes interesadas a menudo participan en la definición de la cobertura de prueba necesaria, en la revisión de los resultados de las pruebas y en la toma de decisiones en base a los resultados de las pruebas
- Jefes de proyecto. Estas partes interesadas son responsables del éxito de sus proyectos, lo que requiere establecer un equilibrio entre la calidad, el calendario, las características y las

prioridades presupuestarias. A menudo procuran los recursos necesarios para las actividades de pruebas y colaboran con el Jefe de Pruebas en la planificación y el control de las pruebas.

- Soporte técnico, atención al cliente y personal del servicio de atención. Estas partes interesadas dan soporte a los usuarios y clientes que se benefician de las prestaciones y calidad del software entregado
- Usuarios directos e indirectos. Estas partes interesadas usan el software directamente (es decir, son usuarios finales), o reciben las salidas o los servicios producidos o soportados por el software

Para más información sobre las partes interesadas de las pruebas, remítase al Capítulo 2 de [Goucher09].

La lista de partes interesadas no es exhaustiva. Los Jefes de Pruebas deben identificar quiénes son las partes interesadas de las pruebas a que se refiere su proyecto o programa. El Jefe de Pruebas también debe conocer la naturaleza exacta de la relación que la parte interesada mantiene con las pruebas y cómo el equipo de pruebas satisface las necesidades de las partes interesadas. Además de identificar a las partes interesadas según lo descrito más arriba, el Jefe de Pruebas debería identificar cualquier otra actividad del ciclo de vida de desarrollo de software y productos de trabajo que pueda afectar a las pruebas y/o que se vea afectada por las pruebas. De lo contrario, el proceso de pruebas podría no ser plenamente efectivo ni eficiente (remítase a la Sección 2.2.3).

2.2.2 Actividades adicionales del ciclo de vida de desarrollo de software y productos de trabajo

Puesto que las pruebas de software constituyen una evaluación de la calidad de uno o varios productos de trabajo producidos fuera de las actividades de pruebas, normalmente se dan en el contexto de un conjunto más amplio de actividades del ciclo de pruebas de desarrollo de software. El Jefe de Pruebas debe planificar y orientar las actividades de pruebas teniendo en cuenta cómo estas otras actividades y sus productos de trabajo afectan a las pruebas, según lo indicado en el Programa de Estudio de Nivel Fundamento, y cómo las pruebas afectan a estas otras actividades y a sus productos de trabajo.

Así por ejemplo, en las organizaciones que utilizan prácticas de desarrollo Ágiles, los desarrolladores a menudo llevan a cabo un desarrollo guiado por pruebas, crean pruebas unitarias automatizadas e integran código de forma continua (junto con las pruebas para dicho código) en el sistema de gestión de la configuración. El Jefe de Pruebas debería colaborar con el jefe de desarrollo para asegurarse de que los probadores están integrados y se ajustan a estas actividades. Los probadores pueden revisar las pruebas unitarias para aportar sugerencias para una mayor cobertura y efectividad de estas pruebas y para obtener un conocimiento más profundo del software y de su implementación. Los probadores pueden evaluar distintas formas de integrar sus propias pruebas automatizadas, especialmente pruebas de regresión funcionales, en el sistema de gestión de la configuración. [Crispin09]

Mientras que la relación específica entre las actividades de pruebas, el resto de partes interesadas, las actividades de trabajo del ciclo de vida de desarrollo y los productos de trabajo puede variar en función del proyecto, el ciclo de vida de desarrollo de software elegido y una serie de otros factores, las pruebas están estrechamente interconectadas y relacionadas con lo siguiente:

- Gestión e ingeniería de requisitos. El Jefe de Pruebas tiene que tener en cuenta los requisitos a la hora de determinar el alcance y la estimación del esfuerzo de prueba, así como permanecer alerta ante posibles cambios en los requisitos y aplicar medidas de control en las pruebas para ajustarse a dichos cambios. Los Analistas Técnicos de Pruebas y los Analistas de Pruebas deberían participar en las revisiones de requisitos.
- Gestión del proyecto. El Jefe de Pruebas, en colaboración con los Analistas de Pruebas y los Analistas Técnicos de Pruebas, debe facilitar los requisitos de calendario y recursos al Jefe

de Proyecto. El Jefe de Pruebas debe colaborar con el Jefe de Proyecto para conocer los cambios que ha sufrido el plan del proyecto y adoptar medidas de control para ajustarse a dichos cambios.

- Gestión de la configuración, gestión de entregas y gestión de cambios. El Jefe de Pruebas, en colaboración con el equipo de pruebas, debe establecer los procesos y mecanismos de entrega de los objetos de prueba, y registrarlos en el plan de pruebas. El Jefe de Pruebas puede pedir a los Analistas de Pruebas y a los Analistas Técnicos de Pruebas que creen pruebas de verificación de la construcción y que garanticen el control de versiones durante la ejecución de las pruebas.
- Desarrollo y mantenimiento de software. El Jefe de Pruebas debe colaborar con los Jefes de Desarrollo para coordinar la entrega de los objetos de prueba, incluyendo el contenido y las fechas de cada entrega de pruebas, así como participar en la gestión de defectos (remítase al Capítulo 4).
- Soporte técnico. El Jefe de Pruebas debe colaborar con el Jefe de Soporte Técnico para garantizar la entrega adecuada de los resultados de las pruebas durante el cierre de las pruebas de manera que las personas implicadas en el soporte del producto después de la entrega conozcan los fallos y las rutas de trabajo identificados. Además, el Jefe de Pruebas debe colaborar con el Jefe de Soporte Técnico para analizar los fallos de producción con el fin de implementar mejoras al proceso de pruebas.
- Elaboración de documentación técnica. El Jefe de Pruebas debe colaborar con el Jefe de Documentación Técnica para garantizar la entrega oportuna de la documentación de pruebas, así como la gestión de los defectos previstos en dichos documentos.

Además de identificar las partes interesadas según lo descrito más arriba, el Jefe de Pruebas debe identificar el resto de actividades del ciclo de vida de desarrollo de software y productos de trabajo que afectan a las pruebas y/o que se ven afectadas por las pruebas. De lo contrario, el proceso de pruebas no será plenamente efectivo ni eficiente.

2.2.3 Alineación de las actividades de pruebas y demás actividades del ciclo de vida

Las pruebas deberían formar parte integrante del proyecto, independientemente de los modelos de desarrollo de software que se utilicen. Esto incluye:

- Modelos secuenciales, como el modelo en cascada, el modelo-V y el modelo-W. En un modelo secuencial, todos los productos de trabajo y actividades de una fase dada (tales como requisitos, diseño, implementación, pruebas unitarias, pruebas de integración, pruebas de sistema y pruebas de aceptación) se llevan a cabo antes de que empiece la siguiente fase. La planificación de pruebas, el análisis, el diseño y la implementación de las pruebas se solapan con la planificación del proyecto, el análisis de negocio/requisitos, el diseño de software y bases de datos, y la programación; la naturaleza de dicho solapamiento dependerá del nivel de prueba en cuestión. La ejecución de las pruebas se produce de manera secuencial de conformidad con los niveles de prueba descritos en el Programa de Estudio de Nivel Fundamentos y en este programa de estudio.
- Modelos iterativos o incrementales, tales como el desarrollo rápido de aplicaciones (RAD) y el proceso unificado racional (RUP). En un modelo iterativo o incremental, las características a implementar se agrupan (por ejemplo, en función de la prioridad de negocio o del riesgo), y a continuación, para cada grupo de características, se ejecutan las distintas fases del proyecto, incluyendo sus productos de trabajo y actividades. Las fases pueden realizarse secuencialmente o de manera solapada, y las propias iteraciones pueden ser secuenciales o solaparse. Durante la puesta en marcha del proyecto, la planificación de las pruebas de alto nivel y el análisis de las pruebas se realizan en paralelo a la planificación de proyecto y al análisis de negocio/requisitos. Las actividades de planificación detallada, el análisis y la implementación de las pruebas se llevan a cabo al principio de cada iteración, de una manera solapada. La ejecución de las pruebas a menudo supone el solapamiento de los niveles de

prueba. Cada nivel de prueba empieza lo antes posible y puede continuar una vez iniciados los niveles de prueba posteriores o superiores.

- **Ágil**, como SCRUM y Programación Extrema (XP). Estos son ciclos de vida iterativos en los que las iteraciones son muy cortas (a menudo de dos a cuatro semanas). Los productos de trabajo y las actividades de cada iteración finalizan antes de iniciar la siguiente iteración (es decir, las iteraciones son secuenciales). Las pruebas se realizan de la misma forma que con los modelos iterativos, pero con un mayor nivel de solapamiento entre las distintas actividades de pruebas y las actividades de desarrollo, incluyendo un importante solapamiento entre la ejecución de las pruebas (en distintos niveles) y las actividades de desarrollo. Todas las actividades incluidas en una iteración, incluyendo las actividades de pruebas, deben finalizarse antes de iniciar la siguiente iteración. En un proyecto Ágil, la función del Jefe de Pruebas normalmente pasa de una función directiva directa a una función de autoridad técnica o de consulta.
- **Espiral**. En un modelo espiral, se utilizan prototipos en una etapa temprana del proyecto para confirmar la viabilidad y experimentar con distintas decisiones de diseño e implementación, basándose en el nivel de prioridad de negocio y riesgo técnico para establecer el orden de ejecución de los experimentos de prototipo. Estos prototipos se prueban para establecer qué aspectos de los problemas técnicos siguen pendientes de resolver. Una vez resueltos los principales problemas técnicos, el proyecto prosigue de conformidad con un modelo secuencial o iterativo.

Con el fin de alinear adecuadamente las actividades de pruebas dentro del ciclo de vida, el Jefe de Pruebas debe tener un conocimiento detallado de los modelos de ciclo de vida que se utilizan en su organización. Por ejemplo, en un modelo-V, el proceso de pruebas fundamental del ISTQB aplicado al nivel de pruebas del sistema podría alinearse como sigue:

- La planificación de las pruebas de sistema concurre con la planificación del proyecto, y el control de las pruebas persiste hasta la finalización de la ejecución y el cierre de las pruebas de sistema.
- Las actividades de análisis y diseño de las pruebas de sistema concurren con la especificación de requisitos, la especificación de diseño (de alto nivel) de la arquitectura y del sistema, y la especificación de diseño (de bajo nivel) de los componentes.
- La implementación de las pruebas de sistema puede empezar durante el diseño del sistema, pero normalmente la mayor parte de estas actividades se realizan simultáneamente a la codificación y a las pruebas de componente; a veces, las actividades de implementación de las pruebas de sistema se extienden hasta pocos días antes del inicio de la ejecución de las pruebas.
- Las actividades de ejecución de las pruebas de sistema se inician cuando se cumplen todos los criterios de entrada de las pruebas de sistema (o cuando estos no son exigibles), lo que normalmente supone que, como mínimo, las pruebas de componentes y, a veces, también las pruebas de integración de componentes han concluido. La ejecución de las pruebas de sistema persistirá hasta que se cumplan los criterios de salida de las pruebas de sistema.
- La evaluación de los criterios de salida de las pruebas de sistema y la elaboración de informes sobre los resultados de las pruebas de sistema tendrán lugar a lo largo del tiempo de ejecución de las pruebas de sistema, generalmente con mayor frecuencia y urgencia a medida que se acercan los plazos límite del proyecto.
- Las actividades de cierre de las pruebas de sistema se llevan a cabo una vez cumplidos los criterios de salida de las pruebas de sistema y una vez finalizada la ejecución de las pruebas de sistema, si bien en ocasiones pueden retrasarse hasta después de finalizar las pruebas de aceptación y todas las actividades del proyecto.

En un modelo de ciclo de vida iterativo o incremental, deben realizarse las mismas tareas, si bien los tiempos y la extensión pueden variar. Por ejemplo, en lugar de ser capaz de implementar el entorno de pruebas completo al principio del proyecto, puede ser más eficiente implementar solo la parte necesaria para la iteración en curso. Con un modelo de ciclo de vida iterativo o incremental, cuanto

más tarde se produzca la planificación, más lejos podrá ampliarse el alcance del proceso de pruebas fundamental.

Además de las fases de planificación que tienen lugar para cada proyecto, la ejecución de pruebas y la elaboración de informes también pueden verse influenciadas por el ciclo de vida utilizado por el equipo. Por ejemplo, en un ciclo de vida iterativo, puede ser efectivo elaborar informes completos y llevar a cabo sesiones de revisión después de cada iteración y antes de iniciar la siguiente iteración. Al tratar cada iteración como un miniproyecto, el equipo tiene la oportunidad de corregir y ajustar en función de lo que haya ocurrido durante la iteración anterior. Dado que las iteraciones pueden ser cortas y percederas, puede tener sentido acortar el tiempo y el esfuerzo dedicados a la elaboración de informes y a la evaluación, llevándose a cabo como una forma de hacer un seguimiento del progreso general de las pruebas y para identificar cualquier área problemática lo más rápido posible. De no adoptarse medidas correctivas, los problemas de proceso acaecidos en una iteración pueden afectar fácilmente a, e incluso reproducirse en, la siguiente iteración.

La estrategia de pruebas puede contener información general sobre cómo alinear las pruebas con otras actividades del ciclo de vida (remítase a la Sección 2.4.2). El Jefe de Pruebas debería llevar a cabo la alienación específica del proyecto, para cada nivel de prueba y para cada combinación seleccionada de ciclo de vida de desarrollo de software y proceso de pruebas, durante la planificación de las pruebas y/o la planificación del proyecto.

Dependiendo de las necesidades de la organización, del proyecto y del producto, podrán requerirse niveles de prueba adicionales más allá de los definidos en el Programa de Estudio de Nivel Fundamentos, tales como:

- pruebas de integración de hardware-software
- pruebas de integración de sistema
- pruebas de interacción de prestación
- pruebas de integración de producto del cliente

Cada nivel de prueba debería disponer de los siguientes elementos claramente definidos:

- objetivos de prueba, con metas alcanzables
- alcance de la prueba y elementos de prueba
- base de prueba, junto con un medio para medir la cobertura de dicha base (es decir, trazabilidad)
- criterios de entrada y de salida
- entregables de la prueba, incluidos los informes de resultados
- técnicas de pruebas aplicables, junto con la forma empleada para garantizar el nivel adecuado de cobertura empleando dichas técnicas
- mediciones y métricas relevantes para los objetivos de prueba, los criterios de entrada y salida, y los informes sobre resultados (incluyendo las mediciones de cobertura)
- herramientas de pruebas a aplicar para tareas de prueba específicas (si procede)
- recursos (por ejemplo, entornos de pruebas)
- personas y grupos responsables, tanto dentro como fuera del equipo de pruebas
- cumplimiento con los estándares organizativos, normativos y demás normas (si procede)

Según lo indicado más adelante en este capítulo, la mejor práctica es definir estos elementos de una manera coherente en todos los niveles de prueba para evitar vacíos inútiles y peligrosos en los distintos niveles de pruebas parecidas.

2.2.4 Gestión de pruebas no funcionales

La no planificación de pruebas no funcionales puede dar lugar al descubrimiento de problemas de calidad graves, y a veces desastrosos, en un sistema después de su entrega. Sin embargo, muchos tipos de pruebas no funcionales resultan caras, por lo que el Jefe de Pruebas debe elegir qué

pruebas no funcionales realizar basándose en el riesgo y en las limitaciones. Además, existen muchos tipos diferentes de pruebas no funcionales, algunas de las cuales pueden no ser apropiadas para una aplicación determinada.

Dado que el Jefe de Pruebas puede no disponer de la pericia suficiente para gestionar todas las consideraciones asociadas a la planificación, el Jefe de Pruebas tiene que delegar parte de las responsabilidades de planificación de pruebas en los Analistas Técnicos de Pruebas (y a veces en los Analistas de Pruebas) asignados a las actividades de pruebas no funcionales. El Jefe de Pruebas debe insistir para que los analistas tengan en cuenta los siguientes factores generales:

- requisitos de las partes interesadas
- herramientas necesarias
- entorno de pruebas
- factores organizativos
- seguridad

Para más detalles, remítase al Capítulo 4 del programa de estudio de Analistas Técnicos de Pruebas de Nivel Avanzado [ISTQB ATTA SYL].

Otra consideración importante que deben tener en cuenta los Jefes de Pruebas es cómo integrar las pruebas no funcionales en el ciclo de vida de desarrollo de software. Un error común es esperar hasta finalizar todas las pruebas funcionales antes de poner en marcha las pruebas no funcionales, lo que puede dar lugar al descubrimiento tardío de defectos no funcionales críticos. En lugar de ello, las pruebas no funcionales deberían priorizarse y secuenciarse en función del riesgo. Muchas veces los riesgos no funcionales pueden mitigarse en niveles de prueba tempranos o incluso durante la fase de desarrollo. Por ejemplo, las revisiones de usabilidad de los prototipos de interfaz de usuario durante el diseño de sistema pueden ser bastante efectivas a la hora de identificar los defectos de usabilidad que darían lugar a importantes problemas de calendario de detectarse al final de las pruebas de sistema.

En los ciclos de vida iterativos, el ritmo de los cambios y de las iteraciones puede dificultar mantener el foco en determinadas pruebas no funcionales que requieren la construcción de sofisticados marcos de trabajo de pruebas. Las actividades de diseño e implementación de pruebas que superen los plazos previstos de una iteración deberían organizarse como actividades de trabajo independientes, fuera de las iteraciones.

2.2.5 Gestión de pruebas basadas en la experiencia

Si bien las pruebas basadas en la experiencia son ventajosas porque permiten encontrar defectos de una manera más eficiente que otras técnicas de pruebas, pueden obviar y sirven para comprobar la compleción de dichas técnicas, también plantean algunos retos a los Jefes de Pruebas. El Jefe de Pruebas debería conocer tanto los retos como las ventajas de las técnicas basadas en la experiencia, en particular de las pruebas exploratorias. Resulta difícil establecer la cobertura alcanzada durante estas pruebas, dado el típico registro ligero y mínima preparación previa de las pruebas. La reproducibilidad de los resultados de las pruebas requiere una atención especial de la dirección, especialmente cuando hay varios probadores implicados.

Una forma de gestionar las pruebas basadas en la experiencia, sobre todo las pruebas exploratorias, es dividir la labor de pruebas en periodos breves de 30 a 120 minutos denominados sesiones de pruebas. Esta división temporal limita y concentra el trabajo a realizar durante una sesión y proporciona un nivel de monitorización y programación. Cada sesión cubre un contrato de pruebas, que el Jefe de Pruebas comunica por escrito u oralmente al probador. El contrato de pruebas establece las condiciones de prueba a cubrir en la sesión de prueba, lo que también ayuda a mantener el foco y a evitar solapamientos si hay varias personas realizando las pruebas exploratorias a la vez.

Otra técnica para gestionar las pruebas basadas en la experiencia es integrar las pruebas autodirigidas y espontáneas en sesiones de pruebas prediseñadas y más tradicionales. Por ejemplo, los probadores pueden dar permiso (y asignar tiempo) para explorar más allá de los pasos explícitos y los resultados esperados en sus pruebas predefinidas. También pueden asignarse dichas sesiones de pruebas autodirigidas a los probadores como parte de sus pruebas diarias, antes, durante o después de un día de ejecución de pruebas predefinidas. Si dichas sesiones de pruebas identifican defectos o áreas interesantes para pruebas futuras, las pruebas predefinidas podrán actualizarse.

Al inicio de la sesión exploratoria, el probador comprueba y lleva a cabo la serie necesaria de tareas para las pruebas. Durante la sesión, el probador aprende acerca de la aplicación objeto de la prueba, diseña y ejecuta las pruebas según la técnica que se está aplicando y lo que ha aprendido de la aplicación, investiga todos los defectos y refleja los resultados de la prueba en un registro. (Si las pruebas requieren repetibilidad, los probadores también deberían registrar las entradas, las medidas y los eventos de las pruebas). Una vez finalizada la sesión, puede celebrarse una sesión informativa para establecer el sentido de sesiones posteriores.

2.3 Pruebas basadas en riesgos y otros enfoques para la priorización de las pruebas y la asignación del esfuerzo

Un reto universal de la gestión de pruebas es la correcta selección, asignación y priorización de las pruebas. Es decir, a partir de un número prácticamente infinito de condiciones de prueba y combinaciones de condiciones que podrían cubrirse, el equipo de pruebas tiene que seleccionar un conjunto de condiciones finito, determinar la cantidad adecuada de esfuerzo a asignar para cubrir cada condición con los casos de prueba, y secuenciar los casos de prueba resultantes en un orden priorizado que optimice la efectividad y eficiencia de la labor de pruebas a realizar. El Jefe de Pruebas puede servirse de la identificación y el análisis de riesgos, junto con otros factores, para resolver este problema, si bien la interacción de muchas limitaciones y variables puede requerir una solución de compromiso.

2.3.1 Pruebas basadas en riesgos

El riesgo es la posibilidad de un resultado o evento negativo o no deseado. El riesgo existe siempre que pueda surgir algún problema que perjudique las percepciones del cliente, usuario, participante o parte interesada sobre la calidad del producto o el éxito del proyecto. Cuando el efecto principal del problema potencial se refiere a la calidad del producto, los problemas potenciales se denominan riesgos de la calidad, riesgos del producto o riesgos de la calidad del producto. Cuando el efecto principal del problema potencial afecta al éxito del proyecto, los problemas potenciales se denominan riesgos del proyecto o riesgos de planificación.

En las pruebas basadas en riesgos, los riesgos de la calidad se identifican y evalúan durante un análisis de los riesgos de la calidad del producto con las partes interesadas. A continuación, el equipo de pruebas diseña, implementa y ejecuta pruebas para mitigar los riesgos de la calidad. La calidad incluye todas las prestaciones, comportamientos, características y atributos que afectan a la satisfacción del cliente, del usuario y de la parte interesada. Por lo tanto, un riesgo de la calidad es una situación potencial en la que un producto puede presentar problemas de calidad. Algunos ejemplos de riesgos de la calidad en un sistema son: cálculos incorrectos en informes (un riesgo funcional relacionado con la exactitud), respuesta lenta a las entradas de los usuarios (un riesgo no funcional relacionado con la eficiencia y el tiempo de respuesta) y dificultad para entender las pantallas y los campos (un riesgo no funcional relacionado con la usabilidad y la comprensibilidad). Si las pruebas revelan defectos, las pruebas habrán mitigado el riesgo de la calidad al revelar los defectos y brindar una oportunidad de solucionarlos antes de la entrega. Si las pruebas no revelan defectos, las pruebas habrán mitigado el riesgo de la calidad al garantizar que, bajo las condiciones probadas, el sistema funciona correctamente.

Las pruebas basadas en riesgos emplean los riesgos de la calidad del producto para seleccionar las condiciones de prueba, asignar el correspondiente esfuerzo de prueba a dichas condiciones y priorizar los casos de prueba resultantes. Existen varias técnicas de pruebas basadas en riesgos, que pueden variar mucho en función del tipo y nivel de documentación recopilada y del nivel de formalidad aplicado. Las pruebas basadas en riesgos tienen por objeto, explícita o implícitamente, utilizar las pruebas para reducir el nivel general de riesgos de la calidad y, específicamente, para reducir dicho nivel de riesgo hasta un nivel aceptable.

Las pruebas basadas en riesgos constan de cuatro actividades principales:

- Identificación de riesgos
- Evaluación de riesgos
- Mitigación de riesgos
- Gestión de riesgos

Estas actividades se solapan. En las siguientes subsecciones se estudiará cada una de estas actividades.

Para ser más efectivos, en la identificación y evaluación de los riesgos deberían participar representantes de todas las partes interesadas del proyecto y del producto, si bien en ocasiones la realidad del proyecto tiene como resultado que algunas partes interesadas actúen como sustitutas de otras partes interesadas. Así por ejemplo, en el desarrollo de software de un mercado de masas, se podría pedir a una pequeña muestra de clientes potenciales que identifique los defectos potenciales que más impacto tendrían en su uso del software; en este caso, la muestra de clientes potenciales sustituye a la base completa de clientes potenciales. Dada su experiencia específica en riesgos de la calidad del producto y fallos, los probadores deberían participar activamente en la identificación de riesgos y en el proceso de evaluación.

2.3.1.1 Identificación de riesgos

Las partes interesadas pueden identificar riesgos mediante una o varias de las siguientes técnicas:

- Entrevistas con expertos
- Evaluaciones independientes
- Uso de plantillas de riesgos
- Retrospectivas de proyecto
- Talleres de riesgos
- Lluvia de ideas
- Listas de comprobación
- Revisión de experiencias anteriores

Si en el proceso de identificación de riesgos participa una muestra lo más amplia posible de las partes interesadas habrá más posibilidades de identificar la mayor parte de los principales riesgos de la calidad del producto.

El proceso de identificación de riesgos a menudo da lugar a subproductos, es decir, identifica problemas que no constituyen riesgos de la calidad del producto. Algunos ejemplos de ello son cuestiones o problemas generales del producto o del proyecto, o problemas en documentos referenciados, tales como las especificaciones de requisitos y de diseño. En ocasiones la identificación de riesgos de la calidad puede acabar identificando riesgos del proyecto, si bien estos no constituyen el foco principal de las pruebas basadas en riesgos. No obstante, en todas las pruebas es importante gestionar los riesgos del proyecto, no solo en las pruebas basadas en riesgos. Para más detalles, remítase a la Sección 2.4.

2.3.1.2 Evaluación de riesgos

Una vez finalizada la identificación de riesgos, puede iniciarse la evaluación de riesgos, es decir, el estudio de los riesgos identificados. En particular, la evaluación de riesgos consiste en categorizar los riesgos y establecer la probabilidad y el impacto asociados a cada uno de ellos. La evaluación de riesgos también puede incluir la evaluación o la asignación de otras características a cada riesgo, tales como el propietario del riesgo.

La categorización del riesgo consiste en colocar cada riesgo en su clase correspondiente, como rendimiento, fiabilidad, funcionalidad, etc. Algunas organizaciones se basan en las características de calidad para la categorización previstas en la norma ISO 9126 [ISO9126] (sustituída por la norma ISO 25000 [ISO25000]), si bien muchas organizaciones utilizan otros esquemas de categorización. Para categorizar los riesgos a menudo se utiliza la misma lista de comprobación que la empleada para identificar los riesgos. Hay listas de comprobación genéricas de riesgos de la calidad que muchas organizaciones personalizan. Cuando se utilizan listas de comprobación como base para la identificación de riesgos, la categorización del riesgo muchas veces se produce durante el proceso de identificación.

