



Curso ISTQB® Foundation LEVEL AGILE TESTER



Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil





2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

Pruebas gestión de la configuración

¡Recuerdo de memoria!

¿ Qué es la gestión de la configuración?

- La gestión de la configuración para ISTQB es una disciplina que se centra en establecer procesos y prácticas para identificar, controlar y rastrear los elementos de la configuración de software durante el desarrollo y las pruebas, con el fin de garantizar la integridad y la calidad del producto.
- El objetivo principal es garantizar la integridad y la trazabilidad de los elementos de la configuración, controlar los cambios en ellos y facilitar la reproducción de entornos de prueba.



2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

Pruebas gestión de la configuración

Pruebas gestión de la configuración

- Los proyectos ágiles a menudo implican un gran uso de herramientas automatizadas para desarrollar, probar y gestionar el desarrollo
- Desarrolladores utilizan herramientas para:
 - Análisis estático, pruebas unitaria y cobertura de código.
 - Comprueban continuamente el código y las pruebas unitarias en un sistema de gestión de la configuración a través a marcos de trabajo automatizados de construcción y de pruebas.
 - Estos marcos de trabajo permiten la integración continua



2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

Pruebas gestión de la configuración

Pruebas automatizadas en la Gestión de la configuración

- Se pueden ejecutar pruebas automatizadas incluyendo pruebas funcionales a nivel de integración y de sistema.
- Cómo las pruebas automatizadas funcionales son de mayor duración, se ejecutan con menos frecuencia que las pruebas unitarias.
 - Ejemplo: Las pruebas unitarias se ejecutan cada vez que se hace una subida de código, y las funcionales más largas sólo cada pocos días.
- El objetivo de las pruebas automatizadas es confirmar que el software construido funciona y es instalable.
 - Si alguna prueba automatizada falla, el equipo arregla el defecto a tiempo para la siguiente subida de código
- Las pruebas automatizadas y las herramientas de construcción permiten gestionar el riesgo de regresión asociado a los cambios frecuentes.
 - Las pruebas unitarias tienen una eficacia limitada para detectar defectos, por lo que son necesarias las pruebas automatizadas a nivel de integración y sistema

Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil





2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

Opciones organizativas para pruebas independientes

¡Recuerdo de memoria!

- ¿A qué se refiere con probadores independientes?
 - Son profesionales de pruebas de software que realizan pruebas de manera imparcial y objetiva, sin estar involucrados en el desarrollo de software.
 - El objetivo es identificar problemas y defectos en el software de una perspectiva externa y proporcionar una evaluación crítica de la calidad del producto.



2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

Opciones organizativas para pruebas independientes

Existen tres opciones para implementar los equipos de tester en los proyectos ágiles:

- Probadores independientes suelen ser más eficaces en la búsqueda de defectos
 - En metodologías ágiles, muchos desarrolladores realizan el papel de los probadores y generan pruebas automatizadas
 - Uno o más probadores pueden ser integrarse dentro del equipo, para realizar muchas de las tareas de prueba . Sin embargo, dado la integración fuerte de estos probadores puede romper la independencia y no ser objetivos
- Equipos de prueba separados independientes, la asignación de los testers bajo demanda
 - Se integran al equipo los probadores durante los últimos días del sprint
 - esto puede preservar la independencia
 - bajo este enfoque, la presión del tiempo y la falta de comprensión de las nuevas características a menudo conducen a problemas



2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

Opciones organizativas para pruebas independientes

- Equipo de pruebas independiente separado
 - al comienzo del proyecto, los probadores están asignados al equipo ágil sobre una base a largo plazo
 - manteniendo su independencia
 - ganando una buena comprensión del producto
 - la construcción de una fuerte relación con otros miembros del equipo (estructura de matriz) y las partes interesadas
- Probadores especializados fuera de los equipos ágiles :
 - trabajar en largo plazo y/o actividades iteración-independientes
 - el desarrollo de herramientas de pruebas automatizadas
 - la realización de las pruebas no funcionales
 - crear y apoyar entornos de prueba y datos
 - la realización de los niveles de prueba que podrían no encajar bien dentro de un sprint(por ejemplo , pruebas de integración de sistemas)

¿Cuál es mejor?

Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

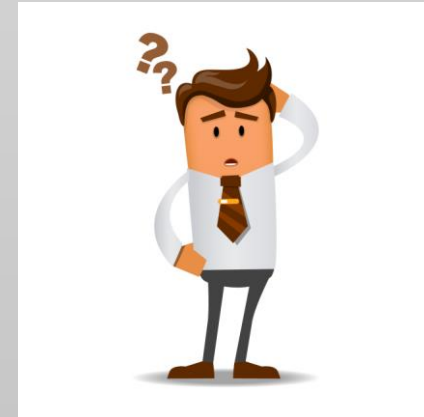
- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil



2.2 Estado de las pruebas en los proyectos ágiles

Generalidades

- En los proyectos ágiles los cambios se producen rápido.
- Estos cambios pueden ser:
 - Evoluciones constantes en el estado de pruebas
 - Cambios en los progresos de las pruebas
 - Cambios en la calidad del productos



¿Cómo probadores que debemos hacer?



2.2 Estado de las pruebas en los proyectos ágiles

Generalidades

¿Cómo probadores que debemos hacer?

- Nuestra responsabilidad es encontrar la forma de hacer llegar la información de estos cambios al equipo, para tomar mejores decisiones.
- Debemos hacer notar que estos cambios pueden afectar a entregas y/o iteraciones anteriores.
- Debemos actualizar las pruebas manuales y automatizadas para enfrentar los riesgos en la regresión

Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil





2.2 Estado de las pruebas en los proyectos ágiles

Comunicación del estado de la prueba, progreso y calidad del producto

- El entregable en los equipos ágiles es: **UN SOFTWARE QUE FUNCIONA AL FINAL DE CADA ITERACION.**
- Para saber cuándo se tendrá este software, hay que hacer seguimiento del progreso de todos los elementos de trabajo de la iteración y de la entrega.
- Algunos métodos de seguimiento al trabajo:
 - Resultados de pruebas automatizadas
 - El progreso de las tareas de prueba y las historias en el tablero de tareas ágil
 - Gráficos de burndown que muestran el progreso del equipo



2.2 Estado de las pruebas en los proyectos ágiles

Comunicación del estado de la prueba, progreso y calidad del producto

¿Cómo comunicaciones los resultados del seguimiento?

Dashboard, wiki, emails estilo dashboard o verbalmente durante las reuniones o con herramientas automáticas.

La comunicación debe incluir: Métricas del proceso de prueba

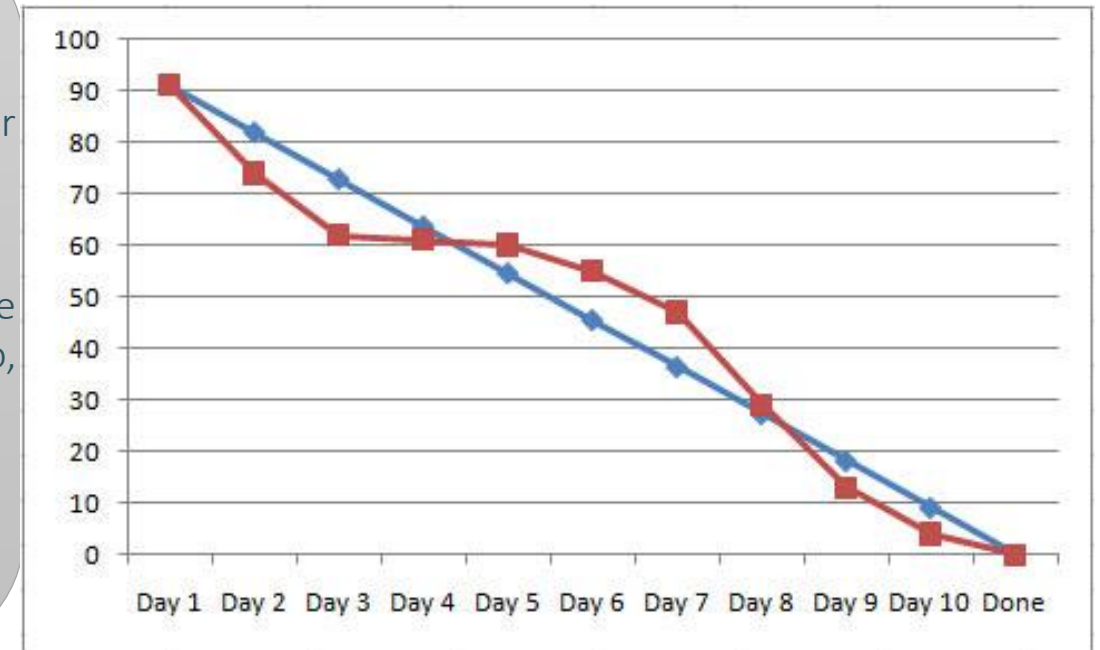
- Hacer la comunicación de forma automatizada, deja más tiempo y libera a los probadores para concentrarse en diseñar y ejecutar más casos de prueba

