



Curso ISTQB® Foundation LEVEL AGILE TESTER



Capítulo 1:

Desarrollo ágil de software

1.1 Fundamentos del desarrollo ágil de software

- Desarrollo ágil de software y el Manifiesto Ágil
- Enfoque de equipo completo
- Feedback temprano y frecuente

1.2 Aspectos de enfoques ágiles

- Enfoques de desarrollo ágil de software
- Creación de historias de usuario colaborativas
- Retrospectivas
- Integración continua
- Planificación de la entrega y planificación de la iteración



Capítulo 1:

Desarrollo ágil de software

1.1 Fundamentos del desarrollo ágil de software

- Desarrollo ágil de software y el Manifiesto Ágil
- Enfoque de equipo completo
- Feedback temprano y frecuente

1.2 Aspectos de enfoques ágiles

- Enfoques de desarrollo ágil de software
- Creación de historias de usuario colaborativas
- Retrospectivas
- Integración continua
- Planificación de la entrega y planificación de la iteración



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

HISTORIAS DE USUARIO

¿Qué son?

- Es la descripción de los requisitos desde la perspectiva de los desarrolladores, probadores y representantes del negocio.
- Es manera simple de describir una tarea de forma concisa que aporta valor al usuario o al negocio.
- Las historias de usuario pueden ser creadas durante las conversaciones con los usuarios interesados (stakeholders) sobre nuevas funcionalidades o mejoras del proyecto.
- Las historias de usuario son una invitación a la conversación.
- Las historias de usuario deben incluir características funcionales como no funcionales

HU01 - Imagen del Sistema

Como: Usuario del sistema

Quiero: Visualizar que el sistema se identifique con la imagen de la Universidad.

Para: Lograr pertenencia hacia la Universidad

Condiciones:

- Los colores amarillo y verde deben predominar. ([Ver mapa de colores](#))
- En la ventana de ingreso debe estar el lema de la Universidad.
- El encabezado debe ser similar al de la página web. ([Ver imagen de portal](#))
- El login del usuario debe ser visible en todo momento.



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

HISTORIAS DE USUARIO

¿Por qué las usamos?

Utilizamos las historias de usuario porque siguen los principios básicos de los requisitos ágiles:

- Potencian la participación del equipo en la toma de decisiones.
- Se crean y evolucionan a medida que el proyecto avanza.
- Son peticiones concretas y pequeñas.
- Contiene la información imprescindible. Menos es más.
- Apoyan la cooperación, colaboración y conversación entre los miembros del equipo, lo que es fundamental.

Los equipos ágiles varían en términos de como documentan las historias de usuarios. Independiente del enfoque para documentar las historias de usuario, lo importante es que sean concisas, suficientes y necesarias.

1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

La creación de la historia de usuario consta de 3 partes

- **Tarjeta** Una descripción escrita en lenguaje de negocio que sirve como identificación y recordatorio del requerimiento y ayuda para la planificación mediante la priorización. (Quién, qué y para qué)
- **Conversación** El dialogo que ocurre entre los miembros del equipo y el PO para aclarar los detalles y dudas sobre esa HU. Es la parte más importante de la historia.
- **Confirmación** Que pruebas se llevaran a cabo para poder decir que la HU se ha completado con éxito. Puede añadirse en la conversación entre el team y el PO



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

Una buena historia de usuario también sigue el modelo INVEST:

- Independiente
- Negociable
- Estimable
- Pequeña
- Testeable

Y Adicionalmente debe ser siempre Valiosa



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

- Independiente:

una historia debe ser independiente de otras, para facilitar la planificación, priorizar y estimación.

- Negociable

La "tarjeta" de la historia es tan sólo una descripción corta que no incluye detalles. Los detalles se añaden mediante la conversación.



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

- Estimable:

El equipo necesita poder estimar una historia de usuario. Historias demasiado grandes o inconcretas, no se pueden estimar.

- Pequeña

una buena historia debe ser pequeña en esfuerzo u debería ser realizable en menos de una semana.



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

- **Testeable**

una historia necesita poder probarse y saber que la HU se ha completado con éxito.

- **Valiosa**

cada historia tiene que tener valor para el cliente (para el usuario o para el comprador).



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

El modelo Invest también nos habla de otros atributos importantes:

- **Lo completo:** ¿Fue toda la información disponible incluida?, pueden todas las características ser trazadas a documentos y discusiones anteriores?
- **Consistencia:** Formato, el uso del lenguaje y la presentación de características debe ser consistente a través de todos los documentos
- **Sin ambigüedad:** No hay declaraciones incluidas que son ambiguas como "amigable " o "rápida"



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

- **Realizable:** ¿Los requisitos son significativos y pueden ser implementados ?
- **Trazabilidad:** De donde viene el requerimiento?, porque esta en uso y que estrategia de negocios soporta?
- **Específico:** ¿Es toda la información incorporada en el orden en que se va a entregar ?
- **Medible:** ¿Es posible medir si los requisitos estan completamente implementados y listos para su uso?
- **Aceptable:** ¿Es correcto el uso de cálculos , el lenguaje, la lógica y las fórmulas?
- **Alcanzable:** ¿Puede la historia de usuario ser implementada en esta iteración?



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

¿Cómo debe ser una historia de usuario?

- El **título** debe ser un breve texto para poder referenciar esta HU.
- La **descripción** es dónde describimos la HU, de forma tal que se debe focalizar en el objetivo, en el por qué y no en el cómo.