Determinar el nivel de riesgo normalmente conlleva evaluar la probabilidad de ocurrencia y el posible impacto en caso de producirse cada riesgo. La probabilidad de ocurrencia es la probabilidad de que se de el problema potencial en el sistema sometido a prueba. En otras palabras, la probabilidad constituye una evaluación del nivel de riesgo técnico. Algunos factores que afectan a la probabilidad de los riesgos del producto y del proyecto son:

- La complejidad de la tecnología y los equipos
- Cuestiones de personal y formación entre los analistas de negocio, diseñadores y programadores
- Conflictos dentro del equipo
- Problemas contractuales con proveedores
- Equipo distribuido geográficamente
- Legado frente a nuevos enfoques
- Herramientas y tecnología
- Escaso liderazgo técnico o de dirección
- Tiempo, recursos, presupuesto y presión de la dirección
- Falta de actividades de aseguramiento de la calidad en una fase temprana
- Elevados porcentajes de cambio
- Elevados porcentajes de defectos en una fase temprana
- Problemas de interfaz e integración

El impacto en la ocurrencia es la severidad del efecto para los usuarios, clientes y demás partes interesadas. Algunos factores que afectan al impacto de los riesgos del producto y del proyecto son:

- Frecuencia de uso de la característica afectada
- Criticalidad de la prestación para alcanzar un objetivo de negocio
- Daño reputacional
- Pérdida de negocio
- Posibles pérdidas o responsabilidad financieras, medioambientales o sociales
- Sanciones legales civiles o penales
- Pérdida de licencia
- Falta de soluciones temporales razonables
- Visibilidad de fallo que genera una publicidad negativa
- Seguridad

El nivel de riesgo puede evaluarse cuantitativa o cualitativamente. Si la probabilidad y el impacto pueden verificarse cuantitativamente, los dos valores se pueden multiplicar para calcular el número de prioridad de un riesgo. Normalmente, sin embargo, el nivel de riesgo solo puede verificarse

cualitativamente. Es decir, se puede hablar de una probabilidad muy alta, alta, media, baja o muy baja, pero la probabilidad no puede expresarse como un porcentaje con una precisión real; del mismo modo, se puede hablar de un impacto muy alto, alto, medio, bajo o muy bajo, pero el impacto no puede expresarse en términos financieros de una forma completa o precisa. Esta evaluación cualitativa de los niveles de riesgo no debería considerarse inferior a los enfoques cuantitativos; en realidad, cuando las evaluaciones cuantitativas de los niveles de riesgo se usan de forma inapropiada, los resultados confunden a las partes interesadas en cuanto a la medida en que cada una entiende y puede gestionar el riesgo. Las evaluaciones cualitativas de los niveles de riesgo a menudo se combinan, mediante multiplicaciones o sumas, para establecer una puntuación global del riesgo. Esta puntuación global del riesgo podría expresarse como un número de prioridad del riesgo, si bien debería interpretarse como una calificación cualitativa y relativa en una escala ordinal.

Además, a menos que el análisis de riesgos dependa de datos de riesgo amplios y estadísticamente válidos, el análisis de riesgos dependerá de las percepciones subjetivas de las partes interesadas por lo que respecta a la probabilidad y al impacto. Normalmente los jefes de proyecto, programadores, usuarios, analistas de negocio, arquitectos y probadores tienen diferentes percepciones y en consecuencia posiblemente opinen de forma diferente sobre el nivel de riesgo de cada elemento de riesgo. El proceso de análisis de riesgos debería incluir alguna forma de llegar a un consenso o, en el peor de los casos, establecer un nivel de riesgo acordado (por ejemplo, por órdenes de la dirección o tomando la media, la mediana o el nivel modelo para el elemento de riesgo). Asimismo, deberían comprobarse los niveles de riesgo y su correcta distribución en todo el rango para asegurar que las calificaciones de riesgo ofrecen una orientación útil en términos de secuenciación, priorización y asignación del esfuerzo de las pruebas. De lo contrario, los niveles de riesgo no podrán utilizarse como guía para las actividades de mitigación de riesgos.

2.3.1.3 Mitigación de riesgos

Las pruebas basadas en riesgos empiezan con un análisis de los riesgos de la calidad (para identificar y evaluar los riesgos de la calidad del producto). Este análisis es la base del plan maestro de pruebas y otros planes de pruebas. Según lo especificado en los planes, las pruebas se diseñan, implementan y ejecutan en orden para cubrir los riesgos. El esfuerzo asociado al desarrollo y a la ejecución de una prueba es proporcional al nivel de riesgo, lo que significa que para los riesgos más altos se utilizan técnicas de pruebas más meticulosas (como las pruebas por pares), mientras que para los riesgos más bajos se emplean técnicas de pruebas menos meticulosas (como las particiones de equivalencia o las pruebas exploratorias con periodos de tiempo establecidos). Además, la prioridad del desarrollo y la ejecución de la prueba dependerán del nivel de riesgo. Algunas normas de seguridad relacionadas (por ejemplo, FAA DO-178B/ED 12B, IEC 61508), prescriben las técnicas de pruebas y el grado de cobertura basado en el nivel de riesgo. Asimismo, el nivel de riesgo debería influenciar las decisiones sobre, por ejemplo, el uso de revisiones de productos de trabajo de proyecto (incluyendo pruebas), el nivel de independencia, el nivel de experiencia del probador y el grado de las pruebas de confirmación y pruebas de regresión ejecutadas.

A lo largo del proyecto, el equipo de pruebas debe ser consciente de toda la información adicional que pueda alterar el conjunto de riesgos de la calidad y/o el nivel de riesgo asociado a los riesgos de la calidad conocidos. Debería realizarse un ajuste periódico del análisis de los riesgos de la calidad, y los ajustes a las pruebas derivados. Estos ajustes deberían realizarse como mínimo en los principales hitos del proyecto. Los ajustes incluyen la identificación de nuevos riesgos, una nueva evaluación del nivel de riesgo existente y la evaluación de la efectividad de las actividades de mitigación de riesgos. A modo de ejemplo, si tuviera lugar una sesión de identificación y análisis de riesgos tomando como base la especificación de requisitos durante la fase de requisitos, una vez finalizada la especificación del diseño, los riesgos deberían volver a evaluarse. Por poner otro ejemplo, si durante las pruebas se considera que un componente contiene un número de defectos superior al esperado, se puede concluir que la probabilidad de los defectos en esta área es superior al esperado, y de esta forma ajustar al alza la probabilidad y el nivel de riesgo global. Esto podría dar lugar a un aumento de la cantidad de pruebas a realizar en este componente.

Los riesgos de la calidad del producto también pueden mitigarse antes de empezar a ejecutar las pruebas. Por ejemplo, si se localizan problemas con los requisitos durante la identificación de riesgos, el equipo del proyecto puede llevar a cabo revisiones exhaustivas de la especificación de requisitos como medida de mitigación. Esto puede reducir el nivel de riesgo, lo que puede suponer que se requieren menos pruebas para mitigar los demás riesgos de la calidad.

2.3.1.4 Gestión de riesgos en el ciclo de vida

Lo ideal es que la gestión de riesgos se lleve a cabo durante todo el ciclo de vida. Si una organización cuenta con un documento de política de pruebas y/o un documento de estrategia de pruebas, estos deben describir el proceso general con el que se gestionan los riesgos del producto y proyecto en las pruebas, y muestren cómo esa gestión del riesgo se integra y afecta a todas las fases de las pruebas.

En una organización madura en la que el conocimiento de riesgos predomina en el equipo de pruebas, la gestión de riesgos se lleva a cabo en muchos niveles, y no solo en las pruebas. Los riesgos importantes no solo se abordan antes en niveles de prueba particulares, sino también en niveles de prueba más tempranos. Por ejemplo, si se identifica que el rendimiento constituye un área de riesgo de la calidad clave, no solo se iniciarán antes las pruebas de rendimiento en el sistema, sino que también se ejecutarán pruebas de rendimiento durante las pruebas unitarias y de integración. Las organizaciones maduras no solo identifican riesgos, sino también el origen de los riesgos y la consecuencia de que esos riesgos se conviertan en resultados. En el caso de aquellos defectos que sí se producen, se emplea el análisis de causas raíz para conocer el origen del riesgo más en profundidad y para implementar mejoras de proceso que eviten la aparición de defectos. La mitigación se produce a lo largo del ciclo de vida de desarrollo de software. El análisis de riesgos es plenamente informado y tiene en cuenta la actividad de trabajo asociada, el análisis del comportamiento del sistema, la evaluación de riesgos basada en los costes, el análisis de los riesgos del producto, el análisis de los riesgos de usuario final y el análisis de los riesgos de responsabilidad. El análisis de riesgo trasciende las pruebas, y el equipo de pruebas participa e influencia un análisis de riesgo que abarca todo el programa.

Casi todos los métodos de pruebas basadas en riesgos también incluyen técnicas que permiten emplear el nivel de riesgo para secuenciar y priorizar las pruebas, garantizando así una cobertura temprana de las áreas más importantes y el descubrimiento de los defectos más relevantes durante la ejecución de las pruebas. En algunos casos, todas las pruebas con mayor riesgo se ejecutan antes de las pruebas de menor riesgo, y las pruebas se ejecutan en estricto orden de riesgo (a menudo denominado "profundidad primero"¹); en otros casos, se utiliza un enfoque de muestreo para seleccionar una muestra de pruebas de entre todos los elementos de riesgo identificados, utilizando el riesgo para ponderar la selección al tiempo que se garantiza la cobertura de cada elemento de riesgo al menos una vez (a menudo denominado "amplitud primero"²).

Tanto si las pruebas basadas en riesgos se realizan en profundidad primero o en anchura primero, es posible que el tiempo asignado para las pruebas se consuma sin que se hayan ejecutado todas las pruebas. Las pruebas basadas en riesgos permiten a los probadores informar a la dirección sobre el nivel restante de riesgo en un punto dado, y permite a la dirección decidir sobre si ampliar las pruebas o transferir el riesgo restante a los usuarios, clientes, atención al cliente/soporte técnico y/o personal operativo.

Durante la ejecución de las pruebas, las técnicas de pruebas basadas en riesgos más sofisticadas - que no tienen por qué ser las más formales o pesadas- permiten a los participantes del proyecto, a los jefes de proyecto y producto, a los ejecutivos, a los directivos y a las partes interesadas del proyecto monitorizar y controlar el ciclo de vida de desarrollo de software, además de adoptar decisiones relativas a la entrega basándose en el nivel residual de riesgo. Para ello es necesario que

¹ profundidad primero es la traducción del término "depth-first". Otra traducción podría ser "primero en profundidad"

² amplitud primero es la traducción del término "breadth-first". Otra traducción podría ser "primero en amplitud".

el Jefe de Pruebas informe de los resultados de las pruebas de forma que las partes interesadas de las pruebas puedan entender.

2.3.2 Técnicas de pruebas basadas en riesgos

Existen varias técnicas de pruebas basadas en riesgos. Algunas de estas técnicas son muy informales. Algunos ejemplos incluyen enfoques en los que el probador analiza los riesgos de la calidad durante las pruebas exploratorias [Whittaker09]. Si bien esto puede ayudar a orientar las pruebas, también puede dar lugar a centrarse excesivamente en la probabilidad de los defectos en lugar de en su impacto. Además, no tiene en cuenta las entradas de las partes interesadas transversales. Asimismo, estos enfoques son subjetivos y dependen de la destreza, la experiencia y las preferencias del probador individual. Como tales, estos enfoques raramente logran todas las ventajas de las pruebas basadas en riesgos.

En un intento por beneficiarse de las ventajas de las pruebas basadas en riesgos minimizando los costes, son muchos los que adoptan enfoques ligeros basados en riesgos. Estos enfoques combinan la capacidad de respuesta y la flexibilidad de los enfoques informales y la potencia y la creación de consenso de los enfoques más formales. Algunos ejemplos de enfoques ligeros son: el análisis y la gestión de riesgos pragmáticos (PRAM) [Black09], las pruebas de software sistemáticas (SST) [Craig02], y la gestión de riesgos del producto (PRisMa) [vanVeenendaal12]. Además de los atributos habituales de las pruebas basadas en riesgos, estas técnicas generalmente tienen los siguientes atributos:

- Mejoradas a lo largo del tiempo tomando como base la experiencia del sector en las pruebas basadas en riesgos, especialmente en aquellos sectores en los que las cuestiones de eficiencia son importantes
- Basadas en la amplia participación de un equipo transversal de partes interesadas, que representan tanto las perspectivas del negocio como las perspectivas técnicas, durante la identificación y evaluación inicial de los riesgos
- Optimizadas si se introducen en las primeras fases del proyecto, cuando las opciones para mitigar los riesgos de la calidad son mayores y los principales productos de trabajo y productos derivados del análisis de riesgos pueden ayudar a influenciar la especificación e implementación del producto con vistas a minimizar el riesgo
- Utilizan las salidas generadas (matriz de riesgos o tabla de análisis de riesgos) como base para el plan de pruebas y las condiciones de prueba, además de para el resto de actividades de gestión y análisis consiguientes
- Soportan la elaboración de informes de resultados de las pruebas en términos de riesgo residual en todos los niveles de las partes interesadas de las pruebas

Algunas de estas técnicas (por ejemplo, SST) requieren especificaciones de requisitos como una entrada del análisis de riesgos y no pueden utilizarse a menos que se faciliten especificaciones de requisitos. Otras técnicas (como PRisMa y PRAM) fomentan el uso de una estrategia combinada basada en riesgos y en requisitos, y utilizan como entrada para el análisis de riesgos los requisitos y/u otras especificaciones, pero pueden funcionar íntegramente basándose en las entradas de las partes interesadas. El uso de requisitos como entrada ayuda a garantizar una buena cobertura de los requisitos, pero el Jefe de Pruebas debe asegurarse de que no se omiten riesgos importantes no sugeridos por los requisitos - especialmente en áreas no funcionales. Cuando se establecen requisitos correctos y priorizados como entrada, normalmente existe una fuerte correlación entre los niveles de riesgo y la prioridad de los requisitos.

Muchas de estas técnicas también fomentan el uso del proceso de identificación y evaluación de riesgos como un modo de generar consenso entre las partes interesadas por lo que respecta al enfoque de las pruebas. Esto constituye una ventaja importante, pero requiere que las partes interesadas dediquen tiempo a participar en sesiones grupales de lluvia de ideas o entrevistas personalizadas. Una participación insuficiente de las partes interesadas tendrá como resultado un

análisis de riesgos deficiente. Por supuesto, las partes interesadas no siempre están de acuerdo en el nivel de riesgo, por lo que la persona que lidere el esfuerzo de análisis de los riesgos de la calidad debe trabajar de manera creativa y positiva con las partes interesadas para lograr el máximo nivel de acuerdo posible. Todas las aptitudes del moderador de reuniones de revisión son aplicables al líder de un análisis de los riesgos de la calidad.

Al igual que las técnicas más formales, las técnicas ligeras permiten el uso de la ponderación de los factores de la probabilidad y del impacto para enfatizar los riesgos de negocio o los riesgos técnicos (dependiendo del nivel de pruebas, por ejemplo). Al contrario que las técnicas más formales, no obstante, las técnicas ligeras:

- emplean solo dos factores: probabilidad e impacto, y
- emplean juicios y escalas sencillos y cualitativos.

El carácter ligero de estos enfoques aporta flexibilidad, aplicabilidad en una serie de dominios y accesibilidad en equipos de todos los niveles de experiencia y aptitudes (incluso con personas no técnicas y junior).

En el extremo pesado de la escala formal, el Jefe de Pruebas tiene una serie de opciones a su alcance:

- El análisis de peligros, que amplía el proceso analítico ascendente, intentando identificar los peligros que subyacen en cada riesgo.
- El coste de exposición, donde el proceso de evaluación del riesgo supone determinar, para cada elemento de riesgo de la calidad, tres factores: 1) la probabilidad (expresada como porcentaje) de un fallo asociado al elemento de riesgo; 2) el coste de la pérdida (expresado como cantidad financiera) asociada a un fallo típico relacionado con el elemento de riesgo, de producirse en producción; y 3) el coste de probar dichos fallos.
- El Análisis de Modos de Fallo y Efectos en el Software (AMFE) y sus variantes [Stamatis03], donde se identifican los riesgos de la calidad, sus causas potenciales y sus efectos probables para después asignar calificaciones de severidad, prioridad y detección.
- El Despliegue de la Función de Calidad (DFC), que es una técnica de gestión de riesgos de la calidad, con implicaciones en las pruebas, que se ocupa, específicamente, de los riesgos de la calidad derivados de un conocimiento incorrecto o insuficiente de los requisitos del cliente o del usuario.
- El Análisis de Árbol de Faltas (AAF), donde varios fallos observados (en las pruebas o en producción), o fallos potenciales (riesgos de la calidad), son sometidos a un análisis de causas raíz partiendo de los defectos que podrían causar el fallo, siguiendo con los errores o defectos que podrían causar dichos defectos, y prosiguiendo hasta identificar las distintas causas raíz.

Las técnicas específicas que deberían utilizarse en las pruebas basadas en riesgos, y el nivel de formalidad de cada técnica, dependen de consideraciones del proyecto, del proceso y del producto. Así por ejemplo, un enfoque informal, como la técnica exploratoria Whittaker's, podría aplicarse a un camino o arreglo rápido. En los proyectos Ágiles, el análisis de los riesgos de la calidad está totalmente integrado en el periodo inicial de cada sprint y la documentación sobre riesgos está integrada en el seguimiento de las historias de usuario. Los sistemas de sistemas requieren un análisis de riesgos de cada sistema, así como del sistema general de sistemas. Los proyectos críticos para la seguridad y los proyectos de misión crítica requieren niveles más altos de formalidad y documentación.

Las entradas, los procesos y las salidas de las pruebas basadas en riesgos tenderán a fijarse en virtud de la técnica seleccionada. Entre las entradas comunes se encuentran perspectivas de las partes interesadas, especificaciones y datos históricos. Entre los procesos comunes se encuentran la identificación, la evaluación y el control de riesgos. Entre las salidas comunes se encuentran una lista de riesgos de la calidad (con el nivel de riesgo asociado y la asignación del esfuerzo de prueba recomendado), los defectos descubiertos en los documentos de entrada, tales como especificaciones,

cuestiones o problemas relacionados con los elementos de riesgo, y riesgos del proyecto que afectan a las pruebas o al proyecto en general.

Normalmente, el factor de éxito más crítico de las pruebas basadas en riesgos es la participación del equipo correcto de partes interesadas en las fases de identificación y evaluación de riesgos. Todas las partes interesadas tienen su propia visión de qué constituye la calidad del producto y su propia serie de prioridades e inquietudes al respecto. No obstante, las partes interesadas tienden a incluirse en dos categorías: partes interesadas del negocio y partes interesadas técnicas.

Las partes interesadas del negocio incluyen, entre otros, a clientes, usuarios, personal de operaciones y personal del servicio de asistencia (help desk) y soporte técnico. Estas partes interesadas conocen al cliente, y por lo tanto pueden identificar los riesgos y evaluar su impacto desde un punto de vista de negocio.

Entre las partes interesadas técnicas se incluyen los desarrolladores, arquitectos, administradores de bases de datos y administradores de redes, entre otros. Estas partes interesadas conocen las formas subyacentes en la que el software puede fallar, y por lo tanto pueden identificar los riesgos y evaluar su probabilidad desde un punto de vista técnico.

Algunas partes interesadas disponen de ambos puntos de vista, el de negocio y el técnico. Así por ejemplo, los expertos en la materia que desempeñan funciones de pruebas o análisis de negocio a menudo tienen una perspectiva más amplia de los riesgos gracias a su experiencia técnica y de negocio.

El proceso de identificación de los elementos de riesgo genera una lista de riesgos importante. No es necesario que las partes interesadas discutan sobre los elementos de riesgo; siempre que una parte interesada perciba algo como un riesgo para la calidad del sistema, se considerará un elemento de riesgo. No obstante, es importante que las partes interesadas lleguen a un consenso por lo que respecta a las calificaciones de los niveles de riesgo. Por ejemplo, en enfoques ligeros que se sirven de la probabilidad y el impacto como factores de clasificación, parte del proceso debe incluir el establecimiento de un esquema de clasificación común para los dos factores, probabilidad e impacto. Todas las partes interesadas, incluido el grupo de pruebas, deben aplicar esta misma escala y deben ser capaces de converger en una única clasificación de probabilidad e impacto para cada elemento de riesgo de la calidad.

Si van a utilizarse pruebas basadas en riesgos a largo plazo, el Jefe de Pruebas deberá defender con éxito e iniciar pruebas basadas en riesgos con las partes interesadas. El grupo transversal debe reconocer el valor del análisis de riesgos para garantizar el uso continuado de la técnica. Esto exige que el Jefe de Pruebas entienda las necesidades, las expectativas y la disponibilidad de tiempo de las partes interesadas para participar en el proceso.

Un alto nivel de compromiso de las partes interesadas en el proceso de análisis de los riesgos de la calidad supone una ventaja importante para el Jefe de Pruebas. Esta ventaja es que, en proyectos infraespecificados con requisitos débiles o inexistentes, las partes interesadas pueden seguir identificando riesgos si se guían por la lista de comprobación adecuada. Esta ventaja se hace patente cuando, una vez finalizada la implementación de pruebas basadas en riesgos, mejora la efectividad de la detección de defectos del equipo de pruebas. Esto sucede porque se está utilizando una base de prueba más completa -en este caso, la lista de elementos de riesgos de la calidad.

Durante el cierre de las pruebas basadas en riesgos, los equipos de prueba deben medir en qué medida se han beneficiado de las ventajas. En muchos casos, esto supone responder a algunas o a todas las siguientes cuatro preguntas mediante métricas y consultas al equipo:

- ¿Detectó el equipo de pruebas un porcentaje de defectos importantes mayor que de defectos menos importantes?

- ¿Encontró el equipo de pruebas la mayoría de los defectos importantes en una etapa temprana del periodo de ejecución de las pruebas?
- ¿Fue capaz el equipo de pruebas de explicar los resultados de las pruebas a las partes interesadas en términos de riesgo?
- ¿Tenían las pruebas omitidas por el equipo (en caso de existir) un nivel más bajo de riesgo que las pruebas ejecutadas?

En la mayoría de los casos, los resultados de las pruebas basadas en riesgos correctas dan lugar a una respuesta afirmativa para las cuatro preguntas. A largo plazo, el Jefe de Pruebas debería fijar los objetivos de mejora del proceso para estas métricas, además de intentar mejorar la eficiencia del proceso de análisis de los riesgos de la calidad. Por supuesto, las pruebas basadas en riesgos pueden emplear otras métricas y criterios de éxito, y el Jefe de Pruebas debería estudiar detenidamente la relación entre esas métricas, los objetivos estratégicos que persigue el equipo de pruebas y los comportamientos que se derivarán de la gestión basándose en un conjunto específico de métricas y criterios de éxito.

2.3.3 Otras técnicas para la selección de pruebas

Mientras que muchos Jefes de Pruebas emplean las pruebas basadas en riesgos como un elemento de su estrategia de pruebas, muchos también aplican otras técnicas.

Una de las técnicas alternativas más prominente para desarrollar y priorizar las condiciones de prueba son las pruebas basadas en requisitos. Las pruebas basadas en requisitos pueden utilizar varias técnicas, tales como revisiones de la ambigüedad, análisis de las condiciones de prueba y gráficos causa-efecto. Las revisiones de la ambigüedad identifican y eliminan las ambigüedades de los requisitos (que hacen las veces de base de prueba), a menudo mediante el uso de una lista de comprobación de defectos de requisitos comunes (remítase a [Wiegers03]).

Según lo descrito en [Craig02], el análisis de las condiciones de prueba implica un estudio minucioso de la especificación de requisitos para identificar qué condiciones de prueba deben cubrirse. Si esos requisitos tienen una prioridad asignada, esta puede utilizarse para asignar el esfuerzo y priorizar los casos de prueba. De no asignarse esta prioridad, resultará difícil determinar el esfuerzo necesario y el orden de las pruebas sin combinar las pruebas basadas en requisitos con las pruebas basadas en riesgos.

En el módulo de Analistas de Pruebas de Nivel Avanzado se tratan los gráficos causa-efecto en el contexto de cubrir combinaciones de prueba como parte del análisis de las pruebas. Sin embargo, su uso es más amplio en el sentido de que permiten reducir un problema de pruebas extremadamente voluminoso a un número manejable de casos de prueba y seguir ofreciendo una cobertura funcional del 100 % de la base de prueba. Los gráficos causa-efecto también identifican vacíos en la base de prueba durante el diseño del caso de prueba, al permitir identificar defectos en una etapa temprana del ciclo de vida de desarrollo cuando se empiezan a diseñar las pruebas contra los requisitos provisionales. Un importante impedimento a la adopción de gráficos causa-efecto es la complejidad que supone crear estos gráficos. Hay herramientas que soportan este método que puede ser complicado realizar manualmente.

Un obstáculo habitual de las pruebas basadas en requisitos es que a menudo las especificaciones de requisitos son ambiguas, inestables, incompletas o no existentes. No todas las organizaciones están motivadas para resolver estos problemas, por lo que los probadores que se enfrenten a estas situaciones deberán seleccionar otra estrategia de pruebas.

Otro método que a veces se utiliza para aumentar el uso de los requisitos existentes es la creación de perfiles de uso o perfiles operativos, un enfoque basado en modelos, que utiliza un mix de casos de uso, usuarios (a veces denominados personas), entradas y salidas para reflejar con precisión el uso

real del sistema. Esto permite probar no solo la funcionalidad sino también la usabilidad, la interoperabilidad, la fiabilidad, la seguridad y el rendimiento.

Durante el análisis y la planificación de las pruebas, el equipo de pruebas identifica los perfiles de uso e intenta cubrirlos con los casos de prueba. El perfil de uso es una estimación, basada en la información disponible, de un uso realista del software. Esto quiere decir que, al igual que sucede con las pruebas basadas en riesgos, es posible que los perfiles de uso no modelen perfectamente la eventual realidad. No obstante, si se dispone de suficiente información y entradas de las partes interesadas, el modelo será adecuado (remítase a [Musa04]).

Algunos Jefes de Pruebas también aplican enfoques metódicos tales como listas de comprobación para determinar qué, cuánto y en qué orden probar. En el caso de productos que son muy estables, puede bastar con una lista de comprobación de las principales áreas funcionales y no funcionales, sumada a un repositorio de casos de prueba existentes. La lista de comprobación proporciona información heurística para la asignación del esfuerzo y la secuenciación de las pruebas, que normalmente depende del tipo y la cantidad de cambios que se hayan producido. Estos enfoques tienden a ser menos válidos si se utilizan para probar más que solo cambios menores.

Por último, otro método común es adoptar un enfoque reactivo. En un enfoque reactivo, antes de la ejecución de las pruebas se llevan a cabo muy pocas tareas de análisis, diseño o implementación de pruebas. El equipo de pruebas se concentra en reaccionar ante el producto según este es entregado. Los clústeres de bugs, al ser descubiertos, pasan a ser el foco de las pruebas posteriores. Las tareas de priorización y asignación son totalmente dinámicas. El enfoque reactivo puede funcionar como complemento a otros enfoques, pero si se emplea de manera exclusiva, este enfoque tiende a omitir áreas importantes de la aplicación que son relevantes pero que no tienen un gran número de bugs.

2.3.4 Priorización de las pruebas y asignación del esfuerzo en el proceso de pruebas

Independientemente de la técnica -o, incluso mejor, combinación de técnicas- que el Jefe de Pruebas utilice, el Jefe de Pruebas debe incorporar dicha técnica al proyecto y a los procesos de pruebas. Así por ejemplo, en los ciclos de vida secuenciales (por ejemplo, modelo-V), el equipo de pruebas selecciona las pruebas, asigna el esfuerzo de prueba y prioriza las pruebas inicialmente durante la fase de requisitos, con ajustes periódicos, mientras que en los ciclos de vida iterativos o Ágiles se requiere un enfoque de iteración por iteración. La planificación y el control de pruebas deben tener en cuenta el grado en el que evolucionarán los riesgos, los requisitos y/o los perfiles de uso, y responder en consecuencia.