2.2 Estado de las pruebas en los proyectos ágiles

Algunas herramientas de seguimiento:

Seguimiento con gráfico Burndown

- El gráfico burndown muestra las tareas y esfuerzo que aún queda por realizar durante la iteración o sprint.
- Se utiliza la medida de puntos de historia o de días/horas.
- Utilizando las estimaciones anteriores, se dibuja una línea de tendencia, lo que indica la fecha prevista de finalización del proyecto, las desviaciones forman el plan de tiempo del Sprint .

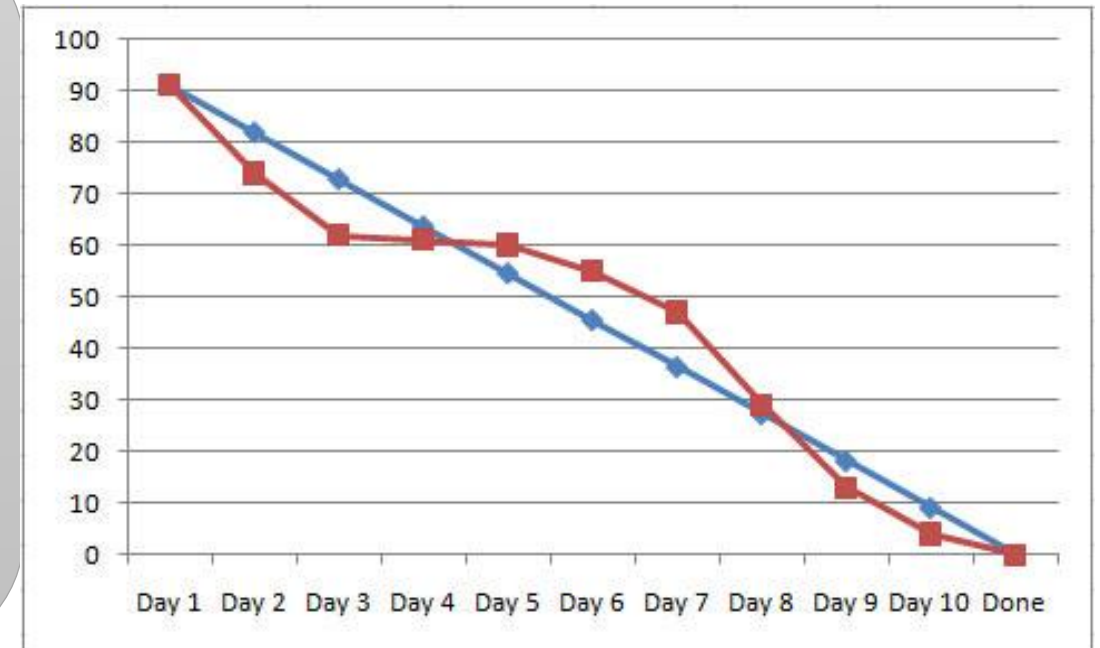


2.2 Estado de las pruebas en los proyectos ágiles

Algunas herramientas de seguimiento:

Seguimiento con gráfico Burndown

- Se puede ocupar tanto para la iteración como para la entrega
- ¿Qué obtienes del gráfico burndown?
 - trabajo todavía no terminado
 - tiempo todavía disponible



2.2 Estado de las pruebas en los proyectos ágiles

Algunas herramientas de seguimiento:

Seguimiento con Tablero de tareas ágiles

- Permite la visualización detallada e instantánea del estado actual del equipo incluyendo el estado de la prueba
- En el tablero se muestran:
 - Tarjetas de las historias
 - Tareas del desarrollo
 - Tareas de pruebas
 - Otras Tareas definidas para la iteración