Usamos la formula "**Como... Quiero... Para...** "

Clarifica que se quiere conseguir, a quien aportara valor y porqué una HU es útil. Este formato deja la posibilidad abierta de cómo debe implementarse una HU, puedes aportar ideas de otras formas de obtener el objetivo de la HU.

Título de la Historia

Como <role, persona>

Persona o rol de usuario que tiene la necesidad

Quiero <objetivo, comportamiento>

Lo que se quiere obtener: una funcionalidad, característica, etc.

Para <motivo, razón, valor>

Motivo por el que se necesita, valor que se obtiene como resultado, etc.

1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

- **Criterios de aceptación**

Una breve descripción de "hecho", que pruebas se llevarán a cabo para poder decir que la HU se ha completado con éxito.

Si es necesario, se adjuntará mensaje de "error" y de mensaje de "éxito"

Ejemplo: "La búsqueda me ha de devolver una línea por cada oferta encontrada o el mensaje de error 'No se encuentran ofertas con el TEXTO BUSCADO'"

Ejemplo: "Si los resultados son más de 20, poder paginar entre ellos"

Historia Negociable 1

Como: Bloguero

Quiero: hacer una entrada al blog

Para: posicionarme como experto en un tema específico

Criterios de Aceptación:

- Debo ser capaz de publicar contenido multimedia (imágenes y video)
- El texto de la entrada debe ser enriquecido (que permita enlaces Web, formato, etc.)
- La entrada se debe poder compartir vía redes sociales
- La entrada se debe poder imprimir
- La entrada se debe poder enviar vía correo electrónico

Prototipo. Como buena práctica, recomendamos el uso de prototipados siempre que podamos como explicación e ilustración de la HU

El equipo ágil considera una tarea acabada cuando se cumplen una serie de criterios de aceptación.

1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

Desde la perspectiva del tester

- Detecta detalles no identificados
- Identifica los requisitos no funcionales que faltan y los completa
- Puede hacer preguntas abiertas a los representantes de las empresas
- Propone formas para probar las historias de usuario
- Confirma criterios de aceptación
- Lluvia de ideas y mapas mentales son técnicas para crear historias de usuario



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario

Ejemplo malas historias de usuario

Este software se va a crear en C ++

Esta no es una historia de usuario , ya que, desde el punto de vista del usuario , el lenguaje de programación no es un tema relevante

El programa se conecta con la base de datos de la agrupación de conexiones

Esta no es una historia de usuario , ya que, desde el punto de vista del usuario , la solución técnica no es un tema relevante





1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Historia de usuario- Ejercicio

Revisar las siguientes historias de usuario, completarlas en caso que falte información y escribir al menos 3 criterios de aceptación.

1. Como Representante de proveedor, quiero ofertar una cotización para un proceso que esté abierto por licitación.
2. Como Ejecutivo de cuenta, quiero poder abrir una nueva cuenta de cheques a un cliente, seleccionando el código de cliente y producto bancario.
3. Como Representante de proveedor, quiero poder consultar los procesos de licitación que están en curso.
4. Como Vendedor, quiero registrar los productos y cantidades que me solicita un cliente para crear un pedido de venta.

Capítulo 1:

Desarrollo ágil de software

1.1 Fundamentos del desarrollo ágil de software

- Desarrollo ágil de software y el Manifiesto Ágil
- Enfoque de equipo completo
- Feedback temprano y frecuente

1.2 Aspectos de enfoques ágiles

- Enfoques de desarrollo ágil de software
- Creación de historias de usuario colaborativas
- Retrospectivas
- Integración continua
- Planificación de la entrega y planificación de la iteración



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Retrospectiva

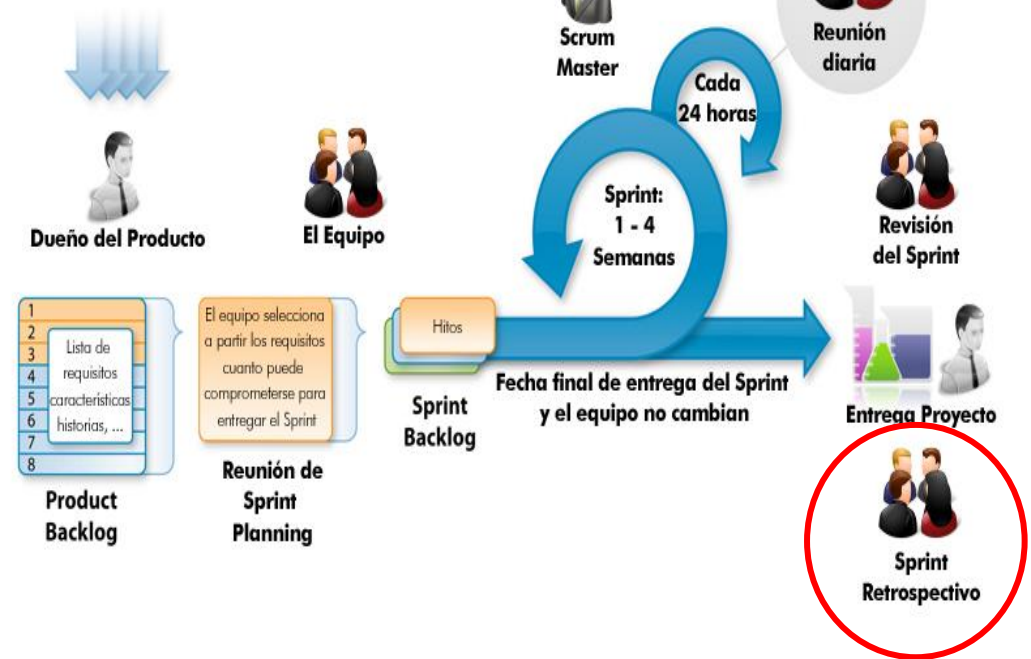
¿Qué es una retrospectiva?