Durante las fases de análisis, diseño e implementación de pruebas, deben aplicarse la asignación y la priorización fijadas durante la etapa de planificación de las pruebas. El proceso de pruebas con frecuencia incluye una fase de estricto análisis y/o modelado, para que esa información no se utilice para guiar el proceso de pruebas en el futuro. Esta etapa normalmente tiene lugar durante el diseño y la implementación.

Durante la ejecución de las pruebas, debe llevarse a cabo la priorización establecida durante la planificación de pruebas, si bien es importante actualizar la priorización periódicamente tomando como base la información obtenida después de la redacción inicial del plan. A la hora de evaluar y reportar los resultados de las pruebas y el estado de los criterios de salida, el Jefe de Pruebas también debe evaluar y reportar los riesgos, los requisitos, los perfiles de uso, las listas de comprobación y otras guías empleadas para seleccionar y priorizar pruebas. En caso necesario, debería realizarse un seguimiento de las pruebas basado en el esquema de priorización de las pruebas.

Como parte de los resultados reportados y la evaluación de los criterios de salida, el Jefe de Pruebas puede medir el nivel de compleción de las pruebas. Esto debería incluir la localización de los casos de prueba y los defectos descubiertos hasta la base de prueba relevante. Por ejemplo, en las pruebas

basadas en riesgos, se realizan todas las pruebas y se detectan defectos, los probadores pueden examinar el nivel de riesgo residual restante. Esto apoya el uso de pruebas basadas en riesgos al determinar el momento adecuado para entregar. La información sobre pruebas debería abordar los riesgos cubiertos y aún abiertos, así como los beneficios logrados y no logrados aún. Como ejemplo de comunicación de resultados de pruebas basadas en cobertura de riesgos, ver [Black03].

Por último, durante el cierre de las pruebas, el Jefe de Pruebas debería evaluar las métricas y los criterios de éxito que son pertinentes para las necesidades y expectativas de las partes interesadas de las pruebas, incluyendo las necesidades y expectativas de los clientes y usuarios por lo que respecta a la calidad. Para que un equipo de pruebas sea realmente efectivo, las pruebas tienen que satisfacer estas necesidades y expectativas.

2.4 Documentación de pruebas y otros productos de trabajo

La documentación a veces se genera como parte de las actividades de gestión de pruebas. Si bien la nomenclatura específica de los documentos de gestión de pruebas y el alcance de cada documento tienden a variar. A continuación, se indican algunos documentos de gestión de pruebas comúnmente encontrados en las organizaciones y en los proyectos:

- Política de pruebas - describe los objetivos y las metas de pruebas de la organización
- Estrategia de pruebas - describe los métodos de prueba generales e independientes del proyecto de la organización
- Plan maestro de pruebas (o plan de pruebas de proyecto) - describe la implementación de la estrategia de pruebas para un proyecto específico
- Plan de pruebas de nivel (o plan de pruebas de fase) - describe las actividades específicas a realizar dentro de cada nivel de prueba

La organización física de estos tipos de documentos puede variar dependiendo del contexto. En algunas organizaciones y en algunos proyectos, pueden combinarse en un único documento; en otras, pueden encontrarse en documentos independientes; y en algunas otras, los contenidos de estos tipos de documentos pueden ser intuitivos, no escritos o constituir metodologías de prueba tradicionales. Las organizaciones y los proyectos más grandes y más formales suelen tener todos estos tipos de documentos en forma de productos de trabajo por escrito, mientras que las organizaciones y los proyectos más pequeños y menos formales suelen tener menos de estos productos de trabajo por escrito. Este programa de estudio describe cada uno de estos tipos de documentos por separado, si bien en la práctica el contexto organizativo y del proyecto determina la utilización correcta de cada tipo.

2.4.1 Política de pruebas

La política de pruebas describe por qué la organización realiza las pruebas. Define los objetivos generales de las pruebas que la organización quiere lograr. Esta política deberían desarrollarla miembros senior de la dirección de pruebas de la organización en colaboración con directivos senior de los grupos de partes interesadas de las pruebas.

En algunos casos, la política de pruebas será complementaria o formará parte de una política de calidad más amplia. Esta política de calidad describe los valores y objetivos generales de la gestión relacionados con la calidad.

Si existe una política de pruebas por escrito, esta podría ser un documento breve, de alto nivel que:

- resume el valor que la organización obtiene de las pruebas
- define los objetivos de las pruebas, tales como crear confianza en el software, detectar defectos en el software y reducir el nivel de riesgo de la calidad (remítase a la Sección 2.3.1)
- describe cómo evaluar la efectividad y la eficiencia de las pruebas a la hora de cumplir estos objetivos

- establece el proceso de pruebas típico, quizás basándose en el proceso de pruebas básico del ISTQB
- especifica cómo la organización mejorará sus procesos de prueba (remítase al Capítulo 5)

La política de pruebas debería abordar las actividades de pruebas tanto para el nuevo desarrollo como para el mantenimiento. También puede hacer referencia a estándares internos y/o externos para los productos de trabajo de las pruebas y la terminología a utilizar en la organización.

2.4.2 Estrategia de pruebas

La estrategia de pruebas describe la metodología de pruebas general de la organización. En ella se incluye la forma en la que se utilizan las pruebas para gestionar riesgos del producto y del proyecto, la división de las pruebas en niveles, y las actividades de alto nivel asociadas a las pruebas. (Una misma organización puede tener estrategias distintas para situaciones diferentes, tales como distintos ciclos de vida de desarrollo de software, distintos niveles de riesgo o distintos requisitos normativos). La estrategia de pruebas, así como los procesos y las actividades descritos en la misma, debería ser coherente con la política de pruebas. Debería facilitar los criterios genéricos de entrada y salida de las pruebas para la organización o para uno o varios programas.

Según lo indicado más arriba, las estrategias de prueba describen las metodologías de prueba generales, que normalmente incluyen:

- Estrategias analíticas, tales como pruebas basadas en riesgos, en las que el equipo de pruebas analiza la base de las pruebas para identificar las condiciones de prueba que deben cubrirse. Así por ejemplo, en las pruebas basadas en requisitos, el análisis de las pruebas establece condiciones de prueba a partir de los requisitos, a continuación se diseñan e implementan las pruebas para cubrir dichas condiciones. Posteriormente las pruebas se ejecutan, a menudo aplicando la prioridad del requisito cubierto por cada prueba para establecer el orden de ejecución de las pruebas. Los resultados de las pruebas se reportan en términos de estado de los requisitos, por ejemplo, requisito probado y pasado, requisito probado y fallido, requisito pendiente de probar íntegramente, requisito de prueba bloqueado, etc.
- Las estrategias basadas en modelos, tales como los perfiles operativos, en las que el equipo de pruebas desarrolla un modelo (basado en situaciones reales o previstas) del entorno en el que existe el sistema, las entradas y las condiciones a las que está sujeto el sistema y cómo debería comportarse el sistema. Por ejemplo, en las pruebas de rendimiento basadas en modelos de una aplicación de dispositivo móvil de rápido crecimiento, se podrían desarrollar modelos de tráfico de red entrante y saliente, de usuarios activos e inactivos, y de la carga de procesos resultante, tomando como base el uso actual y el crecimiento del proyecto a lo largo del tiempo. Asimismo, podrían desarrollarse modelos teniendo en cuenta el hardware, el software, la capacidad de datos, la red y la infraestructura del entorno de producción actual. También podrían desarrollarse modelos para los valores ideales, esperados y mínimos de las tasas de transferencia, los tiempos de respuesta y la asignación de recursos.
- Estrategias metódicas, como las basadas en características de calidad, en las que el equipo de pruebas emplea un conjunto predeterminado de condiciones de prueba, como por ejemplo una norma de calidad (por ejemplo, la ISO 25000 [ISO2500] que sustituye a la ISO 9126 [ISO9126]), una lista de comprobación o una recopilación de condiciones de prueba generales y lógicas que puede referirse a un dominio, a una aplicación o a un tipo de prueba en particular (por ejemplo, pruebas de seguridad), y emplea ese conjunto de condiciones de prueba de una iteración a la siguiente o de una entrega a la siguiente. Por ejemplo, en las pruebas de mantenimiento de una página web sencilla y estable de comercio electrónico, los probadores podrían emplear una lista de comprobación que identifique las funciones clave, los atributos y los vínculos de cada página. Los probadores deben cubrir los elementos relevantes de esta lista de comprobación cada vez que se realice una modificación en la página.

- Estrategias de cumplimiento de procesos o normas, tales como los sistemas médicos sujetos a las normas de la *Food and Drug Administration* de Estados Unidos (Agencia de Alimentos y Medicamentos de los Estados Unidos o FDA), según las cuales el equipo de pruebas se ajusta a una serie de procesos establecidos por un comité de normalización o por cualquier otro panel de expertos, los cuales abordan la documentación, la correcta identificación y utilización de la base de prueba y el oráculo de prueba y la organización del equipo de pruebas. Así por ejemplo, en proyectos que siguen técnicas de gestión Scrum Ágil, en cada iteración los probadores analizan historias de usuario que describen prestaciones específicas, estiman el esfuerzo de prueba de cada prestación como parte del proceso de planificación de la iteración, identifican las condiciones de prueba (a menudo denominadas criterios de aceptación) de cada historia de usuario, ejecutan las pruebas que cubren esas condiciones y notifican el estado de cada historia de usuario (no probada, fallida o pasada) durante la ejecución de las pruebas.
- Estrategias reactivas, tales como el uso de ataques basados en defectos, en las que el equipo de pruebas espera a diseñar e implementar las pruebas hasta haber recibido el software, reaccionando ante el sistema objeto de las pruebas. Por ejemplo, cuando se realizan pruebas exploratorias en una aplicación basada en menús, podría desarrollarse un conjunto de contratos de pruebas para las prestaciones, las selecciones de menú y las pantallas. A cada probador se le asigna un conjunto de contratos de prueba, que a continuación estos utilizan para estructurar sus sesiones de pruebas exploratorias. Los probadores reportarán periódicamente los resultados de las sesiones de pruebas al Jefe de Pruebas, quien podrá revisar los contratos en base a los hallazgos.
- Las estrategias consultivas, tales como las pruebas dirigidas por los usuarios, en las que el equipo de pruebas depende de las entradas de una o varias partes interesadas para establecer las condiciones de prueba que deben cubrirse. Por ejemplo, en las pruebas de compatibilidad externalizadas para una aplicación basada en la web, la empresa puede facilitar al proveedor externo de los servicios de prueba una lista priorizada de las versiones de navegadores, el software anti-malware, los sistemas operativos, los tipos de conexión y otras opciones de configuración que desea que este evalúe en comparación con su aplicación. El proveedor de servicios de pruebas puede entonces recurrir a técnicas como pruebas por pares (para las opciones de alta prioridad) o particiones de equivalencias (para opciones de menos prioridad) para generar las pruebas.
- Las estrategias de pruebas anti regresión, como la automatización extensiva, en las que el equipo de pruebas aplica varias técnicas para gestionar el riesgo de regresión, en especial la automatización de las pruebas de regresión funcionales y/o no funcionales en uno o varios niveles. Así por ejemplo, en las pruebas de regresión de una aplicación basada en la web, los probadores pueden utilizar una herramienta de automatización de pruebas basada en la GUI para automatizar los casos de uso típicos y de excepción de la aplicación. Más tarde, estas pruebas se ejecutarán cada vez que la aplicación se modifique.

Pueden combinarse varias estrategias. Las estrategias específicas que se seleccionen deben adaptarse a las necesidades y los medios de la organización, pudiendo las organizaciones establecer estrategias a medida para operaciones y proyectos específicos.

La estrategia de pruebas puede indicar los niveles de prueba que deben ejecutarse. En estos casos, debería orientar sobre los criterios de entrada y los criterios de salida de cada nivel y las relaciones entre los niveles (por ejemplo, división de los objetivos de cobertura de las pruebas).

La estrategia de pruebas también puede hacer referencia a lo siguiente:

- procedimientos de integración
- técnicas de especificación de pruebas
- independencia de las pruebas (que puede variar dependiendo del nivel)
- estándares obligatorios y opcionales
- entornos de pruebas

- automatización de las pruebas
- herramientas de pruebas
- reusabilidad de los productos de trabajo del software y de los productos de trabajo de las pruebas
- pruebas de confirmación (repetición de pruebas) y pruebas de regresión
- control de pruebas y elaboración de informes
- mediciones y métricas de pruebas
- gestión de defectos
- enfoque de la gestión de la configuración del producto de soporte de pruebas
- funciones y responsabilidades

Podría ser necesario aplicar distintas estrategias de pruebas a corto y largo plazo. No todas las estrategias de prueba son aptas para todas las organizaciones y proyectos. Así por ejemplo, si se trata de aplicaciones críticas para la seguridad, el uso de una estrategia más intensiva podría ser más adecuado que en otros casos. Además, la estrategia de pruebas también varía según los distintos modelos de desarrollo.

2.4.3 Plan maestro de pruebas

El plan maestro de pruebas cubre todas las tareas de pruebas a realizar en un proyecto específico, incluyendo los niveles específicos a realizar y las relaciones entre estos niveles, y entre los niveles de prueba y las correspondientes actividades de desarrollo. El plan maestro de pruebas debería indicar la forma en que los probadores pretenden implementar la estrategia de pruebas en este proyecto (es decir, el enfoque de pruebas). El plan maestro de pruebas debería ser coherente con la política y la estrategia de pruebas y, en aquellas áreas específicas donde no lo sea, deberá justificar dichas desviaciones y excepciones, explicando cualquier posible impacto que pueda derivarse de estas desviaciones. Así por ejemplo, si la estrategia de pruebas de una organización prevé llevar a cabo un paso completo de pruebas de regresión en un sistema invariable inmediatamente antes de la entrega, pero el proyecto actual no incluirá ninguna prueba de regresión, el plan de pruebas debería explicar por qué se ha previsto así y qué medidas van a adoptarse para mitigar los posibles riesgos que puedan derivarse de esta variación de la estrategia habitual. El plan de pruebas también debería explicar cualquier otro efecto que quepa esperar de esta variación. Por ejemplo, el hecho de omitir las pruebas de regresión podría suponer la previsión de una entrega de mantenimiento un mes después de la entrega inicial del proyecto. El plan maestro de pruebas debe ser un complemento del plan de proyecto o de la guía de operaciones en el sentido de que debería describir el esfuerzo de pruebas a nivel de proyecto u operación.

Si bien el contenido y la estructura específicos del plan maestro de pruebas pueden variar dependiendo de la organización, de sus estándares de documentación y de la formalidad del proyecto, algunos aspectos típicos de un plan maestro de pruebas son:

- elementos a probar y a no probar
- características de calidad a probar y a no probar
- calendario y presupuesto de las pruebas (que deberían estar en línea con el presupuesto del proyecto o presupuesto operativo)
- ciclos de ejecución de las pruebas y su relación con el plan de entrega del software
- relaciones y entregables entre las pruebas y otras personas o departamentos
- definición de qué elementos de prueba están dentro y fuera del alcance para cada nivel descrito
- criterios de entrada específicos, criterios de continuación (suspensión/reanudación) y criterios de salida para cada nivel, así como las relaciones entre los niveles
- riesgos del proyecto de pruebas
- gobernanza general del esfuerzo de prueba
- responsabilidades de ejecución en cada uno de los niveles de prueba

- entradas y salidas de cada uno de los niveles de prueba.

Es frecuente que, en proyectos u operaciones menores, en los que solo existe un nivel de prueba formalizado, el plan maestro de pruebas y el plan de pruebas para ese nivel formalizado se combinen en un solo documento. Por ejemplo, si las pruebas de sistema representan el único nivel formalizado, estando las pruebas informales de componentes e integración a cargo de desarrolladores y las pruebas informales de aceptación a cargo de los clientes en el marco de un proceso de pruebas beta, entonces el plan de pruebas de sistema podrá incluir los elementos mencionados en esta sección.

Además, normalmente las pruebas dependen de otras actividades del proyecto. Si estas actividades no estuvieran suficientemente documentadas, en especial en términos de su influencia y relación con las pruebas, los temas relacionados con estas actividades podrían incluirse en el plan maestro de pruebas (o en el plan de nivel de prueba que corresponda). Por ejemplo, si el proceso de gestión de la configuración no está documentado, el plan de pruebas debería especificar cómo deben entregarse los objetos de prueba al equipo de pruebas.

2.4.4 Plan de nivel de prueba

Los planes de nivel de prueba describen las actividades específicas a realizar dentro de cada nivel de prueba o, en algunos casos, el tipo de prueba. Los planes de nivel de prueba, cuando procede, son una extensión del plan maestro de pruebas para el nivel específico o el tipo de prueba que se está documentando. Estos planes establecen detalles de plazos, tareas e hitos que no están necesariamente cubiertos en el plan maestro de pruebas. Además, en la medida en que en diferentes niveles se aplican estándares y plantillas distintos a la especificación de las pruebas, estos detalles se cubrirían en los planes de nivel de prueba.

En proyectos u operaciones menos formales, un plan de pruebas único es con frecuencia el único documento de gestión de pruebas que está por escrito. En estos casos, algunos de los elementos informativos citados más arriba en esta sección podrían estar cubiertos en dicho documento de plan de pruebas.

En el caso de proyectos Ágiles, pueden realizarse planes de pruebas de iteraciones o sprint en lugar de planes de nivel de prueba.

2.4.5 Gestión de riesgos del proyecto

Una parte importante de la correcta planificación incluye abordar los riesgos del proyecto. Los riesgos del proyecto pueden identificarse mediante el uso de procesos parecidos a los descritos en la Sección 2.3. Cuando se identifiquen riesgos del proyecto, estos deberían comunicarse al jefe de proyecto para que actúe en consecuencia. La organización de pruebas no siempre tiene la capacidad de reducir dichos riesgos. Sin embargo, el Jefe de Pruebas sí puede y debe mitigar con éxito algunos riesgos del proyecto tales como:

- entorno de pruebas y disponibilidad de herramientas
- disponibilidad y cualificación del personal de pruebas
- ausencia de normas, reglas y técnicas para el esfuerzo de prueba

Entre los enfoques para la gestión de riesgos del proyecto se encuentra la preparación del producto de soporte de pruebas en una etapa más temprana, la realización de pruebas previas de los entornos de pruebas, la realización de pruebas previas en las versiones anteriores del producto, la aplicación de criterios de entrada más estrictos en las pruebas, la imposición de requisitos de testabilidad, la participación en revisiones de los primeros productos de trabajo del proyecto y la monitorización del progreso y la calidad del proyecto.

Una vez identificado y analizado un riesgo del proyecto, existen cuatro formas posibles de gestionarlo:

1. Mitigar el riesgo a través de medidas preventivas para reducir su probabilidad y/o su impacto
2. Elaborar planes de contingencia para reducir el impacto si el riesgo se materializa
3. Transferir el riesgo a alguna otra parte para que esta lo gestione
4. Ignorar o aceptar el riesgo

La selección de la mejor opción depende de las ventajas y oportunidades que suponga la opción elegida, así como del coste y, potencialmente, de los riesgos adicionales asociados a la misma. Cuando hay un plan de contingencia establecido para un riesgo del proyecto, la mejor práctica consiste en identificar un activador (que determinará cuándo y cómo debe invocarse el plan de contingencia) y un propietario (que ejecutará el plan de contingencia).

2.4.6 Otros productos de pruebas

Las pruebas conllevan la creación de una serie de productos de trabajo adicionales, tales como informes de defectos, especificaciones de casos de prueba y bitácoras de las pruebas. Los Analistas de Pruebas y los Analistas Técnicos de Pruebas generan la mayor parte de estos productos de trabajo. En el Programa de Estudio de Analistas de Pruebas, se estudian consideraciones para generar y documentar estos productos de trabajo. El Jefe de Pruebas debería garantizar la consistencia y la calidad de estos productos de trabajo a través de las siguientes actividades:

- Establecer y hacer un seguimiento de las métricas que monitorizan la calidad de estos productos de trabajo, tales como informes del porcentaje de defectos rechazados
- Colaborar con los Analistas de Pruebas y Analistas Técnicos de Pruebas para seleccionar y personalizar las plantillas adecuadas para estos productos de trabajo
- Colaborar con los Analistas de Pruebas y los Analistas Técnicos de Pruebas para establecer estándares para esos productos de trabajo, tales como el nivel de detalle necesario de las pruebas, los registros y los informes
- Revisar los productos de trabajo de las pruebas empleando las técnicas adecuadas y a través de los participantes y las partes interesadas que procedan

El alcance, el tipo y la especificidad de la documentación de las pruebas pueden verse afectados, entre otras cosas, por el ciclo de vida de desarrollo de software elegido, los estándares y la normativa aplicable y los riesgos de la calidad del producto y del proyecto asociados al sistema específico que se está desarrollando.

Hay varias fuentes de plantillas para los productos de trabajo de las pruebas, tales como la IEEE 829 [IEEE829]. Es importante que el Jefe de Pruebas recuerde que los documentos IEEE 829 han sido diseñados para su uso en cualquier sector. Como tales, las plantillas contienen un alto nivel de detalle que puede o no, ser aplicable a una organización específica. Es una buena práctica adaptar los documentos IEEE 829 para crear plantillas estándar para su uso en una organización concreta. La aplicación coherente de las plantillas reduce los requisitos de formación y ayuda a unificar los procesos en toda la organización.

La realización de pruebas también conlleva la creación de informes de resultados de las pruebas, que normalmente elaboran los Jefes de Pruebas y se describen más adelante en este capítulo.

2.5 Estimación de pruebas

La estimación, como actividad de gestión, es la creación de un objetivo aproximado en cuanto a costes y fechas de compleción asociado a las actividades previstas en una operación o proyecto en particular. Las mejores estimaciones:

- Representan el conocimiento colectivo de los profesionales experimentados y cuentan con el apoyo de los participantes implicados
- Facilitan catálogos detallados y específicos de los costes, recursos, tareas y personas implicadas

- Presentan, para cada actividad estimada, los costes, los esfuerzos y la duración más probable

Desde hace tiempo es conocido que la estimación en el mundo de la ingeniería de software y sistemas plantea dificultades, tanto técnicas como políticas, aunque se hayan aplicado las mejores prácticas para la estimación de gestión de proyectos. La estimación de pruebas es la aplicación de estas mejores prácticas a las actividades de pruebas asociadas a un proyecto u operación.

La estimación de pruebas debería cubrir todas las actividades implicadas en el proceso de pruebas según lo descrito en el Capítulo 1. Las estimaciones de coste, esfuerzo y, en especial, duración de la ejecución de las pruebas a menudo son de especial interés para la dirección, ya que la ejecución de las pruebas normalmente se encuentra en el camino crítico del proyecto. Sin embargo, las estimaciones de ejecución de pruebas suelen ser difíciles de obtener y son poco fiables cuando la calidad global del software es baja o desconocida. Además, es probable que la familiaridad y la experiencia con el sistema afecten a la calidad de las estimaciones. Es práctica habitual estimar el número de casos de prueba necesario para ejecutar las pruebas, si bien esto es válido si podemos asumir que no hay muchos defectos en el software objeto de las pruebas. Los supuestos sobre los que se basa la estimación deberían estar documentados siempre como parte de la estimación.

La estimación de las pruebas debería tener en cuenta todos los factores que pueden influir en el coste, el esfuerzo y la duración de las actividades de pruebas. Estos factores incluyen (entre otros) los siguientes:

- Nivel de calidad requerido del sistema
- Dimensiones del sistema a probar
- Datos históricos de pruebas de proyectos de pruebas anteriores, que pueden ampliarse con datos del sector o datos de referencia procedentes de otras organizaciones
- Factores de proceso, incluyendo la estrategia de pruebas, el ciclo de vida de desarrollo o mantenimiento y la madurez del proceso, así como la precisión de la estimación del proyecto
- Factores materiales, tales como automatización y las herramientas de pruebas, entorno(s) de pruebas, datos de prueba, entorno(s) de desarrollo, documentación del proyecto (por ejemplo, requisitos, diseños, etc.), y productos de trabajo reutilizables.
- Factores de personas, incluyendo jefes y líderes técnicos, compromisos y expectativas de ejecutivos y alta dirección; capacidades, experiencia y actitudes en el equipo del proyecto, estabilidad del equipo del proyecto, relaciones del equipo del proyecto, apoyo al entorno de pruebas y depuración, disponibilidad de contratistas y consultores capacitados y conocimiento del dominio
- Complejidad del proceso, tecnología, organización, número de partes interesadas de las pruebas, composición y ubicación de los subequipos
- Aceleración significativa, formación y necesidades de orientación
- Asimilación o desarrollo de nuevas herramientas, tecnología, procesos, técnicas, hardware a medida o una gran cantidad de productos de soporte de pruebas
- Requisitos para un alto nivel de especificación de pruebas detallada, especialmente para cumplir con un estándar de documentación desconocido
- Calendario complicado de llegada de componentes, especialmente por lo que respecta a las pruebas de integración y al desarrollo de las pruebas
- Datos de prueba frágiles (por ejemplo, datos sensibles al tiempo).

La calidad del software entregado para las pruebas también constituye un factor importante que el Jefe de Pruebas debería tener en cuenta en su estimación. Así por ejemplo, si los desarrolladores adoptan mejores prácticas, tales como la realización de pruebas unitarias automatizadas o la integración continua, se habrá eliminado hasta un 50% de los defectos antes de la entrega del código al equipo de pruebas (remítase a [Jones11] para más información sobre la efectividad de eliminación de defectos de estas prácticas). Algunos han reportado que las metodologías Ágiles, incluyendo el desarrollo guiado por pruebas, permiten obtener niveles más altos de calidad antes de las pruebas.

La estimación puede hacerse de forma ascendente o descendente. Algunas de las técnicas que pueden emplearse para la estimación de pruebas, de forma individual o conjunta, son:

- Intuición, predicciones y experiencia anterior
- Estructuras de Desglose del Trabajo (EDT)
- Sesiones de estimación del equipo (por ejemplo, Delphi de banda ancha)
- Estándares y normas de la empresa
- Porcentajes del esfuerzo global del proyecto o niveles de personal (por ejemplo, ratio probador-programador)
- Historia y métricas organizativas, incluyendo modelos derivados de métricas para calcular el número de defectos, el número de ciclos de prueba, el número de casos de prueba, el esfuerzo medio de cada prueba y el número de ciclos de regresión implicados
- Valores promedios del sector y modelos predictivos, tales como puntos de función, líneas de código, esfuerzo estimado del desarrollador y otros parámetros de proyecto (por ejemplo, remítase a [Jones07])

En la mayoría de los casos, la estimación, una vez elaborada, debe entregarse a la dirección, junto con una justificación (ver la Sección 2.7). Con frecuencia, se suceden negociaciones, que a menudo suponen una reelaboración de la estimación. Lo ideal es que la estimación final de la prueba represente el mejor balance posible entre los objetivos organizativos y del proyecto en las áreas de calidad, plazo, presupuesto y prestaciones.

Es importante recordar que las estimaciones están basadas en la información disponible en el momento de su elaboración. En etapas tempranas del proyecto, la información puede ser bastante limitada. Además, la información de la que se dispone en etapas tempranas del proyecto puede sufrir cambios con el tiempo. Para seguir siendo exactas, las estimaciones deberían actualizarse para reflejar la información nueva y modificada.

