

Testing I

Profesor: Daniel Tejerina

Módulo 1: Fundamentos de Testing y Gestión de Defectos

Sesión 1: Primeros Pasos **Octubre 19 de 2021**

Calidad del Software

"La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario" (IEEE Std 610, 1990).

Testing

El testing es el proceso de ejecución de un programa con la intención de encontrar errores.

QA

La principal función es probar los sistemas informáticos para que funcionen correctamente de acuerdo a los requerimientos del cliente, documentar los errores encontrados y desarrollar procedimientos de prueba para hacer un seguimiento de los problemas de los productos de forma más eficaz y eficiente.

7 principios del Testing

1. La prueba muestra la presencia de defectos, no su ausencia.
2. La prueba exhaustiva es imposible.
3. La prueba temprana ahorra tiempo y dinero.
4. Los defectos se agrupan.
5. Cuidado con la prueba del pesticida.
6. La prueba se realiza de manera diferente según el contexto.
7. La ausencia de errores es una falacia.

Beneficios de la Independencia de la prueba

- Los probadores pueden encontrar errores diferentes.
- El probador puede cuestionar las suposiciones hechas durante la especificación e implementación del sistema.

Desventajas de la Independencia de la prueba

- El desarrollador puede perder sentido de responsabilidad con respecto a la calidad.
- El probador puede ser visto como cuello de botella.
- El probador puede carecer de información importante.

Mesa de tres patas:

- Analista de negocio. Detecta los factores claves del negocio y es el intermediario entre el departamento de sistemas y el cliente final.
- Desarrollador de software
- QA

Reunión de los 3 amigos:

(En metodologías ágiles) Es una sesión en la que participan estos tres roles y cada uno de ellos da su punto de vista respecto al software que está bajo desarrollo.

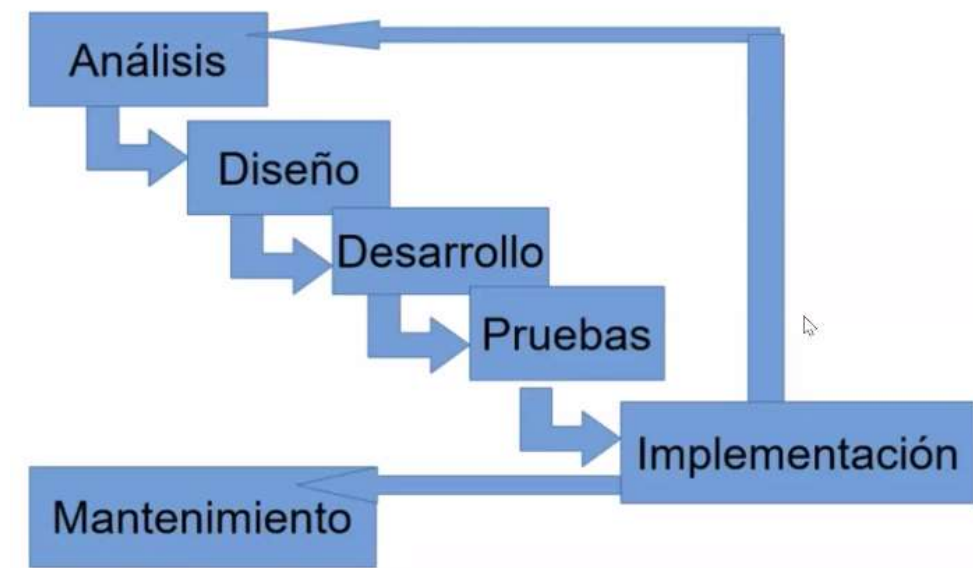
Sincrónico

Verificar vs validar

Verificar: Revisar si el producto se está construyendo correctamente. Se revisa con relación a los requerimientos.

Validar: Verificar si el producto es adecuado para el cliente. Se verifica con relación a la necesidad real del usuario.

Ciclo de vida del software (tradicional):



Sesión 2: Gestión de Defectos

Octubre 21 de 2021

Ciclo de vida de un defecto:

→ Nuevo/inicial → Asignado → En progreso → Corregido → En espera de verificación → En verificación → Verificado → Cerrado



Asignado:

- Duplicado. Se repite o existe otro con la misma causa raíz.
- Devuelto o rechazado. Se rechaza o se solicita más información.
- Diferido. No es prioritario y se pospone para otra versión.

En verificación:

- Reabierto. "La prueba de confirmación indica que el defecto no se ha solucionado".

Proceso de Gestión de Defectos

- Detectar
- Registrar
- Investigación y seguimiento
- Clasificación / Resolución

Escribir un informe de defectos:

- Los bugs deben tener identificadores únicos
- Una falla debe ser reproducible para reportarla
- Ser específico
- Reportar cada paso realizado para reproducirlo

Partes de un informe de defectos:

- ID
- Título
- Descripción
- Fecha del informe del defecto
- Autor
- Identificación del elemento de prueba
- Versión
- Entorno
- Pasos a reproducir
- Resultado esperado
- Resultado obtenido o actual
- Severidad: Bloqueado / Crítico / Alto / Medio / Bajo / Trivial
- Prioridad: Alta / Media / Baja
- Estado del defecto: Nuevo / Diferido / Duplicado / Rechazado / Asignado / En progreso / Corregido / En espera de verificación / En verificación / Verificado / Reabierto / Cerrado
- Referencias
- Imagen

Sincrónico

- Error
- Defecto
- Fallo
- Falso positivo.
- Falso negativo. El sistema no debería pasar la prueba y la pasa.
- **Bug triage:** Ceremonia de agile para clasificación y asignación de bugs.
- known issues: Defectos conocidos

Herramientas:

- Ej: Mantis (para trazabilidad de bugs?)