- Es una reunión de equipo al final de cada iteración

En esta reunión se revisa:

- Lo que fue un éxito y lo que se puede mejorar
- Como incorporar mejoras en futuras iteraciones
- Cómo conservar los hábitos y prácticas exitosas

Aportaciones de los ejecutivos,
equipo, clientes, usuarios
y partes interesadas



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Retrospectiva

¿Qué es una retrospectiva?

Las retrospectivas cubren temas como:

- El proceso
- Los participantes
- Organizaciones y relaciones
- Herramientas

El objetivo es la mejora continua y realizarlas es crítico para auto-organización y la mejora continua del desarrollo y las pruebas.



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Retrospectiva

Los resultados de una retrospectiva son:

La mejora de las decisiones de prueba con respecto a:

- Efectividad de pruebas
- Productividad de pruebas
- Calidad de casos de prueba
- Satisfacción del equipo

También se analizan los resultados sobre:

- Facilidad de probar la aplicación
- Sobre las historias de usuarios
- Las funcionalidades o las interfaces



Importante: Debe limitarse a unas pocas mejoras en cada iteración

1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Retrospectiva

- Los tiempos, cuando se realiza y como se organiza depende del método ágil que se realiza, por ejemplo XP es cada 3 meses, Scrum al final de cada sprint.

¿Quienes participan?

- Representantes del negocio y equipo, incluidos los probadores.
- El facilitador (scrum master) organiza y dirige la reunión
- Pueden invitar a otros participantes que sean importantes para el equipo



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Retrospectiva

¿Cuál es el rol de tester?

- Los testers deben jugar un papel importante en la retrospectiva
- Los testers son
 - Parte del equipo
 - Traen su perspectiva única
 - Realizan una contribución sustancial al éxito
- Las pruebas son realizadas en todas las iteraciones.
- Las pruebas también se benefician de la contribución de los otros miembros del equipo



Capítulo 1:

Desarrollo ágil de software

1.1 Fundamentos del desarrollo ágil de software

- Desarrollo ágil de software y el Manifiesto Ágil
- Enfoque de equipo completo
- Feedback temprano y frecuente

1.2 Aspectos de enfoques ágiles

- Enfoques de desarrollo ágil de software
- Creación de historias de usuario colaborativas
- Retrospectivas
- Integración continua
- Planificación de la entrega y planificación de la iteración





1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Integración Continua

- La integración continua busca que la entrega de un incremento sea un software confiable, que funciona e integrado al final de cada sprint.
- La integración continua aborda fusionando todos los cambios realizados al software e integrando todos los componentes desarrollados de forma periódica, **al menos una vez al día**.
- Se unen en un solo proceso, repetible y automatizado:
 - ✓ Administración de configuración
 - ✓ Compilación y construcción de SW
 - ✓ Despliegue
 - ✓ Ejecución de pruebas

Cómo los desarrolladores integran su trabajo, construyen y prueban de forma constante, se detectan más rápido los defectos de código



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Integración Continua

Cuando los desarrolladores ya han desarrollado, depurado y subido el código a un repositorio compartido, el proceso de integración continua realiza las siguientes actividades automatizadas:

- ✓ **Análisis de código estático**
Estas revisiones de código se ejecutan y los resultados son informados
- ✓ **Compilación**
compilar y enlazar el código, generando archivos ejecutables
- ✓ **Pruebas unitarias**
Ejecución de pruebas unitarias de código , comprueba la cobertura de código de la prueba y reportar los resultados de las pruebas
- ✓ **Despliegue**
Instala en un entorno de pruebas el software que se ha construido
- ✓ **Prueba de integración**
Ejecutando integración de pruebas y reportando resultados
- ✓ **Reporte- resultados (dashboard)**
Los resultados de las actividades anteriormente se publican en un lugar bien visible o se envían como correo electrónico como un informe de estado



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Integración Continua

- Diariamente se realiza un proceso automatizado de compilación (build) y pruebas para detectar errores de integración de forma rápida y temprana.
- La integración continua permite a los probadores:
 - Ejecutar pruebas automatizadas, proporcionando feedback rápido al equipo sobre la calidad del código.
 - Realizar pruebas automatizadas de regresión, las cuales se pueden realizar durante toda la iteración.
- Los resultados de las pruebas automatizadas deben estar visibles para todos los miembros de equipo.
- Las pruebas automatizadas de regresión:
 - Deben cubrir lo máximo posible de la funcionalidad, incluyendo HU integradas en iteraciones anteriores.
 - Unas buenas pruebas de regresión automatizadas son un gran apoyo a la construcción de sistemas integrados de gran tamaño.
 - Cuando se realizan pruebas automatizadas de regresión, los probadores ágiles quedan liberados para crear y ejecutar pruebas manuales de nuevas funcionalidades, de cambios implementados y pruebas de confirmación de defectos.