2.6 Definición y uso de las métricas de pruebas

Un cliché del mundo de la gestión establece que lo que se mide se hace. De la misma forma, lo que no se mide no se hace, porque es fácil no tener en cuenta aquello que no se mide. Por lo tanto, es importante establecer el conjunto de métricas adecuado para cualquier labor, incluyendo las pruebas.

Las métricas de pruebas pueden clasificarse en una o varias de las siguientes categorías:

- Métricas del proyecto, que miden el progreso hacia los criterios de salida previstos del proyecto, tales como el porcentaje de casos de prueba ejecutados, pasados y fallidos
- Métricas del producto, que miden algunos atributos del producto, tales como el alcance en el que ha sido probado o la densidad del defecto
- Métricas del proceso, que miden la capacidad del proceso de pruebas de desarrollo, tales como el porcentaje de defectos detectados por las pruebas
- Métricas de las personas, que miden la capacidad de los individuos o los grupos, tales como la implementación de casos de prueba dentro de un calendario dado

Una métrica puede estar clasificada en dos, tres o incluso en cuatro categorías. Así por ejemplo, un gráfico de tendencias que demuestre la tasa diaria de llegada de defectos puede asociarse a un criterio de salida (cero defectos nuevos detectados en una semana), la calidad del producto (las pruebas no pueden ubicar más defectos en el producto) y la capacidad del proceso de pruebas (las pruebas encuentran un gran número de defectos en una etapa temprana durante el periodo de ejecución de las pruebas).

Las métricas de personas son especialmente sensibles. Los jefes a veces confunden las métricas que son fundamentalmente métricas del proceso con las métricas de personas, algo que puede dar lugar

a resultados desastrosos si las personas actúan para sesgar las métricas para que les favorezcan. En el Capítulo 7 de este programa de estudio y en el programa de estudio del Jefe de Pruebas de Nivel Experto [ISTQB ETL SYL] se estudia la correcta motivación y evaluación del personal de pruebas.

En el Nivel Avanzado, nos centramos más en el uso de las métricas para medir el progreso de las pruebas, es decir, en las métricas del proyecto. Algunas de las métricas del proyecto que se utilizan para medir el progreso de las pruebas también se refieren al producto y al proceso. Para más información sobre el uso de las métricas del producto y del proceso, remítase al programa de estudio de Nivel Experto "Gestión de Pruebas". Para más información sobre el uso de las métricas del proceso, remítase al programa de estudio de Nivel Experto del ISTQB "Mejora del proceso de pruebas" [ISTQB ITP SYL].

El uso de métricas permite a los probadores comunicar resultados de una manera consistente, y llevar a cabo un seguimiento coherente del progreso a lo largo del tiempo. Los Jefes de Pruebas tienen que presentar métricas en distintas reuniones a las que pueden asistir partes interesadas de distintos niveles, desde personal técnico hasta miembros de la alta dirección. Dado que las métricas a veces se utilizan para establecer el éxito general de un proyecto, debe tenerse especial cuidado a la hora de determinar qué se desea controlar, con qué frecuencia deben reportarse resultados y el método a utilizar para presentar la información. En particular, el Jefe de Pruebas debe tener en cuenta lo siguiente:

- Definición de métricas. Debe definirse un conjunto limitado de métricas útiles. Las métricas deben definirse tomando como base los objetivos específicos del proyecto, el proceso y/o el producto. Las métricas deberían definirse con vistas a establecer un equilibrio, dado que una sola métrica puede dar una falsa impresión de los estados o las tendencias. Una vez definidas las métricas, su interpretación debe ser objeto de acuerdo por todas las partes interesadas, con el fin de evitar confusión a la hora de abordar las mediciones. A menudo tendemos a definir demasiadas métricas, en lugar de concentrarnos en las más pertinentes.
- Seguimiento de las métricas. La comunicación y la fusión de las métricas deben ser actividades lo más automatizadas posible con el fin de reducir el tiempo dedicado a introducir y procesar las mediciones. Las variaciones que se producen con el tiempo en las mediciones de una métrica específica pueden reflejar información distinta de la acordada en la fase de definición de las métricas. El Jefe de Pruebas debería estar preparado para analizar detenidamente cualquier posible divergencia en las mediciones de las expectativas, así como los motivos de dicha divergencia.
- Comunicación de las métricas. El objetivo es proporcionar un conocimiento inmediato de la información, con fines de gestión. Las presentaciones pueden mostrar una "instantánea" de una métrica en un momento determinado o mostrar la evolución de la(s) métrica(s) con el tiempo, de manera que se puedan evaluar las tendencias.
- Validez de las métricas. El Jefe de Pruebas también debe verificar la información que se comunica. Puede darse el caso de que las mediciones de una métrica no reflejen el verdadero estado de un proyecto o trasladen una tendencia generalmente positiva o negativa. Antes de presentar los datos, el Jefe de Pruebas debe revisarlos y asegurar su exactitud y cerciorarse del mensaje que desea trasladar.

Existen cinco dimensiones primarias sobre las que se monitoriza el progreso de las pruebas:

- Riesgos (de la calidad) del producto
- Defectos
- Pruebas
- Cobertura
- Confianza

Los riesgos del producto, los defectos, las pruebas y la cobertura pueden medirse, y a menudo se miden y comunican, de una forma específica a lo largo del proyecto o la operación. Si estas mediciones se refieren a criterios de salida definidos en el plan de pruebas, pueden proporcionar un

estándar objetivo contra el que comparar la compleción del esfuerzo de prueba. La confianza puede medirse a través de encuestas o empleando la cobertura como métrica de sustitución; sin embargo, muchas veces la confianza también se comunica de manera subjetiva.

Las métricas relacionadas con los riesgos del producto incluyen:

- Porcentaje de riesgos totalmente cubiertos por pruebas superadas
- Porcentaje de riesgos en los que algunas o todas las pruebas fallan
- Porcentaje de riesgos pendientes de ser íntegramente probados
- Porcentaje de riesgos cubiertos, clasificados por categoría de riesgo
- Porcentaje de riesgos identificados después del análisis inicial de los riesgos de la calidad

Las métricas relacionadas con los defectos incluyen:

- Número acumulado reportado (encontrado) frente al número acumulado resuelto (solucionado)
- Tiempo medio transcurrido entre fallos o frecuencia de aparición de fallos
- Desglose del número o del porcentaje de defectos categorizados por lo siguiente:
 - Elementos o componentes de prueba específicos
 - Causas raíz
 - Origen del defecto (por ejemplo, especificación de requisitos, nueva prestación, regresión, etc.)
 - Entregas de pruebas
 - Fases de introducido, detectado y eliminado
 - Prioridad/severidad
 - Informes rechazados o duplicados
- Tendencias en el período de tiempo transcurrido desde la comunicación del defecto hasta su resolución
- Número de soluciones de defectos que dieron lugar a nuevos defectos (a veces denominados bugs hijas)

Entre las métricas relacionadas con las pruebas se encuentran:

- Número total de pruebas planificadas, especificadas (implementadas), ejecutadas, pasadas, fallidas, bloqueadas y omitidas
- Estado de las pruebas de regresión y confirmación, incluyendo las tendencias y los totales de los fallos de las pruebas de regresión y de las pruebas de confirmación
- Horas de pruebas previstas por día frente a las horas reales ejecutadas
- Disponibilidad del entorno de pruebas (porcentaje de horas de pruebas previstas cuando el equipo de pruebas pueda utilizar el entorno de pruebas)

Entre las métricas relacionadas con la cobertura de pruebas se encuentran:

- Requisitos y cobertura de los elementos de diseño
- Cobertura de riesgos
- Cobertura del entorno/configuración
- Cobertura del código

Es importante que los Jefes de Pruebas sepan cómo interpretar y utilizar las métricas de cobertura para entender y reportar el estado de las pruebas. Para los niveles más altos de pruebas, tales como las pruebas de sistema, las pruebas de integración del sistema y las pruebas de aceptación, las principales bases de prueba suelen ser productos de trabajo como especificaciones de requisitos, especificaciones de diseño, casos de uso, historias de usuario, riesgos del producto, entornos soportados y configuraciones soportadas. Las métricas de cobertura del código estructural se refieren a niveles más bajos de pruebas, tales como las pruebas unitarias (por ejemplo, cobertura de sentencia y rama) y las pruebas de integración de componentes (por ejemplo, cobertura de interfaz). Si bien los Jefes de Pruebas pueden emplear métricas de cobertura del código para medir el alcance

en el que sus pruebas ejercitan la estructura del sistema objeto de las pruebas, la comunicación de los resultados de las pruebas de más alto nivel normalmente no debería incluir métricas de cobertura del código. Asimismo, los Jefes de Prueba deberían saber que, a pesar de que las pruebas unitarias y las pruebas de integración de componentes alcancen el 100% de sus objetivos de cobertura estructural, aun quedan defectos y elementos de calidad pendientes de abordar en niveles de prueba más altos.

Las métricas también pueden referirse a las actividades del proceso fundamental de pruebas (descrito en el Programa de Estudio Fundamentos y en este programa de estudio). De esta forma, las métricas pueden utilizarse durante todo el proceso de pruebas para monitorizar el propio proceso de las pruebas y el progreso hacia los objetivos del proyecto.

Entre las métricas para monitorizar la planificación y el control de las pruebas se encuentran:

- Riesgo, requisitos y otros elementos de cobertura de la base de prueba
- Descubrimiento de defectos
- Horas previstas frente a horas reales necesarias para desarrollar productos de soporte de pruebas y casos de prueba

Entre las métricas disponibles para monitorizar las actividades de análisis de las pruebas se encuentran:

- Número de condiciones de prueba identificadas
- Número de defectos encontrados durante el análisis de las pruebas (por ejemplo, identificando los riesgos u otras condiciones de prueba a través de la base de prueba)

Entre las métricas disponibles para monitorizar las actividades de diseño de las pruebas se encuentran:

- Porcentaje de condiciones de prueba que cubre los casos de prueba
- Número de defectos encontrados durante el diseño de las pruebas (por ejemplo, desarrollando pruebas contra la base de prueba)

Entre las métricas disponibles para monitorizar las actividades de implementación de las pruebas se encuentran:

- Porcentaje de entornos de pruebas configurados
- Porcentaje de registros de datos de prueba cargados
- Porcentaje de casos de prueba automatizados

Entre las métricas disponibles para monitorizar las actividades de ejecución de las pruebas se encuentran:

- Porcentaje de casos de prueba previstos ejecutados, pasados y fallidos
- Porcentaje de condiciones de prueba cubierto por los casos de prueba ejecutados (y/o pasados)
- Defectos reportados/resueltos previstos frente a defectos reportados/resueltos reales
- Cobertura alcanzada prevista frente a cobertura alcanzada real

Las métricas para monitorizar el progreso de las pruebas y las actividades de compleción incluirán el mapeado hasta los hitos, los criterios de entrada y los criterios de salida (según lo definido y aprobado en la planificación de las pruebas), que a su vez podrán incluir lo siguiente:

- Número previsto y número ejecutado de condiciones de prueba, casos de prueba o especificaciones de prueba, desglosado en función de si las pruebas se pasaron o fallaron
- Número total de defectos, a menudo desglosado por severidad, prioridad, estado actual, subsistema afectado u otra clasificación (remítase al Capítulo 4)
- Número de cambios requeridos, aceptados, integrados y probados
- Coste previsto frente a coste real

- Duración prevista frente a duración real
- Fechas previstas de hitos de pruebas frente a fechas reales de los hitos de las pruebas
- Fechas previstas de los hitos del proyecto asociados a las pruebas frente a fechas reales de los hitos del proyecto asociados a las pruebas (por ejemplo, congelación del código)
- Estados de riesgo (de la calidad) del producto, a menudo desglosados por riesgos mitigados y no mitigados, áreas principales de riesgo, nuevos riesgos, nuevos riesgos descubiertos tras un análisis de las pruebas, etc.
- Porcentaje de pérdida del esfuerzo de prueba, coste o tiempo debido a eventos bloqueantes o a cambios previstos
- Estado de las pruebas de regresión y confirmación

Entre las métricas para monitorizar las actividades de cierre de pruebas se encuentran:

- Porcentaje de casos de prueba ejecutados, pasados, fallidos, bloqueados y omitidos durante la ejecución de las pruebas
- Porcentaje de casos de prueba comprobados en el repositorio de casos de prueba reutilizables
- Porcentaje de casos de prueba automatizados, o previstos de automatizar frente a casos de prueba reales
- Porcentaje de casos de prueba integrados en las pruebas de regresión
- Porcentaje de informes de defectos resueltos/no resueltos
- Porcentaje de productos de trabajo de las pruebas identificados y archivados

Además, para monitorizar el proceso de pruebas a menudo se emplean técnicas estándar para la gestión de proyectos, tales como estructuras de desglose de tareas. En los equipos Ágiles, las pruebas forman parte del progreso de una historia de usuario en el diagrama de trabajo pendiente. Cuando se utilizan técnicas de gestión Lean, el progreso de las pruebas historia por historia a menudo se monitoriza desplazando la tarjeta de historias de usuario por una columna del tablero Kanban.

Dado un conjunto definido de métricas, las mediciones pueden reportarse verbalmente a través de narrativas, numéricamente mediante tablas o pictóricamente mediante gráficos. Las mediciones pueden utilizarse para una serie de objetivos, entre los que se encuentran:

- El análisis, para descubrir qué tendencias y causas pueden discernirse en base a los resultados de las pruebas
- La información, para comunicar los resultados de la prueba a los participantes del proyecto y a las partes interesadas
- El control, para cambiar el curso de la prueba o el proyecto en su totalidad y para monitorizar los resultados de esa corrección de curso

Las formas adecuadas de recopilar, analizar y comunicar estas medidas de las pruebas dependerán de las necesidades específicas de información, objetivos y capacidades de las personas que van a utilizar las mediciones. Además, el contenido específico de los informes de prueba debería variar en función del público.

A efectos de control de las pruebas, es fundamental que las métricas a lo largo de todo el proceso de pruebas (una vez finalizada la planificación de pruebas) faciliten al Jefe de Pruebas la información necesaria para orientar el esfuerzo de prueba hacia la correcta compleción de la misión, estrategias y objetivos de las pruebas. Por lo tanto, durante la etapa de planificación deben tenerse en cuenta estas necesidades de información, y la monitorización debe incluir la recopilación de todas las métricas de productos de trabajo que sean necesarias. El volumen de información necesario y el esfuerzo dedicado a su recopilación dependerán de varios factores del proyecto, incluyendo las dimensiones, la complejidad y el riesgo.

El control de las pruebas debe dar respuesta a la información generada por las pruebas, además de a las condiciones cambiantes del proyecto o entorno. Así por ejemplo, si las pruebas dinámicas revelan la existencia de “clústeres” de defectos en áreas que se consideraban poco predispuestas a incluir muchos defectos, o si el período de ejecución de las pruebas se acorta debido a un retraso en el inicio de las pruebas, deberá revisarse tanto el análisis de riesgos como el plan. Esto podría dar lugar a la re-priorización de las pruebas y a la reasignación del resto del esfuerzo de ejecución de pruebas.

El control de las pruebas debería llevarse a cabo si el informe de progreso de las pruebas pone de manifiesto una divergencia en el plan de pruebas. El control de las pruebas tiene por objeto reorientar el proyecto y/o las pruebas en un sentido más correcto. Cuando se utilicen los resultados de las pruebas para incluir o medir los esfuerzos de control en el proyecto, deberían tenerse en cuenta las siguientes opciones:

- Revisar los análisis de los riesgos de la calidad, las prioridades de las pruebas y/o los planes de las pruebas
- Añadir recursos o incrementar de otra forma el proyecto o el esfuerzo de prueba
- Retrasar la fecha de entrega
- Relajar o fortalecer los criterios de salida de las pruebas
- Cambiar el alcance (funcional y/o no funcional) del proyecto

Implementar estas opciones normalmente requiere el consenso entre las partes interesadas del proyecto o de la operación y el consentimiento de los jefes del proyecto o de la operación.

La información incluida en un informe de pruebas dependerá en gran parte de las necesidades de información del público objetivo, por ejemplo, la dirección del proyecto o la dirección del negocio. Para un jefe de proyecto puede ser de interés contar con información detallada sobre los defectos, mientras que, para el director comercial, el estado de los riesgos del producto podría ser la cuestión clave del informe.

2.7 Valor de negocio de las pruebas

El Jefe de Pruebas debe trabajar para optimizar las pruebas con el fin de ofrecer un alto valor de negocio. Las pruebas excesivas no ofrecen un alto valor de negocio, ya que las pruebas impondrán retrasos injustificados y costarán más de lo que ahorran. Las pruebas escasas tampoco ofrecen un alto valor de negocio, ya que muchos defectos llegarán a los usuarios. El punto óptimo está entre estos dos extremos. El Jefe de Pruebas tiene que ayudar a las partes interesadas de las pruebas a entender este punto óptimo y el valor que ofrecen las pruebas.

Si bien la mayoría de organizaciones considera que las pruebas tienen valor en algún sentido, pocos directivos, incluyendo los Jefes de Pruebas, pueden cuantificar, describir o expresar dicho valor. Además, muchos Jefes de Pruebas, líderes de pruebas y probadores se concentran en los detalles tácticos de las pruebas (aspectos específicos de la función o nivel de prueba, al tiempo que ignoran otras cuestiones estratégicas de mayor calado (de nivel superior) relacionadas con las pruebas que son de interés para otros participantes en los proyectos, especialmente los directivos.

Las pruebas aportan valor a la organización, al proyecto y/o a la operación tanto en el aspecto cuantitativo como cualitativo:

- Los valores cuantitativos incluyen encontrar defectos que se previenen o se arreglan antes de la entrega (no arreglados pero sí documentados, quizás mediante soluciones alternativas), detectar defectos conocidos antes de la entrega, reducir el riesgo ejecutando pruebas y entregar información sobre el estado del proyecto, el proceso y el producto
- Los valores cualitativos incluyen una mejor reputación para la calidad, entregas más fluidas y predecibles, mayor confianza, protección de responsabilidad legal y menor riesgo de pérdida de misiones completas o incluso de vidas

Los Jefes de Pruebas deben entender cuáles de estos valores se aplican a su organización, proyecto y/o operación, y deben ser capaces de comunicar información sobre las pruebas en términos de estos valores.

Un método bien establecido para medir el valor cuantitativo y la eficiencia de las pruebas es conocido como coste de la calidad (o, en ocasiones, coste de la calidad deficiente). El coste de la calidad supone clasificar los costes del proyecto y los costes operativos en cuatro categorías relacionadas con el coste de los defectos del producto:

- Costes de prevención, como por ejemplo: formar a los desarrolladores para que escriban un código más mantenible o seguro
- Costes de detección, como por ejemplo: escribir casos de prueba, configurar entornos de pruebas y revisar requisitos
- Costes de fallos internos, como por ejemplo: arreglar defectos detectados durante las pruebas o revisiones, antes de la entrega
- Costes de fallos externos, como por ejemplo: soportar costes asociados con la entrega a los clientes de un software defectuoso

Parte del presupuesto de pruebas es un coste de detección (es decir, dinero que se gastaría, aunque los probadores no encontraran ningún defecto, como puede ser el dinero dedicado a las pruebas de desarrollo), mientras que el resto son costes de fallos internos (es decir, el coste real asociado a los defectos encontrados). Los costes totales de detección y fallos internos están muy por debajo de los costes de fallos externos, lo que hace de las pruebas un valor excelente. Al distribuir los costes en estas cuatro categorías, los Jefes de Pruebas pueden crear un caso de negocio convincente para las pruebas.

Para más información sobre el valor de negocio de las pruebas, incluyendo los costes de calidad, remítase a [Black03].

2.8 Pruebas distribuidas, externalizadas e internalizadas

En muchos casos, parte o a veces incluso todo el esfuerzo de prueba lo llevan a cabo personas ubicadas en distintos emplazamientos, que están contratadas por distintas empresas y que están lejos del equipo del proyecto. Si el esfuerzo de prueba se lleva a cabo en varios emplazamientos, ese esfuerzo de prueba se denomina esfuerzo de prueba distribuido. Si el esfuerzo de prueba se lleva a cabo en uno o más emplazamientos por personas que no son empleados de la empresa o que no se encuentran en la misma ubicación que el equipo del proyecto, ese esfuerzo de prueba se denomina esfuerzo de prueba externalizado. Si el esfuerzo de prueba lo realizan personas que están en el mismo emplazamiento que el equipo del proyecto pero que no son compañeros de trabajo, el esfuerzo de prueba se denomina esfuerzo de prueba internalizado.

La necesidad de canales de comunicación claros y de expectativas bien definidas para las misiones, tareas y entregables es común para el conjunto de estos esfuerzos de prueba. El equipo del proyecto debe depender menos de los canales de comunicación informales, como las conversaciones de pasillo o los eventos sociales con los compañeros. Es importante que las formas de comunicación estén definidas, incluso las referidas a temas como la remisión de problemas, el tipo de información que debe comunicarse y los métodos de comunicación a utilizar. Todo el mundo, en todas sus relaciones con el equipo, debe tener claro cuáles son sus funciones y responsabilidades, así como cuáles son las funciones y responsabilidades de los demás, para evitar malentendidos y expectativas irrealistas. Aspectos como el emplazamiento, la zona horaria, las diferencias culturales y de idioma propician la aparición de problemas de comunicación y expectativas.

La necesidad de una alineación de las metodologías también es común en todos los esfuerzos de prueba. Si bien cualquier proyecto puede ser objeto de un desajuste en las metodologías, es más probable que este se de en situaciones en las que el trabajo está distribuido y/o se lleva a cabo por

entidades externas. Si dos grupos de pruebas utilizan metodologías diferentes o si el grupo de prueba utiliza una metodología diferente de la del grupo de desarrollo o gestión de proyectos, podrían plantearse problemas importantes, especialmente durante la ejecución de las pruebas. Por ejemplo, si un cliente está utilizando un desarrollo Ágil, mientras que el proveedor de servicios de pruebas cuenta con una metodología de pruebas preestablecida que asume un ciclo de vida secuencial, los tiempos y la naturaleza de la entrega de los elementos de prueba al proveedor de servicios de pruebas constituirán un punto de contención.

Para las pruebas distribuidas, la división del trabajo de pruebas entre los distintos emplazamientos puede decidirse de forma explícita e inteligente. Sin esta orientación, ni el grupo más competente podrá realizar el trabajo de pruebas para el que está cualificado. Asimismo, si los equipos implicados no saben cuáles son sus responsabilidades, es posible que no hagan lo que se espera de ellos. Deben comunicarse con claridad qué se espera de cada uno de los equipos. Sin una gestión adecuada, el trabajo de pruebas en su conjunto podrá tener vacíos (que aumentan el riesgo de la calidad residual en la entrega) y solapamientos (que reducen la eficiencia).

Finalmente, para todos los esfuerzos de prueba, es vital que el equipo del proyecto en su conjunto desarrolle y mantenga la confianza en que todos los equipos de pruebas desempeñarán sus funciones correctamente a pesar de los límites organizativos, culturales, de idioma y geográficos. La falta de confianza genera ineficiencias y retrasos asociados con las actividades de verificación, atribución de responsabilidades por los problemas y entrar en el juego de la política organizativa.

2.9 Gestión de la aplicación de los estándares del sector

En los programas de estudio Fundamentos y Nivel Avanzado se hacen referencia a una serie de estándares. Estas normas referenciadas se refieren a ciclos de desarrollo de software, pruebas de software, características de calidad de software, revisiones y gestión de defectos. Los Jefes de Prueba deben conocer las normas, así como la política de uso de dichas normas que sigue su organización y si dichas normas son obligatorias, necesarias o útiles para ellos.

Las normas pueden proceder de diferentes fuentes, como por ejemplo:

- Internacionales o con alcance internacional
- Nacionales, como son las aplicaciones nacionales de los estándares internacionales
- Del ámbito específico, como sucede cuando un estándar nacional o internacional se adapta a un ámbito concreto, o se desarrolla para ámbitos específicos

Entre los organismos internacionales de normalización se encuentran la ISO y el IEEE. La ISO es la Organización Internacional de Normalización, también denominada IOS (del nombre en inglés *International Organization for Standardization*). Está compuesta por miembros que representan, para su país, al organismo nacional más relevante del área a normalizar. Este organismo internacional ha promovido una serie de normas de gran utilidad para los probadores de software, tales como la ISO 9126 (sustituída por la ISO 25000), la ISO 12207 [ISO12207], y la ISO 15504 [ISO15504].

El IEEE es el Instituto de Ingenieros Eléctricos y Electrónicos, una organización profesional con sede en Estados Unidos, pero con delegaciones nacionales en más de cien países. Esta organización ha propuesto un gran número de normas muy útiles para los probadores de software, entre las que se encuentran la IEEE 829 [IEEE829] y la IEEE 1028 [IEEE1028].

Muchos países cuentan con sus propios estándares nacionales. Algunas de estas normas pueden ser útiles para las pruebas de software. Un ejemplo de ello es la norma británica BS 7925-2 [BS7925-2], que ofrece información sobre muchas de las técnicas de diseño de pruebas previstas en los programas de estudio de Analistas de Pruebas de Nivel Avanzado y Analistas Técnicos de Pruebas de Nivel Avanzado.

Algunas normas son específicas de algunos ámbitos, y algunas de estas normas tienen implicaciones para las pruebas de software, la calidad del software y el desarrollo de software. Por ejemplo, en el ámbito de la aviación, la norma DO-178B (y su equivalente europeo ED 12B) de la Administración Federal de Aviación de los Estados Unidos es aplicable al software empleado en aeronaves civiles. Esta norma prescribe ciertos niveles de criterios de cobertura estructural basándose en el nivel de criticidad del software objeto de las pruebas.

Otro ejemplo de norma específica de un ámbito es el Título 21 CFR Parte 820 [FDA21] de la Agencia de Alimentos y Medicamentos de los Estados Unidos sobre sistemas médicos. Esta norma recomienda determinadas técnicas de pruebas estructurales y funcionales. La norma también sugiere algunas estrategias y principios de prueba que se ajustan al contenido de los programas de estudio del ISTQB.

En algunos casos, las pruebas están influenciadas por normas o metodologías generalizadas que no se refieren principalmente a las pruebas, pero que afectan mucho al contexto de los procesos de software en los que se llevan a cabo las pruebas. Un ejemplo es el marco de mejora de procesos de software CMMI®. En él se incluyen dos áreas de procesos clave, verificación y validación, que a menudo se interpretan como que se refieren a niveles de pruebas (tales como pruebas de sistema y pruebas de aceptación, respectivamente). También tiene implicaciones en términos de la estrategia de pruebas, que a menudo se interpreta como que requiere la inclusión de pruebas basadas en requisitos analíticos como parte de la estrategia de pruebas.

Otros tres ejemplos importantes son PMI's PMBOK, PRINCE2® e ITIL®. PMI y PRINCE2 son marcos de gestión de proyectos comúnmente utilizados en Estados Unidos y Europa respectivamente. ITIL es un marco para garantizar que un grupo de TI ofrece servicios valiosos a la organización en la que está integrado. La terminología y las actividades especificadas en estos marcos difieren significativamente de las de los programas de estudio y el glosario del ISTQB. Cuando trabaje en una organización que utilice PMI's PMBOK, PRINCE2, y/o ITIL, el Jefe de Pruebas debe conocer suficientemente los marcos seleccionados, sus implementaciones y su terminología para poder trabajar de forma efectiva en ese contexto.