Se recomienda ocupar tarjetas de colores por tipo de tarea



2.2 Estado de las pruebas en los proyectos ágiles

Algunas herramientas de seguimiento:

Seguimiento con Tablero de tareas ágiles

- Durante la iteración:
 - Se mueven las tareas en las distintas columnas definidas por el equipo , por ejemplo: "para hacer", "En curso", "Verificar", "Hecho"
 - Las tareas de prueba normalmente se reflejan en los criterios de aceptación de las historias de usuarios. Cuando las pruebas automatizadas, manuales y exploratorias pasan a estado aprobado, la tarea se pasa a la columna " Hecho"

Estos tableros se pueden automatizar o ocupar herramientas online como trello

Scrum Board			
Stories	To Do	In Progress	Done
Story 1			
Story 2	 	 	 
Story 3	 		

2.2 Estado de las pruebas en los proyectos ágiles

Algunas herramientas de seguimiento:

Seguimiento con Reuniones diarias

- En esta reunión todos comunican:
 - ¿Que ha hecho desde la ultima reunión?
 - ¿Que tiene previsto hacer antes de la próxima reunión?
 - ¿Algún impedimento?
- En esta reunión se comunica cualquier problema que pueda bloquear el progreso de las pruebas, de manera que todo el equipo este informado y apoye en la solución



2.2 Estado de las pruebas en los proyectos ágiles

Comunicación del estado de la prueba, progreso y calidad del producto

Otras formas de comunicación para mejorar la calidad del producto

- Encuestas de satisfacción para recibir feedback sobre si el producto cumple con las expectativas del cliente
- Pueden utilizar otros indicadores similares a los capturados en las metodologías tradicionales de desarrollo
 - tasa de error
 - tasa de descubrimiento de defectos
 - resultados de las pruebas de confirmación y de regresión
 - densidad de defectos
 - defectos encontrados y reparados, código modificado para mejorar la calidad del producto
 - la cobertura de los requisitos, la cobertura de riesgo, la cobertura de código



Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil





2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

- En los proyectos ágiles, el producto crece a medida que van completándose las iteraciones. Lo que significa que el alcance de las pruebas crecen.
- Además de probar los cambios de código realizados en la iteración actual, deben comprobar que no se ha introducido ningún cambio en las iteraciones anteriores.
 - El riesgo de introducir regresión es alto en metodologías ágiles debido a la alta rotación de código (líneas de código añadidas, modificadas o eliminadas)
- También pueden haber cambios en iteraciones anteriores por necesidades del negocio
- Dar respuesta al cambio es un principio clave en las metodologías ágiles.

2.2 Estado de las pruebas en los proyectos ágiles

Cómo gestionar el cambio para no aumentar la deuda técnica

- Es fundamental que los equipos inviertan tiempo temprano en la automatización de pruebas en todos los niveles de prueba lo antes posible.
- Todos los activos de prueba se deben mantener actualizados en cada iteración
 - Pruebas automatizadas
 - Datos de pruebas
 - Otros artefactos
- Se recomienda que todos los activos se mantengan en una herramienta de gestión de la configuración, permitiendo:
 - Control de versiones
 - Garantizar acceso por todos los del equipo
 - Facilitar la realización de los cambios necesarios a causa de cambios de la funcionalidad sin perder información histórica



2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

- Debido a que es difícil repetir todas las pruebas en la regresión, los probadores deben **asignar tiempo en cada iteración para revisar los casos de pruebas manuales y automatizadas procedentes de iteraciones anteriores y actuales** para :
 - seleccionar los casos de prueba que pueden ser candidatos para el conjunto de pruebas de regresión
 - Retirar los casos de prueba que ya no son relevantes (por ejemplo, debido a los cambios de función)
- Mientras revisan los casos, hay que tener en cuenta que casos de prueba pueden ser automatizados.
- Es importante automatizar el máximo de número de pruebas de iteraciones anteriores y actuales, para reducir el esfuerzo y riesgo de las regresiones.
- Los probadores deben tener la capacidad de
 - identificar rápidamente y actualizar los casos de prueba de iteraciones/entregas anteriores afectadas por los cambios realizados en la iteración actual
- Durante la planificación de la entrega, el equipo debe definir como se diseñarán, se escribirán y almacenarán los casos de prueba.