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Integración Continua

Beneficios

- Permite una detección temprana y un análisis más sencillo de la causa raíz de problemas de integración
- Se entrega retroalimentación frecuente al equipo de desarrollo si el código se está trabajando
- Los testers no están a más de un día detrás de los desarrolladores
- Reduce los riesgos de regresión a causa de la rápida refactorización del código
- Las pruebas de re-test y regresión se realizan después de un pequeño conjunto de cambios
- Proporciona confianza en una sólida base para el trabajo de cada día
- Hace visible el avance hacia la finalización del incremento del producto, lo que anima a los probadores y desarrolladores.
- Elimina los riesgos asociados a la integración tipo big-bang.
- Un software ejecutable está constantemente disponible en todo el sprint para la prueba, demostraciones o para enseñanza
- Reduce las actividades de pruebas manuales repetitivas
- Ofrece una rápida retroalimentación sobre las decisiones tomadas para la mejorar calidad y pruebas



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Integración Continua

Riesgos y desafíos

- Las herramientas utilizadas para la integración continua necesitan implementarse y mantenerse, ya que se requiere herramientas para las pruebas, herramientas para automatizar el proceso de construcción y herramientas para el control de versiones.
- Se deben definir y establecer los procesos de integración continua
- Automatización de pruebas
 - Requiere especialistas en el equipo
 - Puede ser muy complejo para establecer
- A través de la cobertura de pruebas exhaustivas es esencial para lograr las ventajas de las pruebas automatizadas
- Tendencia a la sobre confianza en las pruebas unitarias y realizan muy pocas pruebas integración / sistema y pruebas de aceptación

Capítulo 1:

Desarrollo ágil de software

1.1 Fundamentos del desarrollo ágil de software

- Desarrollo ágil de software y el Manifiesto Ágil
- Enfoque de equipo completo
- Feedback temprano y frecuente

1.2 Aspectos de enfoques ágiles

- Enfoques de desarrollo ágil de software
- Creación de historias de usuario colaborativas
- Retrospectivas
- Integración continua
- Planificación de la entrega y planificación de la iteración

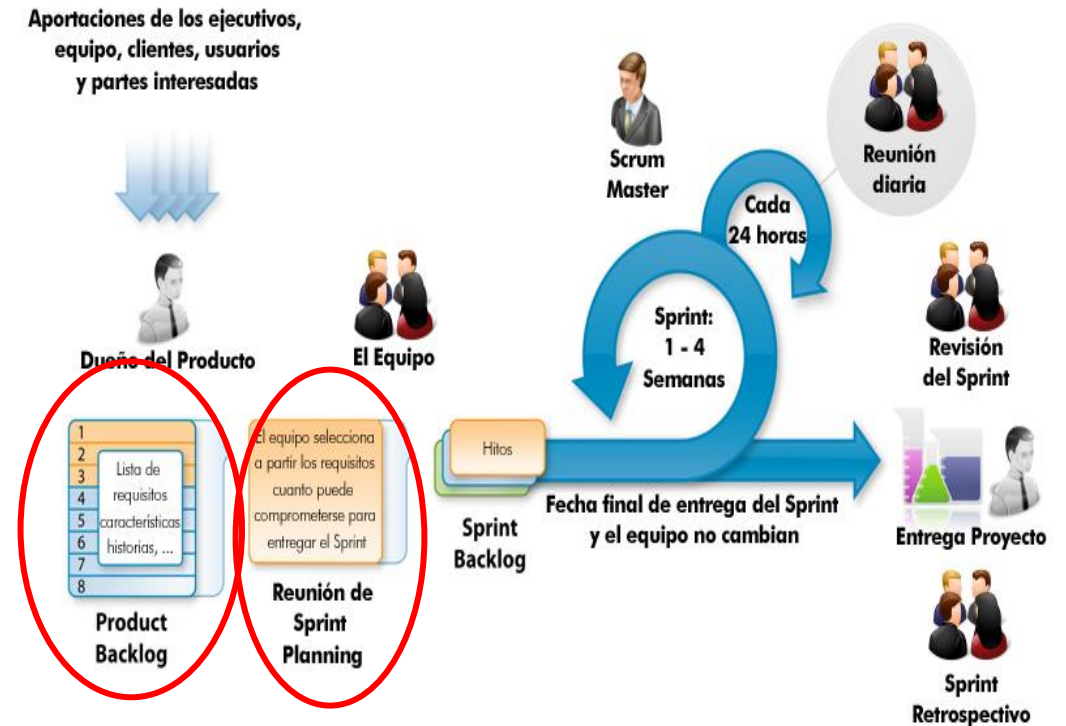


1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de entregas e iteración

¿Qué es la planificación?

- La planificación es una actividad constante que se realiza tanto para los enfoques tradicionales o ágiles
- En los ciclos de vida ágiles tenemos planificación de entregas (Product Backlog) y planificación de iteración (Sprint Backlog)