Independientemente de las normas o metodologías que se adopten, es importante recordar que ambos han sido creados por grupos de profesionales. Las normas reflejan la experiencia y sabiduría colectivas del grupo fuente, pero también sus debilidades. Los Jefes de Pruebas deben conocer las normas aplicables a su entorno y contexto, ya sean normas formales (internacionales, nacionales o de ámbito específico) o normas internas y prácticas recomendadas.

Cuando esté considerando el uso de varias normas, recuerde que algunas normas son inconsistentes con otras, proponiendo incluso definiciones conflictivas. El Jefe de Pruebas debería determinar la utilidad de las distintas normas para el contexto específico en el que se están llevando a cabo las pruebas. La información que ofrece una norma puede resultar útil para un proyecto, pero también puede hacerlo más complicado. No obstante, las normas pueden facilitar una referencia a mejores prácticas probadas, y ofrecer una base para organizar el proceso de pruebas.

En algunos casos, el cumplimiento de las normas es obligatorio y tiene consecuencias en las pruebas. El Jefe de Pruebas debe conocer estos requisitos sobre la adhesión a normas y asegurarse de que se mantiene un cumplimiento adecuado.

3. Revisiones - 180 minutos

Palabras clave

auditoría, revisión informal, inspección, revisión de gestión, moderador, revisión, plan de revisión, revisor, revisión técnica, revisión guiada

Objetivos de aprendizaje de las Revisiones

3.2 Revisiones de gestión y auditorías

TM-3.2.1 (K2) Comprender las características clave de las revisiones de gestión y las auditorías

3.3 Revisiones de gestión

TM-3.3.1 (K4) Analizar un proyecto para seleccionar el tipo de revisión adecuado y definir un plan para la realización de revisiones, con el fin de garantizar una correcta ejecución, un buen seguimiento y una correcta responsabilidad

TM-3.3.2 (K2) Conocer los factores, las aptitudes y el tiempo necesario para la participación en revisiones

3.4 Métricas para las revisiones

TM-3.4.1 (K3) Definir métricas de proceso y de producto a utilizar en las revisiones

3.5 Gestión de revisiones formales

TM-3.5.1 (K2) Explicar mediante ejemplos las características de una revisión formal

3.1 Introducción

Las revisiones se introdujeron en el Programa de Estudio de Nivel Fundamentos del ISTQB como actividades de pruebas estáticas para productos. Las auditorías y las revisiones de gestión se centran más en el proceso de software que en los productos de trabajo de software.

Dado que las revisiones son una forma de prueba estática, los Jefes de Pruebas pueden ser responsables de su éxito general, en particular por lo que respecta a los productos de soporte de las pruebas. En el contexto más amplio de los proyectos de software, no obstante, esta responsabilidad debería ser una cuestión de política organizativa. Dada la amplia aplicación posible de las revisiones formales en muchas disciplinas, tanto antes como dentro de los proyectos de software, la parte responsable puede ser un Jefe de Pruebas o un Responsable del Aseguramiento de la Calidad o un Coordinador de Revisiones debidamente formado. En este programa de estudio, se considera que la parte responsable (independientemente de quién sea) es el líder de las revisiones.

El líder de las revisiones debería garantizar que existe un entorno propicio para la implementación de los factores de éxito previstos en el Programa de Estudio Fundamentos del ISTQB. Asimismo, el líder de las revisiones debería diseñar un plan de mediciones para asegurarse de que las revisiones proporcionan un valor efectivo.

Dado que los probadores disponen de un gran conocimiento sobre el comportamiento operativo y las características necesarias del sistema de software, es importante que el probador participe en los procesos de revisión.

Las personas que participen en revisiones deben contar con la debida formación en materia de revisiones que les permita entender mejor cuáles son sus respectivas funciones en un proceso de revisión dado. Todos los participantes de un proceso de revisión deben comprometerse a llevar a cabo un proceso de revisión bien llevado.

Cuando se hacen correctamente, las revisiones son la mayor, mejor y más rentable contribución a la calidad general ofrecida. Por lo tanto, es de vital importancia que los líderes de las revisiones sean capaces de implementar revisiones eficientes en sus proyectos y de demostrar las ventajas que suponen estas revisiones.

Algunas de las posibles revisiones en el marco de un proyecto son:

- Revisiones contractuales, iniciadas al principio del proyecto y en los principales hitos del proyecto
- Revisiones de los requisitos, iniciadas cuando hay requisitos disponibles para su revisión, que idealmente cubren requisitos funcionales y no funcionales
- Revisiones del diseño de máximo nivel, iniciadas cuando el diseño arquitectónico general está disponible para su revisión
- Revisiones del diseño detallado, iniciadas cuando el diseño detallado está disponible para su revisión
- Revisiones del código, llevadas a cabo a medida que se crean módulos individuales de software, las cuales, además del propio código, pueden incluir las pruebas unitarias y sus resultados
- Revisiones del producto de trabajo de las pruebas, que pueden cubrir los planes de pruebas, las condiciones de prueba, los resultados del análisis de los riesgos de la calidad, los datos de prueba, los entornos de pruebas y los resultados de las pruebas
- Revisiones de las entradas de las pruebas (disposición de las pruebas) y revisiones de las salidas de las pruebas para cada nivel de prueba, que comprueban respectivamente los criterios de entrada de las pruebas antes de iniciar la ejecución de las pruebas y los criterios de salida de las pruebas antes de finalizar las pruebas

- Revisiones de aceptación, que se utilizan para obtener la aprobación de un sistema por parte del cliente o las partes interesadas

Además de aplicar varios tipos de revisiones a un producto, es importante que el líder de las revisiones recuerde que, aunque las revisiones pueden encontrar defectos en el documento estático, las revisiones deberían ampliarse a otras formas de pruebas estáticas (tales como el análisis estático) y pruebas dinámicas del código. El uso de una combinación de estas técnicas mejora la cobertura de las pruebas y permitirá localizar más defectos.

Las distintas técnicas se concentran en distintos aspectos. Por ejemplo, una revisión puede eliminar un problema en el nivel de requisitos antes de que el problema se implemente en el código. El análisis estático puede ayudar a aplicar los estándares de codificación y a comprobar que no existen problemas que sean demasiado laboriosos como para que el equipo los descubra mediante el examen del producto de trabajo. Las inspecciones no solo pueden dar lugar al descubrimiento y la eliminación de los defectos, sino que también puede formar a los autores para evitar que creen defectos en sus productos de trabajo.

En el Programa de Estudio Fundamentos del ISTQB se introducen los siguientes tipos de revisión:

- Revisión informal
- Revisión guiada
- Revisión técnica
- Inspección

Además, los Jefes de Pruebas también pueden participar en las siguientes revisiones:

- Revisiones de gestión
- Auditorías

3.2 Revisiones de gestión y auditorías

Las revisiones de gestión se utilizan para monitorizar el progreso, evaluar el estado, y tomar decisiones sobre acciones futuras. Estas revisiones soportan decisiones sobre el futuro del proyecto, tales como adaptar el nivel de recursos, implementar acciones correctivas o modificar el alcance del proyecto.

Las siguientes son características clave de las revisiones de gestión:

- Llevadas a cabo por jefes que tienen responsabilidad directa sobre el proyecto o sistema
- Llevadas a cabo por las partes interesadas o por alguien con responsabilidad para tomar decisiones, como un jefe de alto nivel o director
- Comprueban la coherencia con los planes y las desviaciones de los planes
- Comprueban la idoneidad de los procedimientos de gestión
- Evalúan los riesgos del proyecto
- Evalúan el impacto de las acciones y las formas de medir estos impactos
- Generan listas de elementos de acciones, problemas a resolver y decisiones adoptadas.

Las revisiones de gestión de los procesos, tales como las retrospectivas de proyectos (es decir, las lecciones aprendidas), forman parte de las actividades de mejora del proceso.

Los Jefes de Pruebas deben participar y pueden iniciar las revisiones de gestión del progreso de las pruebas.

Por lo general, las auditorías se realizan para demostrar la conformidad con un conjunto definido de criterios, seguramente una norma aplicable, un obstáculo normativo o una obligación contractual. Como tales, las auditorías tienen por objeto ofrecer una evaluación independiente del cumplimiento

de los procesos, la normativa, las normas, etc. A continuación se indican las características clave de las auditorías:

- Realizadas y moderadas por un auditor líder
- Evidencias del cumplimiento recopiladas a través de entrevistas, testigos y documentos de examen
- Los resultados documentados incluyen observaciones, recomendaciones, acciones correctivas y una evaluación final que indica si se ha superado o no la auditoría

3.3 Revisiones de gestión

Las revisiones deberían planificarse para realizarse durante paradas naturales o hitos del proyecto de software. Normalmente, las revisiones deberían realizarse después de definir los requisitos y el diseño, y las revisiones asociadas deben empezar por los objetivos de negocio y continuar hasta el nivel de diseño más bajo. Las revisiones de gestión deberían realizarse en los principales hitos del proyecto, a menudo como parte de una actividad de verificación antes, durante y después de la ejecución de las pruebas y demás fases importantes del proyecto. La estrategia de revisión debe estar coordinada con la política de pruebas y con la estrategia de pruebas general.

Antes de formular un plan de revisiones general a nivel de proyecto, el líder de las revisiones (que podrá ser un Jefe de Pruebas) debe tener en cuenta los siguientes factores:

- Qué debería revisarse (productos y procesos)
- Quién debería participar en las revisiones específicas
- Qué factores de riesgo relevantes deben cubrirse

En una etapa temprana de la fase de planificación del proyecto, el líder de las revisiones debería identificar los elementos a revisar y seleccionar el tipo adecuado de revisión (revisión informal, revisión guiada, revisión técnica o inspección, o una mezcla de dos o tres tipos), así como el nivel de formalidad. En este punto sería recomendable recibir más formación en materia de revisiones. A partir de este momento, puede asignarse un presupuesto (tiempo y recursos) al proceso de revisión. La determinación del presupuesto debería incluir una evaluación de riesgos y un cálculo del retorno sobre la inversión.

El retorno sobre la inversión en las revisiones es la diferencia entre el coste de realizar la revisión y el coste de abordar esos defectos en una etapa posterior (u obviarlos todos) en caso de no haber realizado la revisión. El cálculo del coste de la calidad previsto en la Sección 2.7 puede servir para determinar esta cifra.

Fijar el tiempo óptimo para llevar a cabo las revisiones depende de lo siguiente:

- La disponibilidad de los elementos a revisar en un formato suficientemente definitivo
- La disponibilidad del personal adecuado para la revisión
- El momento en que debería estar disponible la versión definitiva del elemento
- El tiempo necesario para el proceso de revisión de ese elemento en particular

El líder de las revisiones deberá establecer las métricas adecuadas para evaluar la revisión durante la etapa de planificación de las pruebas. Si se utilizan las inspecciones, entonces deberían realizarse inspecciones breves a petición del autor, a medida que se disponga de fragmentos de documentos completos (por ejemplo, requisitos individuales o secciones).

Los objetivos del proceso de revisión deben definirse durante la planificación de las pruebas. Esto incluye llevar a cabo revisiones efectivas y eficientes y alcanzar decisiones de consenso por lo que respecta a la realimentación³ de revisión.

³“realimentación” es la traducción de “feedback”.

A menudo las revisiones de proyecto se refieren al sistema en general, aunque también pueden tener que realizarse en subsistemas e incluso en elementos de software individuales. El número de revisiones, el tipo de revisiones, la organización de las revisiones y las personas implicadas dependerán del tamaño y la complejidad del proyecto, además de los riesgos del producto.

En aras a la eficiencia, los participantes en las revisiones deben disponer del nivel de conocimiento adecuado, tanto técnico como procesal. La exhaustividad y la atención a los detalles son algunas de las aptitudes que los revisores necesitan para llevar a cabo revisiones efectivas. La claridad y la correcta priorización son atributos deseables en los comentarios de las revisiones. La necesidad de conocimiento procesal puede suponer que se requiera cierta formación para asegurar que los revisores entienden sus funciones y responsabilidades en el proceso de revisión.

La planificación de las revisiones debería abordar los riesgos asociados a factores técnicos, factores organizativos y problemas de personas durante la realización de las revisiones. La disponibilidad de revisores con conocimiento técnico suficiente es fundamental para el éxito de toda revisión. Todos los equipos del proyecto deberían participar en las revisiones, que deberían garantizar que cada equipo está comprometido con el éxito del proceso de revisión. La planificación debe garantizar que cada organización está asignando tiempo suficiente a los revisores para preparar y participar en las revisiones en los momentos que corresponda del calendario del proyecto. Asimismo, debería preverse tiempo para la formación técnica o procesal de los revisores. Deberían nombrarse revisores de reserva en previsión de que los revisores clave pasaran a no estar disponibles debido a cambios en los planes de negocio o por motivos personales.

Durante la ejecución efectiva de las revisiones formales, el líder de las revisiones debe asegurarse de que:

- Los participantes proporcionan las mediciones adecuadas que permiten evaluar la eficiencia de la revisión
- Se crean listas de comprobación, que se mantienen para mejorar revisiones futuras
- Se establece la severidad de los defectos y la evaluación de la prioridad para su uso en la gestión de defectos de problemas encontrados durante las revisiones (remítase al Capítulo 4)

Después de cada revisión, el líder de revisiones debería:

- Recopilar las métricas de la revisión y asegurarse de que los problemas identificados están suficientemente resueltos como para cumplir los objetivos de prueba específicos de la revisión
- Utilizar las métricas de la revisión como entradas para establecer el retorno de la inversión (ROI) de las revisiones
- Facilitar comentarios a las partes interesadas relevantes
- Facilitar comentarios a los participantes en la revisión

Para evaluar la efectividad de las revisiones, los Jefes de Pruebas pueden comparar los resultados reales encontrados en pruebas posteriores (es decir, después de las revisiones) con los resultados de los informes de las revisiones. En caso de que un producto de trabajo se revise y se apruebe en base a la revisión, pero más tarde se descubra que este es defectuoso, el líder de las revisiones debería estudiar las formas en las que el proceso de revisión puede haber permitido que se escaparan esos defectos. Entre las causas probables se incluyen problemas con el proceso de revisión (por ejemplo, malos criterios de entrada/salida), una composición inadecuada del equipo de revisión, el uso de herramientas de revisión inadecuadas (listas de comprobación, etc.), formación y experiencia insuficientes del revisor y demasiada poca preparación y tiempo para la reunión de revisión.

La existencia de un patrón de defectos no detectados (especialmente defectos importantes), que se repita en varios proyectos, indica que hay problemas importantes con la realización de las revisiones. En estos casos, el líder de revisiones tiene que revisar el proceso y adoptar las medidas necesarias. También es posible que, por varios motivos, las revisiones pierdan su efectividad con el tiempo. Este

tipo de efecto se pondrá de manifiesto en retrospectivas del proyecto a través de una menor efectividad de las revisiones para detectar defectos. Llegados a este punto, el líder de revisiones también debe investigar y arreglar las causas. En cualquier caso, las métricas de revisión no deberían servir para castigar o premiar a revisores o autores individuales, sino que deberían centrarse en el propio proceso de revisión.

3.4 Métricas para las revisiones

Los líderes de las revisiones (que, según lo citado en las secciones anteriores, también pueden ser Jefes de Pruebas) deben asegurarse de que hay métricas disponibles para:

- Evaluar la calidad del elemento revisado
- Evaluar el coste de llevar a cabo la revisión
- Evaluar el beneficio posterior de haber llevado a cabo la revisión

Los líderes de las revisiones pueden utilizar las mediciones para establecer el retorno sobre la inversión y la eficiencia de las revisiones. Estas métricas también pueden utilizarse para elaborar informes y para actividades asociadas a la mejora del proceso.

Por cada producto de trabajo revisado, pueden medirse las siguientes métricas y reportarse para la evaluación del producto:

- Dimensiones del producto de trabajo (páginas, líneas de código, etc.)
- Tiempo de preparación (previo a la revisión)
- Tiempo para llevar a cabo la revisión
- Tiempo para realizar adaptaciones para corregir defectos
- Duración del proceso de revisión
- Número de defectos encontrados y su severidad
- Identificación de clústeres de defectos dentro del producto de trabajo (es decir, áreas que tienen una mayor densidad de defectos)
- Tipo de revisión (revisión informal, revisión guiada, revisión técnica o inspección)
- Densidad media de los defectos (por ejemplo, defectos por página o por mil líneas de código)
- Defectos residuales estimados (o densidad de los defectos residuales)

Por cada revisión, pueden medirse y reportarse las siguientes métricas para la evaluación del proceso:

- Efectividad de la detección de defectos (teniendo en cuenta los defectos encontrados más tarde en el ciclo de vida)
- Mejora del esfuerzo y los tiempos del proceso de revisión
- Porcentaje de cobertura de los productos de trabajo previstos
- Tipos de defectos encontrados y su severidad
- Encuestas de los participantes sobre la efectividad y la eficiencia del proceso de revisión
- Métricas sobre el coste de la calidad para los defectos de revisión frente a defectos de pruebas dinámicas y defectos de producción
- Correlación de la efectividad de la revisión (tipo de revisión frente a la efectividad de detección de defectos)
- Número de revisores
- Defectos encontrados por hora de trabajo dedicada
- Tiempo estimado que se ha ahorrado en el proyecto
- Esfuerzo medio de defectos (es decir, la detección total y el tiempo de arreglo dividido por el número de defectos)

Además, las métricas citadas para la evaluación de los productos también son útiles en la evaluación de procesos.

3.5 Gestión de revisiones formales

El Programa de Estudios Fundamentos del ISTQB describe las distintas fases de una revisión formal: planificación, puesta en marcha, preparación individual, reunión de revisión, adaptaciones y seguimiento. Para implementar correctamente revisiones formales, los líderes de las revisiones tienen que asegurarse de que se siguen todos los pasos del proceso de revisión.

Entre las características de las revisiones formales se encuentran:

- Criterios de entrada y de salida definidos
- Listas de comprobación a utilizar por los revisores
- Entregables como informes, hojas de evaluación u otras hojas resumen de la revisión
- Métricas para elaborar informes sobre la eficacia, la eficiencia y el progreso de la revisión

Antes de iniciar una revisión formal, el líder de las revisiones debería confirmar el cumplimiento de los prerequisites de la revisión (según lo previsto en el procedimiento o en la lista de criterios de entrada).

De no cumplirse las condiciones prerrequeridas para la revisión formal, el líder de las revisiones puede proponer una de las siguientes opciones a sus superiores para tomar la decisión final:

- Redefinición de la revisión con objetivos revisados
- Acciones correctivas necesarias para poder proceder a la revisión
- Aplazamiento de la revisión

Como parte del control de una revisión formal, estas revisiones se monitorizan en el contexto del programa general (alto nivel), y están asociadas a las actividades para el aseguramiento de la calidad del proyecto. El control de las revisiones formales incluye dar realimentación a través de las métricas del producto y del proceso.

4. Gestión de defectos – 150 minutos

Palabras clave

anomalía, defecto, comité de triaje de defectos, fallo, resultado de falso-negativo, resultado de falso-positivo, contención en fase, prioridad, causa raíz, severidad

Objetivos de aprendizaje de Gestión de defectos

4.2 El ciclo de vida de los defectos y el ciclo de vida de desarrollo de software

TM-4.2.1 (K3) Desarrollar un proceso de gestión de defectos para una organización de pruebas, que incluya el flujo de trabajo del informe de defectos y sirva para monitorizar y controlar los defectos de un proyecto durante el ciclo de vida de las pruebas

TM-4.2.2 (K2) Explicar el proceso y los participantes requeridos de una gestión efectiva de los defectos

4.3 Información de los informes de defectos

TM-4.3.1 (K3) Definir los datos y la información de clasificación que debería recopilarse durante el proceso de gestión de defectos

4.4 Evaluar la capacidad del proceso con información del informe de defectos

TM-4.4.1 (K2) Explicar cómo pueden utilizarse las estadísticas de los informes de defectos para evaluar la capacidad de los procesos de pruebas y desarrollo de software

4.1 Introducción

El proceso de gestión de defectos de una organización y la herramienta empleada para gestionar este trabajo son fundamentales no solo para el equipo de pruebas sino también para todos los equipos implicados en el desarrollo de software. La información recopilada gracias a una gestión efectiva de los defectos permite al Jefe de Pruebas y a otras partes interesadas del proyecto disponer de información sobre el estado de un proyecto durante todo el ciclo de vida de desarrollo, y gracias a la recopilación y al análisis de datos a lo largo del tiempo, puede ayudar a localizar áreas de mejora potencial para los procesos de pruebas y desarrollo.

Además de comprender el ciclo de vida general de los defectos y saber cómo este se utiliza para monitorizar y controlar los procesos de pruebas y desarrollo de software, el Jefe de Pruebas debe conocer qué datos es fundamental capturar y debe ser un defensor del uso correcto tanto del proceso como de la herramienta de gestión de defectos seleccionada.

4.2 El ciclo de vida de los defectos y el ciclo de vida de desarrollo de software

Según lo explicado en el programa de estudio Fundamentos, los defectos se introducen cuando una persona comete un error durante la creación de un producto de trabajo. Este producto de trabajo puede ser una especificación de requisitos, una historia de usuario, un documento técnico, un caso de pruebas, el código del programa o cualquier otro producto de trabajo creado durante un proceso de mantenimiento o de desarrollo de software.

Los defectos pueden introducirse en cualquier momento en el ciclo de vida de desarrollo de software y en cualquier producto de trabajo relacionado con el software. Por lo tanto, cada fase del ciclo de vida de desarrollo de software debe disponer de métodos para detectar y eliminar los posibles fallos. Por ejemplo, las técnicas de pruebas estáticas (es decir, revisiones y análisis estáticos) pueden utilizarse en las especificaciones de diseño, especificaciones de requisitos y código antes de entregar esos productos de trabajo a actividades posteriores. Cuanto antes se detecte y elimine un defecto, menor será el coste general de calidad para el sistema; el coste de la calidad de un nivel dado de defectos se minimiza si cada defecto se elimina en la misma fase en la que se ha introducido (es decir, cuando el proceso de software alcanza una contención en fase perfecta). Además, según se explica en el Programa de Estudio de Nivel Fundamentos, las pruebas estáticas encuentran defectos directamente, en lugar de encontrar fallos, lo que hace que el coste de eliminación de los defectos sea menor porque no hay que realizar actividades de depuración para aislar el defecto.

Durante las actividades de pruebas dinámicas, como las pruebas unitarias, las pruebas de integración y las pruebas de sistema, la presencia de un defecto se revela cuando provoca un fallo, lo que resulta en una discrepancia entre los resultados reales y los resultados esperados de una prueba (es decir, una anomalía). En algunos casos, se produce un resultado de falso negativo cuando el probador no observa la anomalía. Si el probador observa una anomalía, entonces estamos ante una situación que requiere una mayor investigación. Esta investigación se inicia con la elaboración de un informe de defectos.

En el desarrollo guiado por pruebas, las pruebas unitarias automatizadas se utilizan como una forma de especificaciones de diseño ejecutables. A medida que se desarrolla el código, este se ejercita utilizando estas pruebas. Hasta que se haya completado el desarrollo de la unidad, alguna o todas las pruebas fallarán. Por lo tanto, el fallo de este tipo de pruebas no constituye un defecto, y normalmente no es objeto de seguimiento.

4.2.1 Flujo de trabajo de defectos y estados

La mayoría de las organizaciones de pruebas utilizan una herramienta para gestionar los informes de defectos durante el ciclo de vida de los defectos. Un informe de defectos normalmente sigue un flujo de trabajo y pasa por una secuencia de estados a medida que avanza en el ciclo de vida de los defectos. En la mayor parte de estos estados, solo un participante del ciclo de vida de los defectos es propietario del informe y tiene la responsabilidad de llevar a cabo una tarea que, una vez finalizada, hará que el informe de defectos pase al siguiente estado (y se asigne a la siguiente parte responsable). En los estados terminales, como cuando un informe de defectos se cierra (normalmente porque el defecto subyacente se ha resuelto y dicha resolución ha sido comprobada mediante una prueba de confirmación), se anula (normalmente por que el informe de defectos no es válido), es irreproducible (normalmente porque la anomalía ya no puede observarse) o se difiere (normalmente porque la anomalía se refiere a un defecto real, pero dicho defecto no va a arreglarse durante el proyecto), el informe no tiene un propietarios, ya que no es preciso realizar más acciones.

En el caso de defectos descubiertos por los probadores durante las pruebas, hay tres estados en particular sobre los que actúa el equipo de pruebas:

- El estado inicial
 - En este estado, uno o más probadores recopilan la información necesaria para la persona responsable para resolver el defecto para reproducir la anomalía (remítase a la Sección 4.3 para saber más sobre la información a incluir en el informe de defectos).
 - También se le puede denominar estado “abierto” o “nuevo”.
- El estado devuelto
 - En este estado, el receptor del informe ha rechazado el informe o está pidiendo al probador que facilite más información. Este estado puede indicar un déficit en la recopilación inicial de información o de las propias pruebas. Los Jefes de Pruebas deberían controlar las tasas excesivas de devoluciones. Los probadores tienen que facilitar la información adicional, o confirmar que efectivamente el informe debe rechazarse.
 - También se le puede denominar estado “rechazado” o “de aclaración”.
- El estado de prueba de confirmación
 - En este estado, el probador ejecutará una prueba de confirmación (a menudo a continuación de los pasos para reproducir el fallo a partir del propio informe de defectos) para establecer si la resolución ha resuelto efectivamente el problema. Si la prueba de confirmación indica que el defecto se ha reparado, el probador debe cerrar el informe. Si la prueba de confirmación indica que el defecto no se ha reparado, el probador debe volver a abrir el informe, haciendo que se reasigne a su propietario anterior, que podrá entonces completar el trabajo necesario para reparar el defecto.
 - También se le puede denominar estado “resuelto” o “de verificación”.

4.2.2 Gestión de informes de defectos no válidos y duplicados

En algunos casos, una anomalía no se produce como el síntoma de un defecto, sino como la consecuencia de un problema con el entorno de pruebas, los datos de prueba o cualquier otro elemento de los elementos de soporte de pruebas, o un malentendido del propio probador. Si el probador abre un informe de defectos que más tarde resulta no referirse a un defecto en el producto de trabajo objeto de las pruebas, estaremos ante un resultado de falso positivo. Este tipo de informes normalmente se anulan o cierran como informes de defectos no válidos. Además, en algunos casos un defecto puede mostrar distintos síntomas que el probador o los probadores pueden considerar como totalmente independientes. Si se presentan dos o más informes de defectos que más tarde resultan referirse a la misma causa raíz, normalmente se conserva uno de los informes de defectos mientras que el resto se cierra en calidad de informes duplicados.

Si bien los informes de defectos no válidos y duplicados representan un cierto nivel de ineficiencia, es inevitable que existan algunos y el Jefe de Pruebas debería aceptarlos como tal. Cuando los jefes intentan eliminar todos los informes de defectos no válidos y duplicados, el número de falsos negativos normalmente aumenta, ya que se está desanimando a los probadores a presentar informes de defectos. Esto reduce la efectividad de la detección de defectos de la organización de pruebas, que en la mayoría de los casos está relacionada con un objetivo clave de la organización de pruebas.