2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

Beneficios de Automatización de pruebas en todos los niveles

- La automatización de pruebas en todos los niveles, permite a los equipos ágiles proporcionar una rápida retroalimentación sobre la calidad del producto
- Las pruebas automatizadas bien escritas proporcionan un documento vivo de la funcionalidad del sistema [Crispin08]
- Se recomienda tener las pruebas automatizadas y los resultados en el sistema de gestión de la configuración.

1. Pruebas Unitarias

2. Pruebas de
aceptación
Automatizadas

3. Pruebas de
Verificación de Versión

4. Pruebas de
regresión

2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

Beneficios de Automatización de pruebas en todos los niveles

Pruebas unitarias automatizadas

- Se ejecutan antes de que el código fuente se registre en el sistema de gestión de la configuración para garantizar que los cambios del código no rompen la versión del software.
- El objetivo es garantizar que los cambios del código no rompen la versión del software y así relentizar el progreso de todo el equipo.
- Los resultados de las pruebas unitarias proporcionan información inmediata sobre código y calidad de construcción
- La calidad del producto no es el foco de pruebas unitarias

1. Pruebas Unitarias

2. Pruebas de
aceptación
Automatizadas

3. Pruebas de
Verificación de Versión

4. Pruebas de
regresión

2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

Beneficios de Automatización de pruebas en todos los niveles

Pruebas de aceptación automatizada

- Se ejecutan continuamente, como parte de la integración continua de todo el sistema de construcción
- están dirigidas contra una versión completa del sistema (integrada), pero no se ejecutan en cada subida de código.
- tardan más tiempo en ejecutarse que las pruebas unitarias automatizadas, por lo que no se ejecutan en cada subida de código ya que puede relentizar el proceso.
- Los resultados de estas pruebas, proporcionan feedback sobre la calidad del producto en relación con la regresión desde la última versión, pero no establecen el estado de la calidad general del producto.

1. Pruebas Unitarias

2. Pruebas de
aceptación
Automatizadas

3. Pruebas de
Verificación de Versión

4. Pruebas de
regresión

2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

Beneficios de Automatización de pruebas en todos los niveles

Pruebas verificación de versión

- Pruebas automatizadas que pueden ejecutarse de manera continua contra el sistema, inmediatamente después de desplegar una nueva versión en el entorno de prueba.
- Son un subconjunto de pruebas automatizadas que cubren los puntos críticos de la funcionalidad y de la integración del sistema.
- Dan feedback instantáneo sobre el software después del despliegue, así los equipos no pierden tiempo probando una versión inestable.

1. Pruebas Unitarias

2. Pruebas de
aceptación
Automatizadas

3. Pruebas de
Verificación de Versión

4. Pruebas de
regresión

2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

Pruebas de regresión automatizadas se ejecutan como parte de la construcción principal diaria

- Se deben ejecutar en un entorno de integración continua
- Corren cada vez que una nueva construcción se despliega en el entorno de prueba
- cuando una prueba de regresión automatizada falla
 - el equipo deja de otras actividades e investiga las razones del fracaso
 - la prueba puede haber fallado debido a cambios funcionales, por lo tanto, necesitan actualizar la prueba y la historia de usuario.
 - como alternativa, otra prueba debe ser escrita y se elimina la prueba fallida
 - si la prueba ha fallado debido a un defecto, la mejor práctica es solucionar el defecto antes de avanzar con nuevas características.

1. Pruebas Unitarias

2. Pruebas de
aceptación
Automatizadas

3. Pruebas de
Verificación de Versión

4. Pruebas de
regresión



2.2 Estado de las pruebas en los proyectos ágiles

Riesgos en las pruebas de regresión

Otras automatizaciones

- Generación de datos de prueba
- Carga de datos de pruebas en los sistema
- Despliegue de versiones construidas en entornos de pruebas
- Restauración de un entorno de pruebas a una versión anterior
- Comparación de los datos de salida

La automatización de estas tareas reduce los costes y permite al equipo dedicar tiempo a desarrollar y probar nuevas prestaciones.

Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil



2.3 Función y habilidades de un probador en un equipo ágil

Función y habilidades de un probador en un equipo ágil

- En un equipo ágil, los probadores deben trabajar en estrecha colaboración con los demás miembros del equipo y con las partes interesadas del negocio.
- Esto significa que hay que tener habilidades distintas a las de un probador tradicional



Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil



2.3 Función y habilidades de un probador en un equipo ágil

Habilidades del tester ágil

- habilidades del tester de CTFL [ISTQB_FL_SYL]
- además, para el proceso de trabajo ágil e iterativo , alta competencia en:
 - automatización de pruebas
 - desarrollo basado en pruebas (TDD)
 - basado en pruebas de aceptación del desarrollo (ATDD)
 - Pruebas de caja blanca y caja negra
 - pruebas basadas en la experiencia





2.3 Función y habilidades de un probador en un equipo ágil

Habilidades del tester ágil

- Las habilidades blandas que debe tener cada uno para interactuar correctamente deben ser:
 - Ser positivos y estar orientados a soluciones con los miembros del equipo y otras partes interesadas
 - Demostrar un pensamiento crítico, orientado hacia la calidad y escéptico sobre el producto
 - Adquirir activamente información de las partes interesadas (en lugar de confiar íntegramente en las especificaciones escritas)
 - Evaluar y reportar con exactitud los resultados de las pruebas, el progreso de las pruebas y la calidad del producto.
 - Trabajar junto con los representantes de negocio y las partes interesadas de forma eficaz para definir historias de usuario testeables y especialmente criterios de aceptación,
 - Colaborar con el equipo, trabajar por pares con programadores y otros miembros del equipo
 - Responder rápidamente al cambio, cambiando, añadiendo o mejorando los casos de prueba
 - Planificar y organizar su propio trabajo

Capítulo 2:

Principios, prácticas y procesos clave de las pruebas ágiles

2.1 Diferencias entre pruebas según un enfoque tradicional o un enfoque ágil

- Actividades de pruebas y desarrollo
- Productos de trabajo del proyecto
- Niveles de prueba
- Pruebas y gestión de la configuración
- Opciones organizativas para las pruebas independientes

2.2 Estado de las pruebas en los proyectos ágiles

- Comunicación del estado de la prueba, progreso y calidad del producto
- Gestión de riesgos de regresión con casos de prueba manuales y automatizados

2.3 Función y habilidades de un probador en un equipo ágil

- Habilidades de los probadores ágiles
- Función de un probador en un equipo ágil



2.3 Función y habilidades de un probador en un equipo ágil

Rol de un tester en un equipo ágil

- El rol de un tester en el equipo ágiles incluyen actividades que generan y proporcionan feedback sobre:
 - Estado de las pruebas
 - Progreso de las pruebas
 - Calidad del producto
 - Calidad del proceso





2.3 Función y habilidades de un probador en un equipo ágil

Rol de un tester en un equipo ágil

- Adicional a las actividades vistas en los capítulos anteriores, el tester debe
 - Conocer, implementar y actualizar la estrategia de pruebas
 - Medir y reportar la cobertura de pruebas en todas las dimensiones de cobertura aplicables
 - Asegurar el uso adecuado de las herramientas de pruebas
 - Configurar, usar y gestionar entornos de pruebas y datos de pruebas
 - Reportar defectos y trabajar con el equipo para solucionarlos
 - Orientar a otros miembros del equipo en cuanto a aspectos relevantes de las pruebas.
 - Asegurar que se planifiquen las tareas de pruebas adecuadas durante la planificación de la entrega y de la iteración
 - Colaborar activamente con los desarrolladores y partes interesadas del negocio para aclarar los requisitos, especialmente en términos de facilidad de prueba, consistencia e integridad
 - Participar proactivamente en las reuniones retrospectivas de equipo, sugiriendo e implementando mejoras

* En un equipo ágil, cada miembro del equipo es responsable de la calidad del producto y desempeña una función en la ejecución de tareas relacionadas



2.3 Función y habilidades de un probador en un equipo ágil

Riesgos en organizaciones ágiles

- Existen algunos riesgos que las organizaciones ágiles pueden enfrentarse:
 - Los probadores trabajan tan cerca de los desarrolladores que pierden la mentalidad apropiada de probador
 - Los probadores se vuelven tolerantes o silencian prácticas ineficaces, ineficientes o de baja calidad dentro del equipo
 - Los probadores no pueden llevar el ritmo de los cambios entrantes en iteraciones limitadas por tiempo

¿Qué podríamos hacer para mitigar estos riesgos?



Curso ISTQB® Foundation LEVEL AGILE TESTER