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

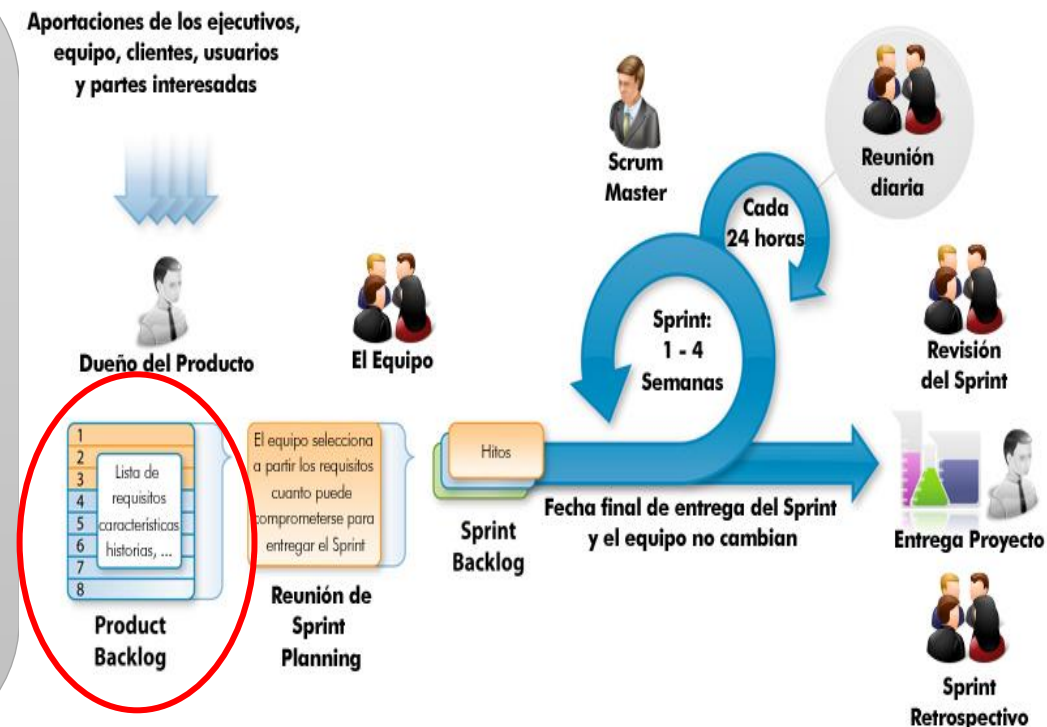
Planificación de entrega

Reunión de planificación de entregas (Product Backlog)

La planificación de la entrega prevee la entrega del producto, a menudo unos cuantos meses del inicio del producto.

La planificación tiene como objetivo

- Definir y redefinir el backlog del producto
- Redefinir historias de usuarios (se independizan las muy grandes, o se revisan definiciones)
- Proporcionar la base para el enfoque de las pruebas y el plan de pruebas.
- Realizar una planificación de alto nivel



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de entrega

- En esta planificación se :
 - Establecen y priorizan las historias de usuario para la entrega, en colaboración con el equipo
 - Se identifican los riesgos del proyecto y de calidad
 - Se realiza estimación del esfuerzo (alto nivel)



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de entrega

Los probadores entregan valor en la planificación de la entrega mediante:

- La definición de historias de usuario testeables, incluyendo criterios de aceptación
- Participan en el análisis del proyecto y en el análisis de los riesgos de calidad
- Estiman del esfuerzo de pruebas asociado a las historias de usuario
- Definen los niveles de pruebas necesarios
- Planifican pruebas para que se realizarán en la entrega

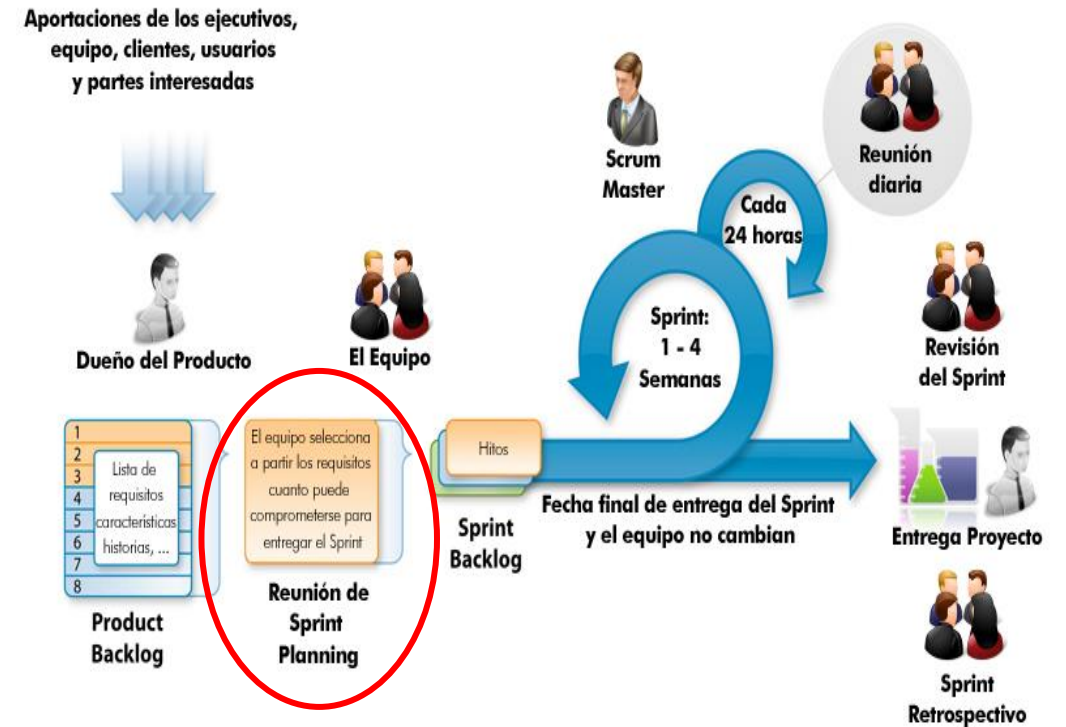


1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de la iteración

La planificación de la iteración:

- Se inicia después de la planificación de la entrega
- La planificación de la iteración abarca sólo una iteración (sprint) y se refiere sólo al backlog de esta.