4.2.3 Gestión de defectos transversal

Si bien la organización de pruebas y el Jefe de Pruebas normalmente son propietarios del proceso general de gestión de defectos y de la herramienta de gestión de defectos, por lo general un equipo transversal es responsable de gestionar los defectos reportados de un proyecto dado. Además del Jefe de Pruebas, los participantes del comité de gestión de defectos (o triaje de defectos) normalmente incluyen personal de desarrollo, gestión de proyecto, gestión de producto y otras partes interesadas que tienen un interés en el software en desarrollo.

A medida que se descubren y se introducen las anomalías en la herramienta de gestión de defectos, el comité de gestión de defectos debería reunirse para establecer si cada informe de defectos representa un defecto válido, y si este debería solucionarse o diferirse. Esta decisión requiere que el comité de gestión de defectos tenga en cuenta las ventajas, los riesgos y los costes asociados a la resolución o no resolución del defecto. Si el defecto debe resolverse, el equipo debería establecer la prioridad de resolver el defecto por lo que respecta a otras tareas del proyecto. Para determinar el grado de importancia de un defecto, se debe consultar al Jefe de Pruebas y al equipo de pruebas, que siempre deberán facilitar información objetiva.

No es aconsejable reemplazar una buena comunicación por una herramienta de seguimiento de defectos, ni utilizar las reuniones del comité de gestión de defectos como excusa para evitar el uso efectivo de una buena herramienta de seguimiento de defectos. La comunicación, una herramienta de soporte adecuada, un ciclo de vida de los defectos bien definido y un comité de gestión de defectos comprometido son elementos necesarios para una gestión de los defectos efectiva y eficiente.

4.3 Información de los informes de defectos

Cuando se detecta un defecto (como parte de las pruebas estáticas), o se observa un fallo (como parte de las pruebas dinámicas), la persona implicada debería recopilar los datos e incluirlos en el informe de defectos. Esta información debería ser suficiente para tres fines:

- Gestión del informe durante el ciclo de vida de los defectos
- Evaluación del estado del proyecto, especialmente en términos de calidad del producto y progreso de las pruebas
- Evaluación de la capacidad del proceso (según lo establecido en la Sección 4.4 a continuación)

Los datos necesarios para la gestión de los informes de defectos y el estado del proyecto pueden variar en función de cuándo se detecta el defecto en el ciclo de vida, siendo la información requerida menor en etapas anteriores (por ejemplo, revisiones de requisitos y pruebas unitarias). No obstante, la información básica recopilada debería ser coherente durante todo el ciclo de vida e, idealmente, en todos los proyectos para permitir una comparación significativa de los datos de defectos del proceso durante el proyecto y en todos los proyectos.

La recopilación de los datos de defectos puede ayudar a monitorizar el progreso de las pruebas, el control y la evaluación de los criterios de salida. Así por ejemplo, la información de defectos debería incluir análisis de la densidad de los defectos, análisis de tendencia de los defectos detectados y resueltos, tiempo medio desde la detección del defecto hasta su resolución, e intensidad de los fallos (por ejemplo, análisis de tiempo medio entre fallos, TMEF).

Entre los datos de defectos a recopilar se encuentran:

- El nombre de la persona que descubrió el defecto
- La función de la persona (por ejemplo, usuario final, analista de negocio, desarrollador, persona de soporte técnico)
- El tipo de pruebas que se están llevando a cabo (por ejemplo, pruebas de usabilidad, pruebas de rendimiento, pruebas de regresión)
- Un resumen del problema
- Una descripción detallada del problema
- Los pasos para reproducir el fallo (en el caso de un defecto), junto con los resultados reales y esperados (subrayando la anomalía), incluyendo pantallazos, volcados de bases de datos y registros cuando proceda
- La fase del ciclo de vida de introducción, detección y eliminación del defecto, incluyendo el nivel de prueba si procede
- El producto de trabajo en el que ha surgido el defecto
- La severidad del impacto en el sistema y/o las partes interesadas del producto (normalmente establecida por el comportamiento técnico del sistema)
- La prioridad para solucionar el problema (normalmente establecida por el impacto de negocio del fallo)
- El subsistema o el componente en el que se encuentra el defecto (para el análisis de clúster de defectos)
- La actividad del proyecto que estaba produciéndose cuando se detectó el problema
- El método de identificación que reveló el problema (por ejemplo, revisión, análisis estático, pruebas dinámicas, uso de producción)
- El tipo de defecto (que normalmente corresponde a una taxonomía de defectos cuando se utilicen)
- La característica de calidad afectada por el defecto
- El entorno de pruebas en el que se ha observado el defecto (para las pruebas dinámicas)
- El proyecto y el producto en el que existe el problema
- El actual propietario; es decir, la persona actualmente asignada para trabajar en el problema, asumiendo que el informe no está en un estado final
- El estado actual del informe (normalmente gestionarlo por la herramienta de seguimiento de defectos como parte del ciclo de vida)
- Los productos de trabajo específicos (por ejemplo, elementos de prueba y sus números de entrega) en los que se ha observado el problema, junto con los productos de trabajo específicos en el que el problema se resolvió en última instancia
- El impacto en el proyecto y en los intereses de las partes interesadas del producto
- Las conclusiones, recomendaciones y aprobación para las medidas adoptadas o no adoptadas para resolver el problema
- Riesgos, costes, oportunidades y ventajas asociadas a la resolución o no resolución del defecto
- Las fechas en las que se produjeron varias transiciones del ciclo de vida de los defectos, los propietarios del informe basado en cada transición y las medidas adoptadas por los miembros del equipo de proyecto para aislar, reparar y comprobar la solución del defecto
- Una descripción de cómo se ha resuelto el defecto en última instancia y recomendaciones para probar el arreglo (si el defecto se resolvió mediante un cambio en el software)
- Otras referencias, tales como la prueba que reveló el defecto y el riesgo, requisito o cualquier otro elemento de la base de prueba asociado con el defecto (para las pruebas dinámicas)

Existen varias normas y documentos, tales como la ISO 9126 [ISO9126] (sustituida por la ISO 25000), la IEEE 829 [IEEE829], IEEE 1044 [IEEE1044], y la Clasificación Ortogonal de de Defectos para ayudar al Jefe de Pruebas a determinar qué información recopilar para elaborar los informes de defectos.

Independientemente de la información específica que se establezca como necesaria para los informes de defectos, es fundamental que los probadores introduzcan información completa, concisa, exacta, objetiva relevante y oportuna. Incluso cuando la intervención manual y la comunicación cara a cara superan problemas con datos de informes de defectos en términos de resolver un defecto individual, los problemas con los datos de los informes de defectos pueden plantear obstáculos insalvables para la correcta evaluación del estado del proyecto, el progreso de las pruebas y la capacidad del proceso.

4.4 Evaluar la capacidad del proceso con información del informe de defectos

Tal y como se ha expuesto en el Capítulo 2, los informes de defectos pueden ser útiles para el seguimiento y la elaboración de informes sobre los estados del proyecto. Mientras que las implicaciones procesales de las métricas se abordan principalmente en el programa de estudio de Gestión de Pruebas de Nivel Experto [ISTQB ETM SYL], en el Nivel Avanzado, los Jefes de Pruebas deberían conocer qué significan los informes de defectos en términos de evaluar la capacidad de las pruebas y los procesos de desarrollo de software.

Además de la información de seguimiento del progreso de las pruebas citada en el Capítulo 2 y en la Sección 4.3, la información de los defectos tiene que incluir iniciativas para la mejora de los procesos. Entre otros ejemplos, destacan:

- El uso de la fase de introducción, detección y eliminación de información, fase por fase, para evaluar la contención en fase y sugerir formas de mejorar la eficacia de la detección de defectos en cada fase
- El uso de la fase de introducción de información para hacer un análisis de Pareto de las fases en las que se han introducido el mayor número de defectos, para permitir que las mejoras objetivo reduzcan el número total de defectos
- El uso de información sobre las causas raíz de los defectos para establecer los motivos subyacentes de la introducción de los defectos, para permitir mejoras de proceso que reduzcan el número total de defectos
- El uso de la fase de introducción, detección y eliminación de información para llevar a cabo análisis de coste de la calidad, para minimizar el coste asociado a los defectos
- El uso de la información de los componentes de defectos para llevar a cabo análisis de clústeres de defectos, para entender mejor los riesgos técnicos (para las pruebas basadas en riesgos) y para permitir la reingeniería de componentes problemáticos

El uso de métricas para evaluar la efectividad y la eficiencia del proceso de pruebas se aborda en el programa de estudio de Gestión de Pruebas de Nivel Experto [ISTQB ETM SYL].

En algunos casos, los equipos pueden optar por no seguir los defectos detectados durante alguno o todos los ciclos de vida de desarrollo del software. Si bien esto a menudo se realiza en aras a la eficiencia y para reducir los gastos generales del proceso, en realidad reduce mucho la visibilidad de las capacidades del proceso de desarrollo de pruebas y de software. Esto hace que las mejoras sugeridas más arriba sean difíciles de implementar debido a la ausencia de datos fiables.

5. Mejora del proceso de pruebas - 135 minutos

Palabras clave

Capability Maturity Model Integration (CMMI), Critical Testing Processes (CTP), Systematic Test and Evaluation Process (STEP), Test Maturity Model integration (TMMi), TPI Next

Objetivos de aprendizaje para mejorar el proceso de pruebas

5.2 Proceso de mejora de pruebas

TM-5.2.1 (K2) Explicar mediante ejemplos por qué es importante mejorar el proceso de pruebas

5.3 Mejorar el proceso de pruebas

TM-5.3.1 (K3) Definir un plan de mejoras del proceso de pruebas siguiendo el modelo IDEAL

5.4 Mejorar el proceso de pruebas con TMMi

TM-5.4.1 (K2) Resumir los antecedentes, el alcance y los objetivos del modelo TMMi para la mejora del proceso de pruebas

5.5 Mejorar el proceso de pruebas con TPI Next

TM-5.5.1 (K2) Resumir los antecedentes, el alcance y los objetivos del modelo TPI Next para la mejora del proceso de pruebas

5.6 Mejorar el proceso de pruebas con CTP

TM-5.6.1 (K2) Resumir los antecedentes, el alcance y los objetivos del modelo CTP para la mejora del proceso de pruebas

5.7 Mejorar el proceso de pruebas con STEP

TM-5.7.1 (K2) Resumir los antecedentes, el alcance y los objetivos del modelo STEP para la mejora del proceso de pruebas

5.1 Introducción

Una vez implementado, el proceso de pruebas general de una organización debería ser objeto de mejoras continuas. En el capítulo se tratan las cuestiones genéricas de mejora, seguidas de una introducción a algunos modelos específicos que pueden utilizarse para mejorar los procesos de pruebas. Los Jefes de Pruebas deberían asumir que serán la fuerza impulsora de los cambios y mejoras en los procesos de pruebas, por lo que deberían estar familiarizados con las técnicas aceptadas en el sector que se abordan en este capítulo. Para más información sobre los procesos de mejora de las pruebas, remítase al programa de estudio de Nivel Experto “Mejora del proceso de pruebas”.

5.2 Proceso de mejora de pruebas

Al igual que las organizaciones utilizan las pruebas para mejorar el software, pueden seleccionarse y utilizarse técnicas de mejora de procesos para mejorar el proceso de desarrollo de software y los entregables de software resultantes. La mejora de procesos también se puede aplicar a los procesos de prueba. Existen diferentes formas y métodos para mejorar las pruebas de software y los sistemas que contienen software. Estos métodos persiguen la mejora de los procesos, y por consiguiente, de los entregables, mediante la determinación de pautas y áreas para la mejora.

A menudo las pruebas representan una parte importante del coste total del proyecto. Sin embargo, en los diferentes modelos de mejora de procesos de software se presta muy poca atención a los procesos de pruebas, como puede ser el CMMI® (ver a continuación para más detalle).

Los modelos de mejora de las pruebas tales como el Test Maturity Model integration (TMMi®), Systematic Test and Evaluation Process (STEP), Critical Testing Processes (CTP) y el TPI Next® se han desarrollado para abordar la escasa atención que se presta a las pruebas en la mayoría de los modelos de mejora de procesos de software. Utilizados correctamente, estos modelos pueden ofrecer un grado de métrica de organización cruzada que puede ser útil para establecer comparaciones de referencia.

Los modelos mencionados en este programa de estudio no pretenden ser una recomendación de uso, sino que se presentan para ofrecer una idea representativa de cómo funcionan estos modelos y de lo que incluyen.

5.2.1 Introducción al Proceso de Mejora

Los procesos de mejora son importantes para los procesos de desarrollo del software y para los procesos de prueba. Aprender de los fallos que uno mismo comete permite mejorar los procesos que utilizan las organizaciones para desarrollar y probar un software. El ciclo de mejora Deming: Planificar, Hacer, Comprobar y Actuar, ha sido utilizado durante varias décadas, y sigue siendo útil cuando los probadores necesitan mejorar los procesos en uso hoy en día.

Una de las premisas del proceso de mejora es la creencia de que la calidad de un sistema está directamente relacionada con la calidad del proceso utilizado para desarrollar el software. La mejora de la calidad en la industria del software reduce la necesidad de otros recursos de mantenimiento de software y ofrece, por lo tanto, más tiempo para crear más y mejores soluciones en el futuro. Los modelos de procesos establecen un punto de partida para la mejora, y en base a estos, se miden los procesos de capacidades de la organización contra el modelo. Los modelos también establecen un marco de referencia para mejorar los procesos de la organización, basándose en el resultado de una evaluación.

La evaluación del proceso conduce a la determinación de la capacidad del proceso, la cual tiene como resultado una mejora del proceso. Esto puede a su vez requerir una evaluación del proceso posterior para cuantificar los efectos de la mejora.

5.2.2 Tipos de mejora de procesos

El uso de modelos de evaluación es un método común que asegura un enfoque normalizado para mejorar los procesos de prueba mediante prácticas probadas y confiables.

Los modelos para la mejora de procesos pueden dividirse en dos tipos:

1. El modelo de referencia de proceso, que ofrece una medición de la madurez como parte de su evaluación para valorar la capacidad de una organización frente al modelo, para evaluar la organización dentro del marco de trabajo y para proporcionar una hoja de ruta para mejorar el proceso.
2. El modelo de referencia de contenido, que ofrece evaluaciones guiadas por el negocio de las oportunidades de una organización para mejorar, incluyendo, en algunos casos, evaluaciones comparativas frente a las medias del sector empleando mediciones objetivas. Esta evaluación puede utilizarse para crear una hoja de ruta para mejorar el proceso.

La mejora de los procesos de prueba también puede prescindir de esos modelos y adoptar, por ejemplo, enfoques analíticos y reuniones retrospectivas.

5.3 Mejora del proceso de pruebas

La industria informática ha comenzado a trabajar con modelos de mejora de procesos de prueba para alcanzar mayores niveles de madurez y profesionalidad. Los modelos estándar del sector están contribuyendo al desarrollo de métricas transversales de la organización y medidas que pueden utilizarse a efectos de comparación. Como respuesta a las necesidades de mejora en la industria de las pruebas, se han materializado varios conjuntos de procesos recomendados. Entre ellos destacan STEP, TMMi, TPI Next y CTP. Los modelos dirigidos, como el TMMi y el CMMI, ofrecen normas comparativas entre diferentes empresas y organizaciones. Los modelos continuos, como CTP, STEP y TPI Next, permiten a una organización abordar los asuntos de máxima prioridad con mayor libertad en el orden de implementación. Todos ellos se analizarán más adelante en esta sección.

Cualquiera de estos modelos permite a la organización determinar en qué punto se encuentra por lo que respecta a los procesos de prueba abiertos. Una vez realizada una evaluación, los modelos TMMi y TPI Next sugieren una hoja de ruta para mejorar el proceso de pruebas. Alternativamente, los modelos STEP y CTP facilitan a la organización los medios para determinar de dónde procederá el mayor retorno sobre la inversión de la mejora del proceso, dejando que la organización sea la que seleccione la hoja de ruta apropiada.

Una vez tomada la decisión de revisar y mejorar los procesos de prueba, los pasos para la implementación de mejoras de proceso que deben seguirse para ello podrían ser los previstos en el modelo IDEALSM [IDEAL96]:

- Iniciar (*Initiating*) el proceso de mejora
- Diagnosticar (*Diagnosing*) la situación actual
- Establecer (*Establishing*) un plan de mejora del proceso de pruebas
- Actuar (*Acting*) para llevar a cabo las mejoras
- Aprender (*Learning*) del programa de mejoras

Iniciar el proceso de mejora

Antes de poner en marcha las tareas de mejora de los procesos, las partes interesadas establecen los objetivos, las metas, el alcance y la cobertura de las mejoras de los procesos. El modelo para la mejora de procesos también se elige en este punto. El modelo puede elegirse a partir de las opciones disponibles públicamente (como CTP, STEP, TMMi y TPI Next) o desarrollarse internamente. Además, deben establecerse los criterios de éxito y el método con el que estos van a medirse durante la actividad de mejora.

Diagnosticar la situación actual

Se adopta el enfoque de evaluación acordado y se crea un informe de evaluación de pruebas con una valoración de las prácticas de pruebas actuales y una lista de posibles mejoras de los procesos.

Establecer un plan de mejora del proceso de pruebas

Se establece un orden de prioridad en la lista de posibles mejoras del proceso. El orden de prioridad podría atender a razones de retorno sobre la inversión, riesgos, alineación con la estrategia organizativa y/o beneficios cuantitativos o cualitativos medibles. Una vez establecido el orden de prioridad, se desarrolla un plan de entrega de las mejoras.

Actuar para llevar a cabo las mejoras

Se aplica el plan de mejoras del proceso de pruebas para llevar a cabo las mejoras. Esto podría incluir cualquier acción de formación u orientación que fuera necesaria, el pilotaje de los procesos y, en última instancia, su despliegue total.

Aprender del programa de mejoras

Una vez desplegadas totalmente las mejoras del proceso, es fundamental comprobar qué ventajas (de las previstas antes, además de las posibles ventajas imprevistas) se han obtenido. También es importante comprobar qué criterios de éxito para mejorar el proceso se han cumplido.

Dependiendo del modelo de proceso aplicado, en esta fase es donde empieza la monitorización del siguiente nivel de madurez y se toma una decisión sobre si se debe volver a empezar el proceso de mejora o si es el momento de detener la actividad.

5.4 Mejorar el proceso de pruebas con TMMi

El Testing Maturity Model (TMMi) se compone de cinco niveles de madurez y pretende complementar al CMMI. Cada uno de los niveles de madurez contiene áreas de proceso definidas que deben estar al 85% de su consecución a través del logro de objetivos específicos y genéricos para que la organización pueda pasar al siguiente nivel.

Los niveles de madurez del TMMi son:

- Nivel 1: Inicial
El nivel inicial representa un estado donde no hay un proceso de pruebas formalmente documentado ni estructurado. Las pruebas se realizan ad hoc después de la codificación y las pruebas se consideran un sinónimo de la depuración. En este nivel, se entiende que el objetivo de las pruebas es demostrar que el software funciona.
- Nivel 2: Gestionado
El segundo nivel se alcanza cuando los procesos de pruebas están claramente separados de la depuración. Puede alcanzarse mediante el establecimiento de objetivos y políticas de pruebas, la introducción de los pasos incluidos en un proceso de pruebas fundamental (por ejemplo, un plan de pruebas) y la implementación de técnicas y métodos básicos de pruebas.
- Nivel 3: Definido
El tercer nivel se alcanza cuando un proceso de prueba se integra en el desarrollo del ciclo de vida del software, y se documenta con estándares, procedimientos y métodos formales. Se

realizan revisiones y debería haber una función de prueba de software definida que se pueda controlar y monitorizar.

- Nivel 4: Medido
El cuarto nivel se alcanza cuando el proceso de pruebas puede ser medido y gestionado de manera efectiva en un nivel organizativo en beneficio de proyectos específicos.
- Nivel 5: Optimizado
El último nivel representa un estado de madurez del proceso de pruebas en el cual la información obtenida en tal proceso puede ser utilizada para evitar defectos, y la optimización del proceso establecido es prioritaria.

Para más información sobre TMMi, remítase a [vanVeenendaal11] y [www.tmmi.org].

5.5 Mejorar el proceso de pruebas con TPI Next

El modelo TPI Next define 16 áreas clave, cada una de las cuales cubre un aspecto específico del proceso de pruebas, tales como la estrategia de pruebas, las métricas, las herramientas de pruebas y el entorno de pruebas.

En este modelo hay cuatro niveles de madurez:

- Inicial
- Controlado
- Eficiente
- Optimizado

En cada uno de los niveles de madurez se establecen puntos de comprobación específicos para evaluar cada área clave. Las conclusiones se resumen y consultan a través de una matriz de madurez que abarca todas las áreas clave. La definición de los objetivos de mejora y su implementación puede adaptarse conforme a las necesidades y a la capacidad de la organización de pruebas.

El enfoque genérico hace que TPI Next sea independiente de cualquier modelo para la mejora de procesos de software. Cubre tanto aspectos de ingeniería de pruebas como soporte para la toma de decisiones de la dirección [deVries09].

Para más información sobre TPI Next, visite [www.tpinext.com].

5.6 Mejorar el proceso de pruebas con CTP

La premisa básica del modelo de evaluación Critical Testing Process (CTP) es que hay determinados procesos de prueba críticos. Estos procesos críticos, si se llevan a cabo correctamente, contribuirán al éxito de los equipos de prueba. En cambio, si estas actividades se llevan a cabo de forma deficiente, incluso los probadores individuales y Jefes de Pruebas más brillantes tendrán muchas probabilidades de fracasar. Este modelo identifica doce procesos de prueba críticos. El CTP es fundamentalmente un modelo de referencia de contenido.

El modelo CTP es un enfoque sensible al contexto que permite adaptar el modelo, incluyendo:

- La identificación de los retos específicos
- El reconocimiento de atributos de buenos procesos
- La selección del orden e importancia de la implementación de las mejoras del proceso

El modelo CTP es adaptable al contexto de los modelos de desarrollo del ciclo de vida del software.

Además de las entrevistas con participantes, el modelo CTP incluye el uso de métricas para comparar organizaciones con los valores medios y las mejores prácticas del sector.

Para más información sobre CTP, remítase a [Black03].

5.7 Mejorar el proceso de pruebas con STEP

STEP (Systematic Test and Evaluation Process), al igual que CTP y a diferencia de TMMi y TPI Next, no requiere que las mejoras se produzcan en un orden específico.

STEP es básicamente un modelo de referencia de contenidos basado en la idea de que las pruebas son una actividad del ciclo de vida que empiezan cuando se formulan los requisitos y continúan hasta la retirada del sistema. La metodología STEP sigue la máxima “primero prueba, después programa”, empleando una estrategia de pruebas basada en requisitos para garantizar que la creación temprana de los casos de prueba valide la especificación de los requisitos antes que el diseño y la codificación.

Entre las premisas básicas de la metodología se encuentran:

- Una estrategia de pruebas basada en requisitos
- Las pruebas se realizan al principio del ciclo de vida
- Las pruebas se emplean como requisitos y modelos de uso
- El diseño de los productos de soporte de pruebas lidera el diseño del software
- Los defectos se detectan antes o se evitan en conjunto
- Los defectos se analizan de forma sistemática
- Los probadores y desarrolladores trabajan conjuntamente

En algunos casos, el modelo de evaluación STEP se combina con el modelo de madurez TPI Next.

Para más información sobre STEP, remítase a [Craig02].

6. Herramientas y automatización de pruebas - 135 min

Palabras clave

herramienta de código abierto, herramienta personalizada

Objetivos de aprendizaje de Herramientas y automatización de pruebas

6.2 Selección de herramientas

- TM-6.2.1 (K2) Describir los problemas de gestión a la hora de seleccionar una herramienta de código abierto
- TM-6.2.2 (K2) Describir los problemas de gestión a la hora de seleccionar una herramienta a medida
- TM-6.2.3 (K4) Evaluar una situación dada para diseñar un plan para seleccionar herramientas que incluya los riesgos, los costes y las ventajas

6.3 Ciclo de vida de las herramientas

- TM-6.3.1 (K2) Explicar las distintas fases del ciclo de vida de una herramienta

6.4 Métricas de herramientas

- TM-6.4.1 (K2) Describir cómo puede mejorarse la recopilación y evaluación de las métricas mediante el uso de herramientas

6.1 Introducción

Esta sección amplía los contenidos del programa de estudio de Nivel Fundamentos y cubre una serie de conceptos generales que el Jefe de Pruebas debe tener en cuenta por lo que respecta a las herramientas y a la automatización.

6.2 Selección de herramientas

Hay una serie de aspectos que el Jefe de Pruebas debe tener en cuenta a la hora de seleccionar las herramientas de pruebas.

Históricamente, la opción más común es comprar una herramienta a un proveedor comercial. En algunos casos, esta es también la única opción posible. No obstante, existen otras posibilidades, como las herramientas de código abierto y las herramientas a medida, que también son opciones viables.

Independientemente del tipo de herramienta, el Jefe de Pruebas debe estudiar detenidamente el coste total que supone su posesión durante la vida esperada de la herramienta mediante un análisis de rentabilidad. Este tema se cubre en la sección sobre retorno sobre la inversión (ROI) a continuación.

6.2.1 Herramientas de código abierto

Las herramientas de código abierto están disponibles para casi cualquier faceta del proceso de pruebas, desde la gestión de los casos de prueba hasta el seguimiento de defectos pasando por la automatización de los casos de prueba, entre otras. Una característica importante de las herramientas de código abierto es que mientras que la propia herramienta normalmente no tiene un alto coste inicial de compra, es posible que la herramienta no disponga de ningún soporte formal. Muchas herramientas de código abierto, sin embargo, sí tienen un seguimiento dedicado que está dispuesto a facilitar soporte no tradicional o informal a los usuarios.

Asimismo, muchas herramientas de código abierto se crearon inicialmente para resolver un problema específico o para abordar una cuestión en particular, por lo que es posible que la herramienta no disponga de todas las funciones que posee una herramienta comercial similar. A la vista de todo ello, antes de optar por una herramienta de código abierto es importante realizar un análisis detallado de las necesidades reales del grupo de pruebas.

Una ventaja de utilizar herramientas de código abierto es que normalmente sus usuarios pueden modificarlas o ampliarlas. Si la organización de pruebas cuenta con las competencias básicas, la herramienta se podrá modificar para trabajar con otras herramientas o para ajustarse a las necesidades del equipo. Se pueden combinar varias herramientas para resolver problemas que las herramientas comerciales no pueden abordar. Por supuesto, cuantas más herramientas se utilicen y más especificaciones se hagan, mayor será la complejidad y el coste añadido. El Jefe de Pruebas debe asegurarse de que el equipo no empieza a utilizar herramientas de código abierto sin motivo; al igual que con las demás herramientas, el esfuerzo debe estar orientado a obtener un ROI positivo.

El Jefe de Pruebas debe conocer el régimen de licencias de la herramienta seleccionada. Muchas herramientas de código abierto vienen con una variación a la licencia pública general (GPL) por la que se especifica que la distribución del software siempre debe realizarse en los mismos términos que en los que se recibió. Así, en virtud de la licencia de la herramienta, en caso de que el equipo de pruebas modifique la herramienta para dar un mejor soporte a sus pruebas, es posible que estas modificaciones tengan que ponerse a disposición de todos los usuarios externos de la herramienta. El Jefe de Pruebas debería comprobar las consecuencias legales que puede tener para su organización.