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

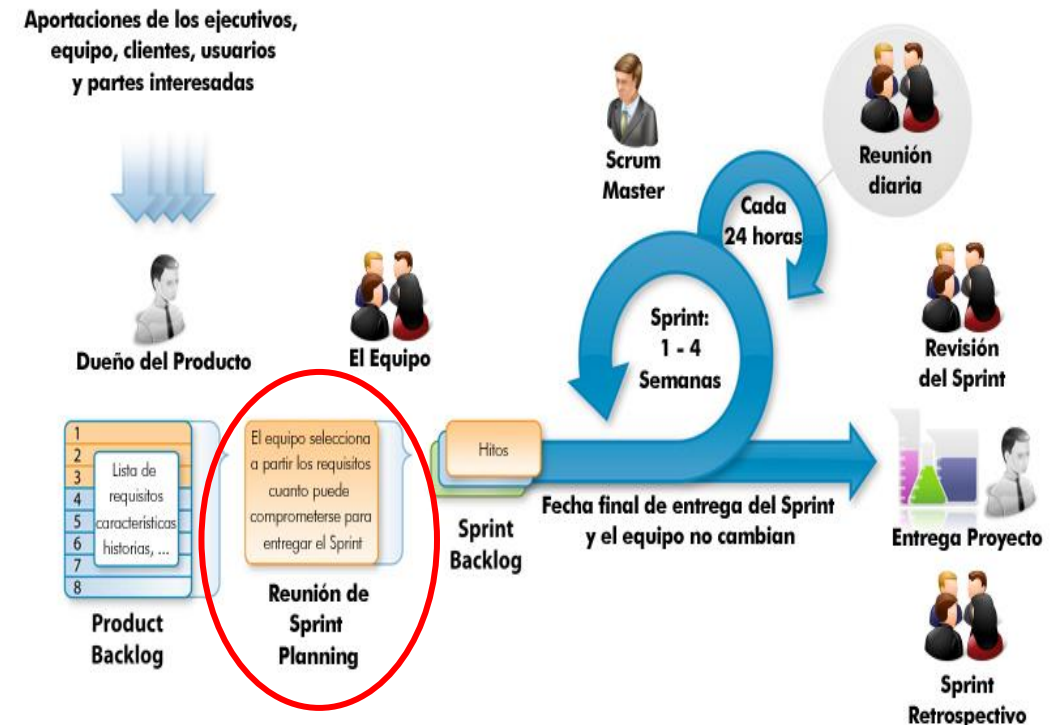
Planificación de la iteración

Tareas que se realizan en la planificación de la iteración:

- El equipo selecciona de historias de usuario a partir del backlog de entrega priorizado
- Elabora las historias de usuario
- Realiza un análisis de riesgo de las historias de usuario
- Estima el trabajo necesario para cada historia de usuario
- Las HU se dividen en tareas, las que realizarán las personas del equipo.

¿Qué pasa si no se entiende o es muy vaga una historia de usuario?

- El equipo puede no aceptarla y seguir con la siguiente historia priorizada.
- El representante del negocio (PO) es el responsable de aclarar las dudas de la HU

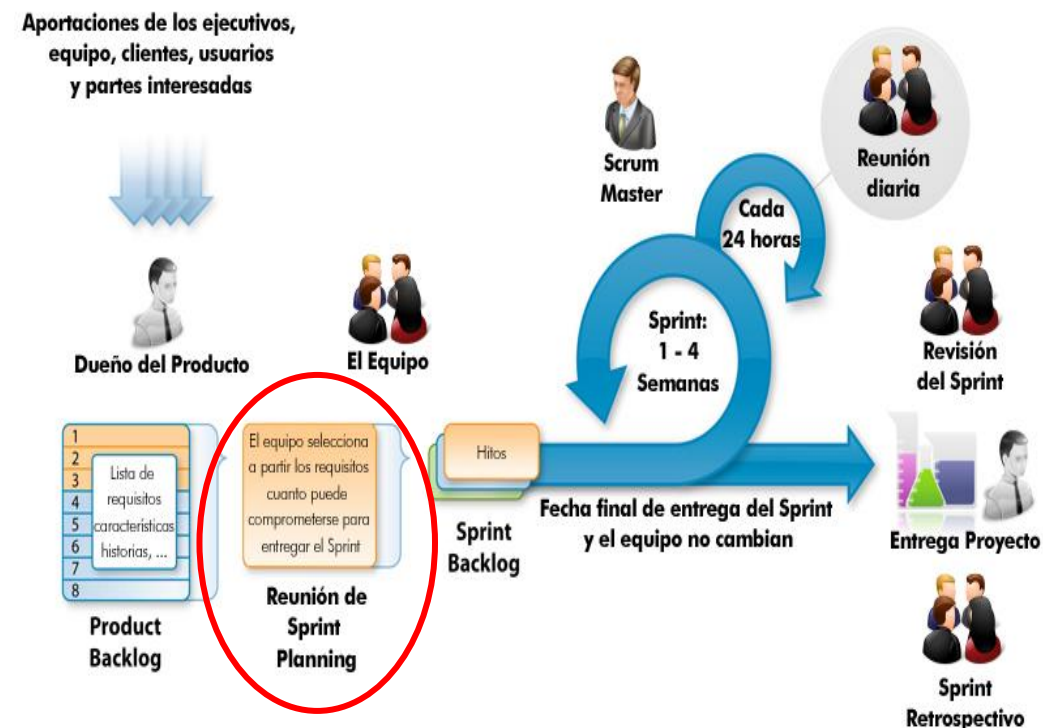


1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de la iteración

Los probadores en esta etapa aportan valor en:

- Participar en el análisis detallado de riesgos
- Determinar la testeabilidad de las historias de usuario
- Crear pruebas de aceptación para las HU
- Dividir historias de usuario en tareas, especialmente las tareas de pruebas
- Calcular el esfuerzo de pruebas para todas las tareas de prueba
- Identificar los aspectos funcionales y no funcionales del sistema a probar
- Dar soporte y participar en la automatización de pruebas en los diferentes niveles de prueba



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de entregas e iteración

La planificación de entrega puede cambiar durante el proyecto

- Las historias de usuarios individuales en el backlop del producto o de la entrega pueden cambiar debido a factores internos o externos
 - factores internos: la velocidad, problemas técnicos, capacidades de entrega
 - factores externos: nuevos mercados y oportunidades, nuevos competidores , amenazas de negocio a consecuencia de un cambio de objetivos y/o plazos



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de entregas e iteración

La planificación de la iteración también puede cambiar

- La planificación de la iteración puede cambiar durante la iteración o sprint.
- Ejemplo: una historia de usuario estimada simple en la planificación, es mucho más compleja. Por lo que se puede definir dejarla para otra iteración o priorizarla y dejar otras HU para una siguiente iteración.



1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de entregas e iteración

Los cambios son un desafío para los tester, por lo que deben:

- Entender el panorama general de la entrega para la planificación de las pruebas
- Deben tener una base de pruebas adecuadas para cada iteración
- La información necesaria para las pruebas debe estar disponible desde lo antes posible, ideal al inicio.
- Los cambios deben ser acogidos de buena forma/abrazados (es un principio ágil)
- Debe tomar decisiones cuidadosas sobre las estrategias y la documentación de las pruebas.





1.2 Desarrollo ágil de software- Aspectos de los Enfoques Ágiles

Planificación de entregas e iteración

- La planificación no sólo para las pruebas, sino también para el desarrollo
- Temas específicos que deben abordarse en relación a las pruebas:
 - El alcance de las pruebas , la extensión de las pruebas para las áreas dentro del alcance, los objetivos de prueba , incluyendo razones de estas decisiones
 - Definición de los miembros del equipo que realizarán a cabo las actividades de las pruebas.
 - Definir entorno de prueba y datos de prueba
 - Definir los tiempos, las secuencias, las dependencias y los requisitos previos para las actividades de prueba funcionales y no funcionales. (Cuándo se ejecutarán las pruebas de regresión, que funcionalidades dependen de otras funcionalidades)
 - Los riesgos del proyecto y la calidad que debe ser abordada.
 - Considerar el esfuerzo necesario para llevar a cabo las actividades de las pruebas.

Capítulo 1:

Desarrollo ágil de software

1.1 Fundamentos del desarrollo ágil de software

- Desarrollo ágil de software y el Manifiesto Ágil
- Enfoque de equipo completo
- Feedback temprano y frecuente

1.2 Aspectos de enfoques ágiles - Ejercicios

- Enfoques de desarrollo ágil de software
- Creación de historias de usuario colaborativas
- Retrospectivas
- Integración continua
- Planificación de la entrega y planificación de la iteración





1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 1

¿Qué significa una estimación alta para una historia de usuario?

1. Una estimación alta generalmente significa que la historia es más relevante
2. Una estimación alta generalmente significa que la historia se entiende bien y debe comenzarse de inmediato
3. Una estimación alta generalmente significa que no se entiende bien o debe dividirse en historias más pequeñas
4. Una estimación alta generalmente significa que es de suma importancia y debe construirse primero



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 2

Usted es un probador que trabaja en un proyecto Agile que está construyendo un sistema de control de stock minorista. El propietario de la empresa ha definido la siguiente historia de usuario:

Como usuario

Quiero ser capaz de gestionar los códigos de stock

Para que pueda eliminar cualquiera que esté obsoleto.

Basado en lo anterior, ¿Cuál sería su próxima tarea?

1. La historia de usuario se construye correctamente, por lo que empezaría a definir casos de prueba para la historia de usuario
2. La historia de usuario se construye correctamente, por lo que trabaja con el desarrollador y el propietario del negocio para definir los criterios de aceptación
3. La historia de usuario no se construye correctamente, por lo que trabaja con el propietario del negocio para ponerla en el formato correcto de Given- When-Then.
4. La historia de usuario no se construye correctamente, por lo que trabaja con el propietario del negocio para aclarar las condiciones.



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 3

Considere un proyecto ágil para desarrollar un sitio web destinado a la venta de bienes. Se pueden extraer estadísticas sobre las ventas del día.
Considere la siguiente historia de usuario de este proyecto ágil:

Como gerente de ventas

Quiero poder guardar el informe de ventas diarias

Para poder compararlo con las ventas del día siguiente.

Para esta historia de usuario se definen los siguientes criterios de aceptación:

- El director de ventas debe recibir un informe de ventas diario.
- El director de ventas puede guardar el informe de ventas al hacer clic en el botón 'Guardar'.