Las organizaciones que desarrollan software crítico para la seguridad o de misión crítica, o que estén sujetas a obligaciones normativas, pueden tener problemas para utilizar herramientas de código abierto. Si bien muchas herramientas de código abierto son de muy alta calidad, es probable que la exactitud de cualquier herramienta de código abierto no esté certificada. Las herramientas comerciales a menudo están certificadas en cuanto a su exactitud e idoneidad para una tarea específica (por ejemplo, DO-178B). Aunque las herramientas de código abierto pueden ser igual de buenas, la certificación puede correr a cargo del grupo que la utilice, lo que supone un coste extra.

6.2.2 Herramientas a medida

La organización de pruebas puede concluir que tiene una necesidad específica para la que no existe ninguna herramienta comercial ni de código abierto. Los motivos pueden variar entre disponer de una plataforma propietaria de hardware, hasta tener un entorno personalizado o un proceso que ha sido objeto de una modificación inusual. En estos casos, si el equipo cuenta con las competencias básicas, es posible que el Jefe de Pruebas quiera considerar desarrollar una herramienta a medida.

Las ventajas de desarrollar una herramienta a medida son que la herramienta puede satisfacer las necesidades del equipo de una forma precisa y puede funcionar con eficiencia en el contexto que el equipo necesita. La herramienta puede escribirse para interactuar con otras herramientas actualmente en uso y para generar datos exactamente de la forma que el equipo necesita. Además, la herramienta puede aplicarse en la organización más allá del proyecto inmediato. Sin embargo, antes de planificar la entrega de la herramienta para su uso en otros proyectos, es importante revisar primero el objeto, el objetivo, las ventajas y los posibles inconvenientes de la misma.

El Jefe de Pruebas que esté estudiando la posibilidad de desarrollar una herramienta a medida también tiene que tener en cuenta los posibles aspectos negativos de este acto. A menudo las herramientas a medida dependen de la persona que las crea. Por lo tanto, las herramientas a medida deben estar debidamente documentadas para que otros puedan mantenerlas. De lo contrario, pueden quedarse huérfanas y caer en el olvido una vez que su creador abandone el proyecto. Con el tiempo, se puede ampliar el alcance de las herramientas a medida más allá de su objeto inicial, lo que puede suponer la aparición de problemas de calidad en la herramienta, dando lugar a informes de defectos con resultados de falso positivo o a la generación de datos inexactos. El Jefe de Pruebas debe recordar que una herramienta a medida no es más que otro producto de software, y como tal, está sujeta a los mismos tipos de problemas de desarrollo que cualquier otro producto de software. Las herramientas a medida deberían diseñarse y probarse para garantizar que funcionan según lo esperado.

6.2.3 Retorno sobre la inversión (ROI)

Es responsabilidad del Jefe de Pruebas garantizar que todas las herramientas presentes en la organización de pruebas añaden valor al trabajo del equipo y pueden suponer un ROI positivo para la organización. Para asegurarse de que la herramienta logrará ventajas reales y duraderas, antes de adquirir o construir una herramienta, debería realizarse un análisis de rentabilidad. En este análisis, el ROI debería tener en cuenta tanto los costes recurrentes como los no recurrentes, algunos de los cuales son monetarios y otros son costes de recursos o tiempo, y los riesgos que pueden reducir el valor de la herramienta.

Algunos ejemplos de costes no recurrentes son:

- Definir requisitos de la herramienta para adaptarla a los objetivos y metas
- Evaluar y seleccionar la herramienta correcta y el proveedor adecuado
- Comprar, adaptar o desarrollar la herramienta
- Llevar a cabo una formación inicial sobre la herramienta
- Integrar la herramienta con otras herramientas

- Comprar el hardware/software necesario para soportar la herramienta

Algunos ejemplos de costes recurrentes son:

- Ser propietario de la herramienta
 - Las cuotas de licencia y soporte
 - Los costes de mantenimiento de la propia herramienta
 - El mantenimiento de artefactos creados por la herramienta
 - Los costes de formación continua y orientación
- Trasladar la herramienta a distintos entornos
- Adaptar la herramienta a necesidades futuras
- Mejorar la calidad y los procesos para asegurar el uso óptimo de las herramientas seleccionadas

Asimismo, el Jefe de Pruebas debe tener en cuenta los costes de oportunidad inherentes a cualquier herramienta. El tiempo dedicado a comprar, administrar, formar y utilizar la herramienta podría haberse dedicado a realizar pruebas efectivas; por lo tanto, es posible que se necesiten más recursos de pruebas hasta que la herramienta esté en línea.

El uso de herramientas supone muchos riesgos; no todas las herramientas ofrecen ventajas que compensen los riesgos. Los riesgos de las herramientas se abordaron en el programa de estudio Fundamentos. El Jefe de Pruebas también debería tener en cuenta los siguientes riesgos a la hora de estudiar el ROI:

- Inmadurez de la organización (no está preparada para utilizar la herramienta)
- Los artefactos creados por la herramienta pueden ser difíciles de mantener, debido a las múltiples revisiones requeridas cuando cambia el software objeto de prueba
- La menor intervención de los Analistas de Pruebas en las tareas de pruebas puede reducir el valor de las pruebas (por ejemplo, la efectividad de detección de defectos puede reducirse al ejecutar solo guiones automatizados)

Por último, el Jefe de Pruebas debe estudiar las ventajas que pueden derivarse del uso de la herramienta. Algunos ejemplos de ventajas de la introducción y del uso de herramientas son:

- Reducción de trabajo repetitivo
- Reducción del tiempo de ciclo de pruebas (por ejemplo, empleando pruebas de regresión automatizadas)
- Reducción de los costes de ejecución de las pruebas
- Aumento de ciertos tipos de pruebas (por ejemplo, pruebas de regresión)
- Reducción de errores humanos en distintas fases de las pruebas. Entre otros ejemplos, destacan:
 - Los datos de prueba pueden ser más válidos si se utilizan herramientas de generación de datos
 - Las comparaciones de los resultados de las pruebas pueden ser más exactas si se utilizan herramientas de comparación
 - La entrada de los datos de prueba puede ser más correcta si se introducen con una herramienta de creación de guiones
- Reducción del esfuerzo necesario para acceder a información sobre las pruebas. Entre otros ejemplos, destacan:
 - Informes y métricas generados por las herramientas
 - Reutilización de activos de prueba, como casos de prueba, guiones de prueba y datos de prueba
- Aumento de las pruebas que no sería posible sin el uso de herramientas (por ejemplo, pruebas de rendimiento, pruebas de carga)

- Mejora del estado de los probadores que diseñan la automatización y de la organización de pruebas en general al demostrar el conocimiento y el uso de herramientas sofisticadas

En general, los grupos de pruebas suelen utilizar más de una herramienta; el ROI total que el equipo de pruebas obtendrá normalmente es una función de todas las herramientas utilizadas. Las herramientas tienen que compartir información y trabajar en estrecha colaboración entre sí. Se recomienda establecer una estrategia de herramientas de pruebas exhaustiva y a largo plazo.

6.2.4 Proceso de selección

Las herramientas de pruebas son una inversión a largo plazo que probablemente afecten a muchas iteraciones de un mismo proyecto y/o afecten a muchos proyectos. El Jefe de Pruebas debe estudiar el uso potencial de una herramienta desde varios puntos de vista.

- Por lo que respecta al negocio, es necesario un buen ROI. Para obtener un alto valor a partir de sus inversiones, la organización debería asegurarse de que las herramientas que deben interoperar –lo que puede incluir herramientas de pruebas y herramientas no de pruebas– pueden trabajar juntas. En algunos casos, para lograr esta interoperabilidad es necesario mejorar los procesos y la conectividad para el uso de herramientas, algo que puede llevar cierto tiempo conseguir.
- De cara al proyecto, la herramienta debe ser efectiva (por ejemplo, para evitar errores durante las pruebas manuales, tales como errores de tipografía durante la entrada de datos). La herramienta puede exigir una cantidad considerable de tiempo para empezar a arrojar un ROI positivo. En muchos casos, el ROI puede producirse en la segunda entrega o durante el mantenimiento más que durante el proyecto inicial en el que se implementa la automatización. El Jefe de Pruebas debería tener en cuenta el ciclo de vida total de la aplicación.
- De cara a la persona que utiliza la herramienta, la herramienta debe ayudar a los miembros del proyecto a realizar sus tareas de una forma más eficiente y efectiva. Debe tenerse en cuenta la curva de aprendizaje para garantizar que los usuarios podrán aprender la herramienta rápido y con el mínimo estrés. Cuando se introducen por primera vez, las herramientas de pruebas requieren que los usuarios reciban formación y orientación.

Para garantizar que se tienen en cuenta todos los puntos de vista, es importante crear una hoja de ruta para la introducción de la herramienta de pruebas.

El proceso de selección de una herramienta de pruebas ya se abordó en el programa de estudio Fundamentos como sigue:

- Valorar la madurez de la organización
- Identificar los requisitos de las herramientas
- Evaluar las herramientas
- Evaluar al proveedor o el servicio de soporte (herramientas de código abierto, herramientas a medida)
- Identificar requisitos internos para impartir preparación y formación sobre el uso de las herramientas
- Evaluar las necesidades de formación habida cuenta de las habilidades de automatización de pruebas del equipo de pruebas
- Calcular la rentabilidad (según lo indicado en la Sección 6.2.3 sobre el ROI).

Para cada tipo de herramienta, e independientemente de la fase de prueba en la que se deba utilizar, el Jefe de Pruebas debe tener en cuenta las capacidades que se detallan a continuación:

- Análisis:
 - ¿Podrá esta herramienta “entender” las entradas que se le proporcionan?
 - ¿Es esta herramienta la más indicada para el objetivo?

- Diseño:
 - ¿Ayudará esta herramienta a diseñar productos de soporte de pruebas basándose en la información existente (por ejemplo, herramientas de diseño de pruebas que crean casos de prueba a partir de los requisitos)?
 - ¿Puede el diseño generarse automáticamente?
 - ¿Puede la herramienta generar todos o algunos de los productos de soporte de pruebas efectivos en un formato mantenible y usable?
 - ¿Puede la herramienta generar los datos de prueba necesarios automáticamente (por ejemplo, datos generados a partir del análisis de código)?
- Selección de datos y pruebas:
 - ¿Cómo selecciona la herramienta los datos que necesita (por ejemplo, qué caso de prueba ejecutar con qué conjunto de datos)?
 - ¿Puede la herramienta aceptar criterios de selección que hayan sido introducidos manual o automáticamente?
 - ¿Puede la herramienta establecer cómo “depurar” los datos de producción basándose en las entradas seleccionadas?
 - ¿Puede la herramienta determinar qué pruebas deben realizarse basándose en criterios de cobertura? Por ejemplo, dado un conjunto de requisitos, ¿puede la herramienta recorrer la trazabilidad para establecer qué casos de prueba son necesarios para la ejecución?
- Ejecución:
 - ¿La herramienta funcionará automáticamente o requerirá intervención manual?
 - ¿Cómo se para y se reinicia la herramienta?
 - ¿Debería la herramienta ser capaz de “escuchar” eventos pertinentes? Por ejemplo, ¿debería una herramienta de gestión de pruebas actualizar automáticamente el estado de un caso de prueba si se cierra un defecto reportado asociado a dicho caso de prueba?
- Evaluación:
 - ¿Cómo determinará la herramienta si ha recibido un resultado adecuado? Por ejemplo, ¿la herramienta utilizará un oráculo de pruebas para determinar la idoneidad de una respuesta?
 - ¿Qué tipo de capacidades de recuperación de errores posee la herramienta?
 - ¿Ofrece la herramienta prestaciones adecuadas de registro y elaboración de informes?

6.3 Ciclo de vida de las herramientas

Hay cuatro etapas distintas en el ciclo de vida útil de una herramienta que todo Jefe de Pruebas debe gestionar, son las siguientes:

1. Adquisición. La herramienta debe adquirirse según lo indicado más arriba (remítase a la Sección 6.2 sobre la selección de herramientas). Una vez tomada la decisión de integrar la herramienta, el Jefe de Pruebas debería asignar a alguien (muchas veces un Analista de Pruebas o un Analista Técnico de Pruebas) para que sea el administrador de la herramienta. Esta persona debería decidir cómo y cuándo va a utilizarse la herramienta, dónde van a guardarse los artefactos creados, las convenciones de nombre, etc. El hecho de adoptar estas decisiones de antemano antes de dejar que vayan tomándose ad hoc puede suponer una diferencia importante en el posible ROI de la herramienta. Es probable que los usuarios de la herramienta deban recibir formación.
2. Soporte y mantenimiento. Será necesario implantar procesos constantes de soporte y mantenimiento de la herramienta. La responsabilidad de mantener la herramienta puede competir al administrador de la herramienta, o a un grupo específico de herramientas. Si la herramienta tiene que trabajar con otras herramientas, entonces debería considerarse el intercambio de datos y los procesos de cooperación. También deben estudiarse decisiones sobre copias de seguridad y la restauración de la herramienta y de sus salidas.

3. Evolución. Debe tenerse en cuenta la conversión. A medida que pasa el tiempo, el entorno, las necesidades del negocio o problemas con el proveedor pueden requerir que se realicen cambios importantes en la herramienta o en el modo de utilizarla. Por ejemplo, el proveedor de la herramienta puede exigir una actualización de la herramienta que cree problemas con las herramientas con las que colabora. Un cambio obligatorio en el entorno por motivos de negocio puede provocar problemas con la herramienta. Cuanto más complejo sea el entorno operativo de una herramienta, más cambios de evolución pueden afectar negativamente a su uso. Llegados a este punto, dependiendo del papel que desempeñe la herramienta en las pruebas, el Jefe de Pruebas puede tener que garantizar que la organización dispone de una forma para garantizar la continuidad del servicio.
4. Retirada. Llegará el momento en que la herramienta agote su tiempo de vida útil. En este punto, la herramienta tendrá que retirarse limpiamente. La funcionalidad proporcionada por la herramienta tendrá que ser reemplazada y los datos tendrán que preservarse y archivarlos. Esto puede suceder porque la herramienta esté al final de su ciclo de vida, o simplemente porque haya llegado un punto en el que las ventajas y oportunidades de conversión a una nueva herramienta superan los costes y riesgos.

Durante todo el ciclo de vida de la herramienta, es responsabilidad del Jefe de Pruebas garantizar el correcto funcionamiento y la continuación sin problemas de los servicios que la herramienta ofrece al equipo de pruebas.

6.4 Métricas de las herramientas

Un Jefe de Pruebas puede diseñar y recopilar métricas de objetivos a partir de las herramientas empleadas por los Analistas Técnicos de Pruebas y los Analistas de Pruebas. Distintas herramientas pueden capturar datos valiosos en tiempo real y pueden reducir el esfuerzo de recopilación de datos. El Jefe de Pruebas puede utilizar estas ventajas para gestionar el esfuerzo de prueba general.

Hay varias herramientas que se centran en recopilar distintos tipos de datos. Algunos ejemplos de estas ventajas son:

- Las herramientas de gestión de pruebas pueden proporcionar una serie de métricas distintas. La trazabilidad desde los requisitos hasta los casos de prueba y guiones automatizados permite la obtención de métricas de cobertura. Puede obtenerse una imagen de las pruebas disponibles actualmente, las pruebas previstas y el estado de ejecución actual (pasado, fallido, omitido, bloqueado, en cola) en cualquier momento.
- Las herramientas de gestión de defectos pueden facilitar mucha información sobre defectos, incluyendo el estado actual, la severidad y la prioridad, la distribución en el sistema, etc. Otros datos reveladores como la fase en la que se introdujeron y encontraron los defectos, tasas de omisión, etc. ayudan al Jefe de Pruebas a impulsar la mejora del proceso.
- Las herramientas de análisis estático pueden ayudar a detectar y reportar problemas de mantenibilidad.
- Las herramientas de rendimiento pueden aportar información valiosa sobre la escalabilidad del sistema.
- Las herramientas de cobertura pueden ayudar al Jefe de Pruebas a entender qué parte del sistema se ha sometido a pruebas.

Los requisitos para la elaboración de informes de las herramientas deben definirse durante el proceso de selección de la herramienta. Estos requisitos deben implementarse debidamente durante la configuración de la herramienta para asegurar que la información seguida por las herramientas puede reportarse en formas que sean comprensibles para las partes interesadas.

7. Habilidades de las personas – Composición del equipo – 210 minutos

Palabras clave

Independencia de las pruebas

Objetivos de aprendizaje de Habilidades de las personas - Composición del equipo

7.2 Habilidades individuales

- TM-7.2.1 (K4) Utilizar una hoja de evaluación de habilidades, analizar las fortalezas y debilidades de los miembros del equipo en relación con el uso de sistemas de software, conocimiento del negocio y dominio, áreas de desarrollo de sistemas, pruebas de software y habilidades interpersonales
- TM-7.2.2 (K4) Analizar una evaluación de habilidades dada para un equipo para definir un plan de desarrollo de habilidades y formación

7.3 Dinámica del equipo de prueba

- TM-7.3.1 (K2) Para una situación dada, debatir sobre las habilidades técnicas y sociales necesarias para liderar un equipo de pruebas

7.4 Integrar las pruebas dentro de una organización

- TM-7.4.1 (K2) Explicar las opciones que ofrecen las pruebas independientes

7.5 Motivación

- TM-7.5.1 (K2) Presentar ejemplos de factores motivadores y desmotivadores para los probadores

7.6 Comunicación

- TM-7.6.1 (K2) Explicar los factores que influyen en la efectividad de la comunicación dentro de un equipo de pruebas, y entre un equipo de pruebas y sus partes interesadas

7.1 Introducción

Los Jefes de Prueba de éxito seleccionan, contratan y mantienen equipos con la combinación adecuada de habilidades. Los requisitos de habilidades pueden cambiar con el tiempo, así que además de contratar a las personas adecuadas en primer lugar, también es importante impartir formación y facilitar oportunidades de crecimiento para retener al equipo de pruebas y mantenerlo en un nivel de máximo rendimiento. Además de las habilidades del equipo, el Jefe de Pruebas también debe mantener un conjunto de habilidades que permitan el funcionamiento efectivo en un entorno de alta presión y rápida evolución.

Este capítulo estudia cómo evaluar las habilidades, cómo rellenar lagunas para crear un equipo sinérgico que esté cohesionado internamente y que sea efectivo en una organización, y también cómo motivar a ese equipo y cómo comunicar de una manera efectiva.

7.2 Habilidades individuales

La habilidad de una persona para probar software se puede obtener a través de la experiencia o educación y formación. Cada una de las siguientes puede contribuir a la base de conocimientos del probador:

- Uso de sistemas de software
- Conocimiento del dominio o negocio
- Participación en diferentes fases del proceso de desarrollo del software; incluyendo el análisis, desarrollo y apoyo técnico
- Participación en actividades de pruebas de software

Los usuarios finales de sistemas de software tienen un buen conocimiento sobre cómo funciona el sistema, dónde los fallos tendrían su mayor impacto y cómo debería reaccionar el sistema ante distintas situaciones. Los usuarios con experiencia de dominio saben qué áreas son más importantes para el negocio y cómo afectan estas áreas a la habilidad del negocio para cumplir sus requisitos. Estos conocimientos se pueden emplear para ayudar a priorizar las actividades de prueba, generar información realista sobre pruebas y casos de prueba o verificar y crear casos de uso.

Los conocimientos sobre el proceso de desarrollo del software (análisis de requisitos, diseño y codificación) proporcionan un grado de comprensión importante sobre cómo se pueden introducir errores, dónde se pueden detectar y cómo prevenir su introducción. La experiencia en soporte técnico aporta conocimientos sobre la experiencia del usuario, expectativas y requisitos de usabilidad. La experiencia del desarrollo de software es importante para el uso de herramientas de pruebas que requieren habilidades de programación y diseño, así como para participar en análisis de código estático, revisiones de código, pruebas unitarias y pruebas de integración técnicamente orientadas.

Los programas de estudio Fundamentos, Analistas de Pruebas de Nivel Avanzado y Analistas Técnicos de Pruebas de Nivel Avanzados, abordan algunas habilidades específicas de las pruebas de software, tales como la capacidad para analizar una especificación, participar en análisis de riesgos, diseñar casos de prueba y la diligencia para ejecutar pruebas y registrar los resultados.

En especial para los jefes de pruebas con conocimientos sobre el tema, son importantes las habilidades y experiencia en gestión de proyectos, ya que la gestión de pruebas incluye muchas actividades de gestión de proyectos, como proponer un plan, hacer un seguimiento del progreso, y comunicarse con las partes interesadas. A falta de un jefe de proyecto, el Jefe de Pruebas puede asumir el papel de Jefe de Pruebas y jefe del proyecto, sobre todo durante las últimas fases de un proyecto. Estas habilidades se suman a las capacidades abordadas en este y en el programa de estudios de Nivel Fundamentos.

Además de aptitudes técnicas, las habilidades interpersonales, como la de hacer y recibir críticas constructivas, tener influencia, y negociar también son importantes a la hora de llevar a cabo procesos de pruebas. Los probadores técnicamente competentes tienen más probabilidad de fallar si no disponen y emplean las habilidades sociales necesarias. Además de trabajar en equipo de forma efectiva, el buen profesional de las pruebas debe ser también una persona organizada, perfeccionista y poseer habilidades importantes de comunicación oral y escrita.

El equipo de pruebas ideal cuenta con una combinación de habilidades y niveles de experiencia, y los miembros del equipo deben tener voluntad y capacidad para enseñar y aprender de sus homólogos. En algunos entornos, algunas habilidades serán más importantes o más respetadas que otras. Por ejemplo, en un entorno de pruebas técnicas donde se requieren pruebas API y habilidades de programación, es posible que se valoren más las habilidades técnicas que el conocimiento del dominio. En un entorno de pruebas de caja negra, la experiencia en el dominio puede ser lo que más se valore. Es importante recordar que los entornos y los proyectos cambian.

A la hora de crear una hoja de evaluación de habilidades, el Jefe de Pruebas debería enumerar todas las habilidades que son importantes para el trabajo y que son adecuadas para el puesto. Una vez enumeradas esas habilidades, se podrá evaluar a cada persona integrante del equipo empleando un sistema de clasificación (por ejemplo, una clasificación del 1 al 5, siendo 5 el nivel de habilidad más alto esperado en esa área). Las personas pueden evaluarse para establecer cuáles son sus fortalezas y sus debilidades y, tomando como base esta información, crear planes de formación individuales o colectivos. El Jefe de Pruebas puede fijar un objetivo de rendimiento para que las personas mejoren sus habilidades en áreas específicas, para lo que debería definir los criterios específicos utilizados para evaluar las habilidades de la persona.

Las personas deberían contratarse a largo plazo, y no para participar en un solo proyecto. Cuando el Jefe de Pruebas invierte en los probadores y crea un ambiente de aprendizaje continuo, los miembros del equipo estarán motivados para aumentar sus habilidades y conocimiento para estar preparados para la próxima oportunidad.

Un Jefe de Pruebas casi nunca podrá contratar al equipo perfecto. Y, aún cuando los miembros del equipo sean perfectos para el proyecto en curso, puede que no sean la combinación perfecta para el siguiente proyecto. El Jefe de Pruebas debe contratar a personas inteligentes, curiosas, adaptables, dispuestas a trabajar, capaces de trabajar con efectividad como parte de un equipo y con ganas y capacidad para aprender. Si bien probablemente no se dispone de ese conjunto de personas perfectas, se puede crear un equipo competente buscando el equilibrio entre los puntos fuertes y débiles de las personas que lo forman.

Mediante la hoja de evaluación de habilidades, el Jefe de Pruebas puede identificar las fortalezas y debilidades del grupo. Esta información conformará las bases de un plan de habilidades y desarrollo. Empezando por las debilidades que tienen un mayor impacto en la efectividad y eficiencia del equipo, el Jefe de Pruebas debe decidir cómo abordar esas áreas. Un posible enfoque es la formación, por ejemplo, mandando a personas a un curso de formación, impartir sesiones de formación en la empresa, desarrollar formación a medida, utilizar cursos de formación a distancia. Otro enfoque es el autoestudio, por ejemplo, libros, seminarios web, recursos de Internet. Y aún otro enfoque es la formación transversal, por ejemplo, poner a alguien interesado en aprender una habilidad a trabajar con alguien que ya posee esa habilidad, hacer que los expertos locales den charlas sobre su área de conocimiento, etc. (la orientación⁴ es un enfoque parecido, donde una persona que acaba de llegar a un puesto trabaja en colaboración con una persona con más experiencia que ya ha ocupado ese puesto, y la persona senior actúa como un recurso constante para prestarle ayuda y asesoramiento). Además de abordar las debilidades, el Jefe de Pruebas no debe olvidar fomentar las fortalezas identificadas en la evaluación de habilidades como parte del plan de formación y desarrollo de

⁴ "orientación" es la traducción de "mentoring"

habilidades. Para más información sobre los planes de desarrollo de los equipos de pruebas, remítase a [McKay07].

7.3 Dinámica del equipo de pruebas

Para crear el mejor equipo posible, la selección de personal es una de las funciones más importantes de los puestos de gestión de una organización. Hay muchos factores a tener en cuenta además de las habilidades personales específicas que requiere el puesto de trabajo. Para seleccionar a una nueva persona para que se sume al equipo, se debe tener en cuenta la dinámica de ese equipo. ¿Será esa persona capaz de complementar las habilidades y tipos de personalidades ya existentes en el equipo de pruebas? Es importante tener en cuenta las ventajas de tener distintos tipos de personalidades en un equipo de pruebas, así como una mezcla de habilidades técnicas. Un equipo de prueba competente será capaz de gestionar múltiples proyectos de diferente complejidad y de mantener satisfactoriamente las relaciones interpersonales entre los miembros del equipo de pruebas.

En muchas ocasiones realizar pruebas conlleva mucha presión. Los plazos de desarrollo de software suelen ser apretados, a veces son incluso irrealistas. Las partes interesadas esperan mucho del equipo de pruebas, a veces de manera poco razonable. El Jefe de Pruebas debe contratar a personas que sepan trabajar en situaciones de presión, que sepan asumir la frustración y que puedan concentrarse en el trabajo, aunque los plazos parezcan imposibles. El Jefe de Pruebas será el que aborde los problemas de plazos y expectativas, pero el Jefe de Pruebas también tendrá que comprender que los miembros del equipo también notan esta presión. Cuando el Jefe de Pruebas contrata a personas para el equipo, es importante que tengan en cuenta el entorno de trabajo y que ajuste los tipos de personalidad a ese entorno. En un entorno donde hay más trabajo que tiempo, el Jefe de Pruebas debería buscar a personas que acaben sus tareas y pregunten qué hacen a continuación.

Las personas trabajarán más duro y tendrán más cuidado en lo que hacen si se sienten valoradas y necesarias. Las personas deben entender que todas ellas son un miembro importante del equipo y que su aportación es vital para el éxito de todo el equipo. Cuando se crea esta dinámica, se producirá una formación transversal informal, los propios miembros del equipo serán los que se encarguen de distribuir la carga de trabajo y el Jefe de Pruebas tendrá más tiempo para abordar problemas externos.