Basado sólo en la información dada, ¿Cuál de los siguientes es más probable ser verdad?

1. Una conversación entre el probador, el desarrollador y el propietario del producto podría resaltar la necesidad de seleccionar un lugar donde guardar el informe.
2. Durante una conversación entre el probador, el desarrollador y el propietario del producto, los criterios de aceptación podrían cambiarse para estar orientados al usuario en lugar de orientados a la interfaz gráfica de usuario.
3. La historia de usuario debe ser reemplazada por el criterio de aceptación, y el criterio de aceptación por la historia
4. La historia de usuario y sus criterios de aceptación están suficientemente detallados.



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 4

¿Los probadores están involucrados en la planificación de iteraciones y, especialmente, añaden valor en cuál de las siguientes actividades?

- (i) Participar en el análisis detallado de riesgos de historias de usuarios.
- (ii) Crear pruebas de aceptación para las historias de usuarios.
- (iii) Calcular el esfuerzo de pruebas para todas las tareas de prueba.

- 1. i y ii
- 2. i , ii y iii
- 3. ii y iii
- 4. Sólo ii



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 5

¿El objetivo de la reunión de retrospectiva es?

1. Discutir, entre otras cosas, qué fue lo exitoso en el equipo sobre el proceso
2. La agenda es la misma que para una reunión diaria de pie.
3. Discutir, entre otras cosas, qué fue exitoso en el equipo sobre el producto.
4. Evaluar e identificar miembros de bajo rendimiento.



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 6

¿Qué significan las C en el concepto 3C considerado durante la creación de la historia del usuario?

1. Tarjeta, Confirmación, Configurable
2. Colaboración, Continua, Confirmación
3. Colaboración, Conversación, Confirmación
4. Tarjeta, Conversación, Confirmación



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 7

Se le ha dado la siguiente historia:

"Como maestro en línea, quiero que el sistema calcule el “pasó/falló” final para cada estudiante en mi clase”.

También se le han dado los siguientes criterios de aceptación:

1. Un estudiante tiene que calificar con por lo menos 25 puntos para poder “pasar” el examen (pasó).
2. El máximo que se puede obtener es 40 puntos.

¿Cuál de los siguientes proporciona los casos de prueba de aceptación válidos adecuados para que el estudiante pueda aprobar el examen?

1. puntuación= 30
2. puntuación=18
3. puntuación=42
4. puntuación = 24



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 8

En un proyecto Agile, ¿cuál de los siguientes ayuda a verificar si alguna compilación no puede codificar errores de compilación?

1. Prueba alfa
2. Marco de integración continua
3. Prueba Beta
4. Pruebas de integración



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 9

¿Cuál de los siguientes evitaría durante las retrospectivas?

1. Blame game (juego de acusaciones)
2. Identificar avances importantes
3. Identificar pequeñas mejoras
4. Dar la oportunidad a todos de hablar



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 10

¿Cuál de los siguientes **NO** es un beneficio de la integración Continua?

1. La integración continua proporciona retroalimentación rápidos sobre las decisiones tomadas para mejorar la calidad y las pruebas
2. Integración continua elimina los riesgos del cronograma asociados con la integración big-bang
3. La integración continua proporciona la confianza de que el trabajo de desarrollo de cada día está basado en fundamento sólido.
4. La integración continua aumenta las actividades repetitivas de prueba manual.



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 11

En un proyecto Agile, ¿qué significa una historia de usuario?

1. Las historias de usuarios se escriben para capturar experiencias desde la perspectiva de los usuarios finales
2. Las historias de usuarios se escriben para capturar los requisitos desde la perspectiva de los interesados
3. Las historias de los usuarios se escriben para capturar los requisitos desde la perspectiva del líder del proyecto.
4. Las historias de usuarios se escriben para capturar los requisitos desde la perspectiva de los desarrolladores, probadores y representantes comerciales



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 12

Durante una sesión de plannig poker, los interesados y los desarrolladores otorgan 5 puntos a una historia de usuario, pero el probador otorga 21 puntos. ¿El equipo discute por qué el probador se siente así? El probador hace preguntas para aclarar los requisitos y destaca que se requerirán más esfuerzos de prueba para garantizar la coherencia. ¿Cuál de los siguientes describe el papel del probador en esta situación?

1. Colaboración activa con desarrolladores y partes interesadas del negocio para aclarar los requisitos, especialmente en términos de comprobabilidad, coherencia e integridad.
2. Medición y reporte de cobertura de prueba en todas las dimensiones de cobertura aplicables
3. Participando proactivamente en retrospectivas de equipo, sugiriendo e implementando mejoras
4. Configurar, usar y administrar entornos de prueba y datos de prueba



1.1 Desarrollo ágil de software- Fundamentos

EJERCICIO 13

¿Cuál de las siguientes es una técnica ágil que verifica regularmente la efectividad del proceso de calidad, busca la causa raíz de los problemas y sugiere ensayos de nuevos enfoques para mejorar la calidad?

1. Historias de usuarios
2. Backlog
3. Sprints
4. Retrospectivas

1.1 Desarrollo ágil de software- Fundamentos

Respuestas

1. 3
2. 4
3. 1
4. 2
5. 1
6. 4
7. 1
8. 2
9. 1
10. 4
11. 4
12. 1
13. 4



Curso ISTQB® Foundation LEVEL AGILE TESTER