Los nuevos miembros del equipo deberán integrarse rápidamente en el equipo y estar debidamente supervisados y asistidos. Cada persona debe tener un papel definido dentro del equipo. Esto puede desprenderse de un proceso de evaluación individual. El objetivo es hacer que cada persona sea competente de forma individual y que también contribuya al éxito del equipo. Esto se consigue en gran medida combinando los tipos de personalidad con las diferentes funciones dentro del equipo y construyendo a partir de las habilidades individuales de cada persona, mejorando así sus respectivas "carteras" de habilidades.

A la hora de decidir a quién contratar o añadir al equipo de pruebas, una evaluación objetiva puede ser de utilidad. Esta evaluación puede hacerse mediante entrevistas, poniendo a prueba al candidato, revisando muestras de su trabajo y comprobando sus referencias. Las habilidades a evaluar incluyen:

Las habilidades técnicas (habilidades duras⁵) se demuestran:

- Estableciendo casos de prueba a partir de un documento de requisitos
- Revisando la documentación técnica (requisitos, código, etc.)
- Redactando comentarios de revisiones de una forma clara, comprensible y objetiva

⁵ "habilidades duras" es la traducción de "hard skills"

- Aplicando varias técnicas correctamente en escenarios dados (por ejemplo, empleando tablas de decisión para probar una serie de reglas de negocio)
- Evaluando un fallo y documentándolo correctamente
- Demostrando que conoce la información de clasificación de defectos
- Demostrando que conoce las causas raíz de los defectos
- Utilizando una herramienta para probar una API, incluyendo pruebas positivas y negativas
- Utilizando SQL para encontrar y alterar información de bases de datos para probar un escenario dado
- Designar un arnés de pruebas de automatización que ejecute varias pruebas y recopile los resultados de las pruebas
- Ejecutando pruebas automatizadas y diagnosticando fallos
- Redactando planes/especificaciones de pruebas
- Redactando un informe resumen de pruebas que incluya una evaluación de los resultados de las pruebas

Las habilidades interpersonales (habilidades blandas⁶) se demuestran:

- Presentando información sobre un proyecto de pruebas que está fuera de plazo
- Explicando un informe de defectos a un desarrollador que considera que no hay ningún defecto
- Formando a un compañero
- Planteando a la dirección un problema con un proceso que no resulta efectivo
- Revisando un caso de prueba creado por un compañero y presentando los comentarios a esa persona
- Entrevistando a un compañero
- Felicitando a un desarrollador

Si bien no se trata de una lista exhaustiva y las habilidades deseadas específicas pueden variar según el entorno y la organización, en ella se recogen las habilidades que normalmente se necesitan. Al crear preguntas de entrevista efectivas y permitir demostraciones de habilidades, el Jefe de Pruebas puede evaluar las habilidades del candidato y determinar sus fortalezas y debilidades. Una vez creada una buena serie de evaluaciones, esta debería aplicarse a todo el personal existente para establecer sus áreas de crecimiento y formación.

Además de las habilidades requeridas por los participantes individuales, el Jefe de Pruebas también debe poseer excelentes habilidades de comunicación y dotes diplomáticas. El Jefe de Pruebas debe ser capaz de resolver situaciones controvertidas, debe conocer los medios correctos a utilizar para la comunicación en cada caso y debe ser capaz de concentrarse en crear y mantener relaciones dentro de la organización.

7.4 Integrar las pruebas dentro de una organización

Las organizaciones tienen muchas formas de integrar las pruebas en su estructura organizativa. Aunque la calidad sea responsabilidad de todos a lo largo del ciclo de vida de desarrollo del software, un equipo de pruebas independiente puede contribuir enormemente a conseguir la calidad del producto. La independencia de las funciones de prueba varía en gran medida en la práctica, como se muestra en la siguiente lista, ordenada de menor a mayor independencia:

- Probadores no independientes
 - En este caso no hay independencia, y el código lo prueba el desarrollador que lo ha escrito

⁶ "habilidades blandas" es la traducción de "soft skills"

- El desarrollador, si cuenta con el tiempo necesario para realizar las pruebas, determinará si el código funciona como esperaba, lo cual puede o no cumplir los requisitos reales
 - El desarrollador puede arreglar rápidamente cualquier defecto que encuentre
- Las pruebas las lleva a cabo un desarrollador diferente al autor del código
 - Hay poca independencia entre el desarrollador y el probador
 - Un desarrollador que pruebe el código de otro desarrollador puede ser reticente a comunicar los defectos
 - La atención de un desarrollador con respecto a los procesos de prueba se centra normalmente en los casos de prueba positivos
- Las pruebas las realiza un probador (o equipo de pruebas) que forma parte del equipo de desarrollo
 - El probador (o el equipo de pruebas) puede reportar a la dirección del proyecto o puede reportar al jefe de desarrollo
 - El propósito del probador se centra más en verificar el cumplimiento de los requisitos
 - Dado que el probador es un miembro del equipo de desarrollo, el probador puede ser responsable de cuestiones de desarrollo además de serlo de los procesos de prueba
 - El jefe del probador puede estar más concentrado en cumplir los plazos que en lograr los objetivos de calidad
 - Las pruebas pueden tener un estatus más bajo que el desarrollo dentro del equipo
 - El probador puede no tener autoridad para influenciar los objetivos de calidad o el cumplimiento de dichos objetivos
- Las pruebas las realizan especialistas de pruebas procedentes de la organización empresarial, comunidad de usuarios u otra organización técnica no encargada del desarrollo
 - La información sobre los resultados de las pruebas se reporta objetivamente a las partes interesadas
 - La calidad es el principal foco de atención de este equipo
 - El desarrollo de habilidades y la formación están enfocados a los procesos de prueba
 - Las pruebas se consideran una vía profesional
 - Hay un equipo de dirección específico dedicado al grupo de pruebas que está concentrado en la calidad
- Los especialistas de pruebas externos llevan procesos de pruebas sobre tipos de prueba específicos
 - La experiencia se aplica a áreas específicas donde es probable que las pruebas generalizadas no sean suficiente
 - Los tipos de pruebas podrían ser usabilidad, seguridad, rendimiento u otras áreas que requieren especialización
 - La calidad debería ser el foco de atención de estas personas, pero su visión se limita a sus áreas de especialidad. Un producto que rinde bien puede no cumplir sus requisitos funcionales, algo que puede pasar inadvertido para el especialista de rendimiento
- Las pruebas las lleva a cabo una organización ajena a la empresa.
 - Este modelo logra la máxima independencia entre el probador y el desarrollador
 - La transferencia de conocimiento puede no ser suficiente para que el probador realice las pruebas de una manera competente
 - Se precisa que los requisitos estén claros y la estructura de comunicación bien definida
 - La calidad de la organización externa se debe auditar periódicamente

La lista se basa en el foco de atención típico de las personas, pero puede no ser cierta en una organización específica. El puesto y el cargo no siempre determinan el foco de atención de la persona. Los jefes de desarrollo pueden estar centrados en la calidad, por lo que pueden ser buenos Jefes de Pruebas. Los Jefes de Pruebas Independientes pueden reportar a una cadena de mando centrada en los plazos, por lo que pueden estar más centrados en los plazos que en la calidad. Para

establecer la mejor ubicación de un equipo de pruebas dentro de una organización, el Jefe de Pruebas debe conocer los objetivos de la organización.

Existen diferentes grados de independencia entre las organizaciones de desarrollo y de pruebas. Es importante tener en cuenta que puede ocurrir que cuando mayor sea la independencia, mayor puede ser aislamiento y menor la transferencia de información. Un grado de independencia menor puede aumentar los conocimientos, pero también puede dar lugar a conflictos en cuanto a los objetivos. El nivel de independencia también lo determinará el software, en función del modelo de desarrollo de software que se esté empleando; por ejemplo, en un desarrollo Ágil, los probadores son, casi siempre, parte del equipo de desarrollo.

Cualquiera de las opciones anteriores puede estar presente en el seno de una organización. Puede que las organizaciones de desarrollo se encarguen de los procesos de prueba, o que sean organizaciones independientes, y puede también haber un proceso de certificación final por parte de una organización externa. Es importante comprender las responsabilidades y expectativas de cada fase del proceso de pruebas y establecer requisitos que maximicen la calidad del producto final, además de respetar las exigencias con respecto a los plazos y presupuesto.

7.5 Motivación

Existen muchas formas de motivar a una persona implicada en un proceso de pruebas. Entre ellas, destacan:

- Reconocimiento del trabajo realizado
- Aprobación por parte de la dirección
- Respeto al equipo del proyecto y entre los compañeros
- Mayor responsabilidad y autonomía
- Recompensas adecuadas por el trabajo realizado (en concepto de aumento de salario, mayor responsabilidad y reconocimiento)

No obstante, existen factores que influyen sobre el proyecto y que hacen que estos mecanismos de motivación sean difíciles de aplicar. Por ejemplo, un probador que trabaja eficientemente en un proyecto con un plazo imposible de cumplir. El probador puede hacer todo lo posible por centrar la atención del equipo en la calidad, hacer horas y esfuerzo extra, y puede que el producto salga antes de lo que debería, debido a factores externos. Como consecuencia, el resultado será un producto con una calidad deficiente a pesar de los grandes esfuerzos del probador. Esto puede resultar muy desmotivador si no se reconoce y analiza la contribución del probador, independientemente de si el producto final es satisfactorio o no.

El equipo de pruebas debe asegurarse de que está aplicando las métricas adecuadas para demostrar que se ha llevado a cabo un buen trabajo, y poder así realizar las pruebas, mitigar los riesgos y registrar de forma correcta los resultados. A no ser que se considere y haga pública esa información, es fácil que el equipo caiga en la desmotivación, al no recibir ningún tipo de reconocimiento que considera merecido por haber hecho un buen trabajo.

El reconocimiento no sólo se demuestra por medio de cuestiones intangibles como el respeto y la aprobación, sino que también debe estar presente en forma de ascensos, mayor responsabilidad y mayor mérito. Si falta el respeto hacia el grupo de prueba, es posible que no se planteen estas oportunidades.

El reconocimiento y el respeto se adquieren cuando queda patente que el probador contribuye al incremento del valor del proyecto. En el marco de un proyecto individual se pueden alcanzar mejor, mediante la participación del probador desde el momento en el que se concibe el proyecto y el mantenimiento de tal participación a lo largo de su ciclo de vida. Con el tiempo, los probadores se ganarán el respeto de las demás partes interesadas del proyecto mediante su aportación positiva al

proyecto. Esta aportación también debería cuantificarse en términos de reducciones del coste de la calidad y mitigación de riesgos.

El Jefe de Pruebas desempeña una función importante en la motivación de las personas que conforman un equipo de pruebas y actúa como el gran defensor del equipo de pruebas de cara al resto de la organización. El Jefe de Pruebas debería reconocer los logros de los probadores individuales, y también debe evaluar de manera justa y honesta los errores cometidos. Un Jefe de Pruebas justo y coherente motivará a los miembros del equipo dando ejemplo.

7.6 Comunicación

La comunicación del equipo de pruebas se lleva a cabo principalmente a través de los siguientes medios:

- Documentación de productos de pruebas: estrategia de pruebas, plan de pruebas, casos de prueba, informes resúmenes de pruebas, informes de defectos, etc.
- Realimentación de los documentos revisados: requisitos, especificaciones funcionales, casos de uso, documentación de pruebas de componentes, etc.
- Recopilación y divulgación de información: interacción con desarrolladores, otros miembros del equipo de prueba, miembros de la dirección, etc.

La comunicación entre los probadores y las demás partes interesadas debe ser profesional, objetiva y efectiva, con el fin de construir y mantener el respeto hacia el equipo de prueba. Se necesita diplomacia y objetividad a la hora de dar respuestas o comentarios, en especial cuando se trata de retroactividad constructiva sobre los productos de trabajo de otros. Además, la comunicación debe estar encaminada a alcanzar objetivos de prueba y a mejorar la calidad tanto en los productos como en los procesos empleados para producir los sistemas de software.

Los Jefes de Pruebas se comunican con diferentes interlocutores, que pueden ser desde los usuarios, los miembros del equipo de pruebas o de la dirección, hasta grupos de prueba externos o clientes. La comunicación debe ser efectiva teniendo en cuenta los interlocutores a quien va orientada. Así por ejemplo, un informe elaborado por el equipo de desarrollo que muestre tendencias en el número y la severidad de los defectos encontrados puede resultar demasiado detallado para su uso en una reunión informativa con un director ejecutivo. En cualquier comunicación, pero sobre todo durante las presentaciones, es importante conocer el mensaje que se quiere trasladar, las formas en las que el mensaje puede percibirse y la explicación que debe darse para crear el entorno adecuado para que el mensaje se acepte. Dado que a menudo el Jefe de Pruebas tiene que presentar información sobre el estado del proyecto, es importante que esta información se encuentre en un nivel de detalle adecuado (por ejemplo, los jefes normalmente quieren ver tendencias de defectos más que los defectos individuales), y que se presente de una forma clara y comprensible (por ejemplo, diagramas sencillos y gráficos de colores). La comunicación eficiente ayuda a mantener la atención del público mientras se traslada el mensaje correcto. El Jefe de Pruebas debería enfrentarse a cada presentación como una oportunidad para promover la calidad y los procesos de calidad.

El Jefe de Pruebas no solo se comunica con personas ajenas al departamento (comunicación externa). Una parte importante del trabajo del Jefe de Pruebas consiste en establecer una comunicación efectiva dentro del grupo de pruebas (comunicación interna) para transmitir noticias, instrucciones, cambios en prioridades y demás información estándar que se imparte dentro del proceso normal de pruebas. Un Jefe de Pruebas también puede comunicarse con personas específicas en sentido ascendente (comunicación ascendente) y descendente (comunicación descendente) dentro de la cadena de dirección de la organización. Independientemente de la dirección de la comunicación, deben aplicarse las mismas reglas; la comunicación debería ser adecuada para el público, el mensaje debería enviarse de manera efectiva y debería confirmarse que se ha comprendido.

Los Jefes de Pruebas deben dominar los distintos medios de comunicación. Mucha información se comunica mediante correo electrónico, interacciones verbales, reuniones formales o informales, informes formales o informales e incluso a través del uso de herramientas de gestión de pruebas como las herramientas de gestión de defectos. Toda la comunicación debe ser profesional y objetiva. La revisión, tanto a efectos de calidad como de contenido, es un paso necesario hasta para la comunicación más urgente. La comunicación por escrito a menudo perdura más que el proyecto en curso. Es importante que el Jefe de Proyecto presente documentación profesional y de calidad que sea representativa de una organización que promueve la calidad.



8. Referencias

8.1 Estándares

- [BS7925-2] BS 7925-2 Estándar de Pruebas de Componentes Software
Capítulo 2
- [FDA21] Título 21 CFR Parte 820 de la Agencia de Alimentos y Medicamentos de los Estados Unidos sobre sistemas médicos
Capítulo 2
- [IEEE829] Norma IEEE para la documentación de pruebas de software
Capítulos 2 y 4
- [IEEE1028] Norma IEEE para revisiones y auditorías de software
Capítulo 2
- [IEEE1044] Norma IEEE para la clasificación de anomalías de software
Capítulo 4
- [ISO9126] ISO/IEC 9126-1:2001, Ingeniería del software - Calidad del producto software
Capítulos 2 y 4
- [ISO12207] ISO 12207, Sistemas e ingeniería de software - Procesos de ciclo de vida del software
Capítulo 2
- [ISO15504] ISO/IEC 15504, Tecnología de la Información - Evaluación de Procesos
Capítulo 2
- [ISO25000] ISO/IEC 25000:2005, Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE)
Capítulos 2 y 4
- [RTCA DO-178B/ED-12B]: Consideraciones para la certificación de software en sistemas y equipos de aviónica, RTCA/EUROCAE ED12B.1992.
Capítulo 2

8.2 Documentos ISTQB

- [ISTQB_AL_OVIEW] Resumen del Nivel Avanzado ISTQB, Versión 1.0
- [ISTQB_ALTM_SYL] Programa de estudio Jefe de Pruebas de Nivel Avanzado ISTQB, Versión 1.0
- [ISTQB_ALTM_SYL] Programa de estudio de Analista Técnico de Pruebas de Nivel Avanzado ISTQB, Versión 1.0
- [ISTQB_ETM_SYL] Programa de Estudio de Gestión de Pruebas de Nivel Experto, Versión 1.0
- [ISTQB_AL_OVIEW] Programa de estudio del Nivel Fundamentos ISTQB, Versión 2011
- [ISTQB_GLOSSARY] Glosario estándar de términos utilizados en pruebas de software, Versión 2.2, 2012
- [ISTQB_ITP_SYL] Programa de estudio de Nivel Experto del ISTQB "Mejora del proceso de pruebas", Versión 1.0

8.3 Marcas comerciales

En este documento se utilizan las siguientes marcas registradas y marcas de servicio:

- CMMI®, IDEALSM, ISTQB®, ITIL®, PRINCE2®, TMMi®, TPI Next®
- CMMI está registrada en la Oficina de Patentes y Marcas de Estados Unidos por la Carnegie Mellon University.
- IDEAL es una marca de servicio del Software Engineering Institute (SEI), Carnegie Mellon University
- ISTQB es una marca registrada del Comité Internacional de Cualificación de Pruebas de Software (International Software Testing Qualifications Board)
- ITL es una marca registrada y una marca comunitaria registrada de la Office of Government Commerce, y está registrada en la Oficina de Patentes y Marcas de Estados Unidos.
- PRINCE2 es una marca registrada de la Cabinet Office.
- TMMi es una marca registrada de la TMMi Foundation.
- TPI Next es una marca registrada de Sogeti Nederland B.V.

8.4 Referencias bibliográficas

- [Black03]: Rex Black, "Critical Testing Processes", Addison-Wesley, 2003, ISBN 0-201-74868-1
- [Black09]: Rex Black, "Managing the Testing Process, third edition," John Wiley & Sons, 2009, ISBN 0-471-22398-0
- [Black11]: Rex Black, Erik van Veenendaal, Dorothy Graham, "Foundations of Software Testing," Thomson Learning, 2011, ISBN 978-1-84480-355-2
- [Craig02]: Rick Craig, Stefan Jaskiel, "Systematic Software Testing," Artech House, 2002, ISBN 1-580-53508-9
- [Crispin09]: Lisa Crispin, Janet Gregory, "Agile Testing: A Practical Guide for Testers and Agile Teams", Addison-Wesley, 2009, ISBN 0-321-53446-8
- [de Vries09]: Gerrit de Vries, et al., "TPI Next", UTN Publishers, 2009, ISBN 90-72194-97-7
- [Goucher09]: Adam Goucher, Tim Riley (editors), "Beautiful Testing," O'Reilly, 2009, ISBN 978-0596159818
- [IDEAL96]: Bob McFeeley, "IDEAL: A User's Guide for Software Process Improvement," Software Engineering Institute (SEI), 1996, CMU/SEI-96-HB-001
- [Jones07]: Capers Jones, "Estimating Software Costs, second edition," McGraw-Hill, 2007, ISBN 978-0071483001
- [Jones11]: Capers Jones and Olivier Bonsignour, "Economics of Software Quality," Pearson, 2011, ISBN 978-0132582209
- [McKay07]: Judy McKay, "Managing the Test People," Rocky Nook, 2007, ISBN 978-1933952123
- [Musa04]: John Musa, "Software Reliability Engineering, second edition," Author House, 2004, ISBN 978-1418493882
- [Stamatis03]: D.H. Stamatis, "Failure Mode and Effect Analysis," ASQC Quality Press, 2003, ISBN 0-873-89300
- [vanVeenendaal11] Erik van Veenendaal, "The Little TMMi," UTN Publishers, 2011, ISBN 9-490-986038
- [vanVeenendaal12] Erik van Veenendaal, "Practical Risk-based Testing," UTN Publishers, 2012, ISBN 978-9490986070
- [Whittaker09]: James Whittaker, "Exploratory Testing," Addison-Wesley, 2009, ISBN 978-0321636416
- [Wieggers03]: Karl Wieggers, "Software Requirements 2," Microsoft Press, 2003, ISBN 978-0735618794

8.5 Otras referencias

Las siguientes referencias apuntan a información disponible en Internet. Estas referencias fueron consultadas en el momento de la publicación de este programa de estudio de Nivel Avanzado.

<http://www.istqb.org>
<http://www.sei.cmu.edu/cmmi/>
<http://www.tmmi.org/>
<http://www.tpinext.com/>



9. Índice terminológico

- actividades de cierre de pruebas, 16
- acuerdos de nivel de servicios (ANS), 10
- Ágil, 10, 14, 21, 23, 32, 35, 38, 40, 43, 47, 50, 85
- amplitud primero, 30
- Análisis de Árbol de Faltas (AAF), 32
- análisis de la densidad de los defectos, 63
- análisis de las condiciones de prueba, 34
- análisis de los riesgos de la calidad, 26
- Análisis de Modos de Fallo y Efectos en el Software (AMFE), 32
- análisis de peligros, 32
- análisis de pruebas, 11
- análisis de riesgos, 18, 29
- anomalía, 60, 61
- auditoría, 53, 55
- base de prueba, 11
- bitácora de la prueba, 8
- caso de prueba, 8, 13
- causa raíz, 60, 65
- ciclo de mejora Deming, 67
- ciclo de vida
 - Ágil, 23
 - casca, 22
 - iterativo, 22
 - modelo-V, 22
- ciclo de vida de las herramientas, 77
- ciclo de vida de los defectos, 62
- cierre de pruebas, 8
- CMMI, 51, 68
- código abierto, 73
- comité de gestión de defectos, 63
- comité de triaje de defectos, 60, 63
- composición del equipo, 79
- comunicación, 86
- comunicación ascendente, 86
- comunicación descendente, 86
- comunicación externa, 86
- comunicación interna, 86
- condición de prueba, 8, 11, 18
- contención en fase, 60, 61
- contrato de pruebas, 25
- control de pruebas, 8, 10, 11, 18, 48
- coste de exposición, 32
- coste de la calidad, 49
- costes de detección, 49
- costes de fallos externos, 49
- costes de fallos internos, 49
- costes de prevención, 49
- critérios de entrada, 24
- critérios de salida, 8, 24
- Critical Testing Processes (CTP), 66
- CTP, 67, 68, 70
- defecto, 60, 61
 - campos, 65
- Delphi de banda ancha, 18, 43
- desmotivador, 85
- Despliegue de la Función de Calidad (DFC), 32
- dinámica del equipo de pruebas, 82
- director de pruebas, 18, 20
- diseño de pruebas, 8, 13
- ejecución de pruebas, 8, 15
- elaboración de informes, 24
- enfoque de prueba, 18, 39
- enfoque reactivo, 35
- enfoques metódicos, 35
- entregables, 24
- error, 61
- especialistas de pruebas externos, 84
- estimación de pruebas, 18, 42
- estrategia de pruebas, 9, 18, 37, 38
- estrategias
 - analítica, 37
 - basadas en modelos, 37
 - consultivas, 38
 - de cumplimiento de procesos o normas, 38
 - metódicas, 37
 - pruebas de regresión, 38
 - reactivas, 38
- Estructuras de Desglose del Trabajo (EDT), 43
- evaluación de habilidades, 81
- evaluación de los criterios de salida y elaboración de informes, 15
- evaluación de riesgos, 18, 28
- fallo, 60, 61
- gestión de pruebas, 18, 20
- gestión de revisiones formales, 59
- gestión de riesgos, 18, 27, 30
- gestión de riesgos del proyecto, 40
- gestión Lean, 47
- gráfico causa-efecto, 34
- guion de prueba, 8
- habilidades de las personas, 79
- habilidades individuales, 80
- habilidades interpersonales, 81, 83
- habilidades técnicas, 82
- herramientas de prueba, 24
- IDEAL, 68

identificación de riesgos, 27
IEEE, 50
IEEE 1028, 51
IEEE 829, 51
impacto, 28
implementación de pruebas, 8, 14
informe resumen de pruebas, 8
inspección, 53
integrar las pruebas dentro de una organización, 83
ISO, 50
ISO 25000, 28
ISO 9126, 28
ITIL, 51
jefe de pruebas, 20
licencia pública general (GPL), 73
líder de pruebas, 18, 20
marco de monitorización, 10
mejora del proceso de pruebas, 67, 68
mejorar el proceso de pruebas con CTP, 70
mejorar el proceso de pruebas con STEP, 71
mejorar el proceso de pruebas con TMMi, 69
mejorar el proceso de pruebas con TPI Next, 70
métricas, 9, 24, 45, 46, 47, 85
métricas de las herramientas, 78
métricas de las personas, 44
métricas de pruebas, 43
métricas del proceso, 43
métricas del producto, 43
métricas del proyecto, 43
métricas para las revisiones, 58
misión, 9
mitigación de riesgos, 18, 29
modelo secuencial, 22
moderador, 53
monitorización de pruebas, 18
motivación, 85
nivel de prueba, 18, 40
nivel de riesgo, 18, 28, 29
normas
 BS-7925-2, 51
 CMMI, 66
 DO-178B, 29, 51
 ED-12B, 29, 51
 IEC 61508, 29
 US FDA Título 21 CFR Parte 820, 51
número de prioridad del riesgo, 28
partes interesadas, 20
partes interesadas de negocio, 33
partes interesadas técnicas, 33
perfil operativo, 37
perfiles de uso, 35
plan de nivel de prueba, 40
plan de pruebas, 18, 22, 29
plan de pruebas de nivel, 18
plan de revisión, 53
plan maestro de pruebas, 18, 39
planificación de pruebas, 8, 9
planificación, monitorización y control de pruebas, 9
PMI, 51
política de pruebas, 18, 36
PRINCE2, 51
prioridad, 60
probabilidad, 28
procedimiento de pruebas, 8
proceso de pruebas básico, 9
profundidad primero, 30
pruebas basadas en la experiencia, 25
pruebas basadas en requisitos, 34
pruebas basadas en riesgos, 9, 10, 18, 26, 27, 30, 31, 32, 33, 34, 35, 36, 37, 65
pruebas de confirmación, 29
pruebas de integración de hardware-software, 24
pruebas de integración de producto del cliente, 24
pruebas de integración de sistema, 24
pruebas de interacción de prestación, 24
pruebas de regresión, 29
pruebas distribuidas, 49
pruebas exploratorias, 25
pruebas externalizadas, 49
pruebas internalizadas, 49
pruebas no funcionales, 24
puntuación global del riesgo, 29
resultado de falso negativo, 60, 61
resultado de falso positivo, 60, 62
retorno sobre la inversión (ROI), 74
revisión, 53, 54
revisión de gestión, 53, 55, 56
revisión de la ambigüedad, 34
revisión guiada, 53
revisión informal, 53
revisión técnica, 53
revisor, 53
riesgo, 18, 26
riesgo de la calidad, 18, 26
riesgo de la calidad del producto, 26
riesgo de planificación, 26
riesgo de producto, 9, 18, 26
riesgo del proyecto, 18, 26, 40
riesgo técnico, 28
sesiones de pruebas, 25
severidad, 60, 64

STEP, 67, 68, 71
Systematic Test and Evaluation Process
(STEP), 66
tablero Kanban, 47
técnicas, 24
tipos de mejora de procesos, 68

TMMi, 66, 67, 68, 69
TPI Next, 66, 67, 68, 70
trazabilidad, 11
valores cualitativos, 49
valores cuantitativos, 49

