Viernes 25 de Marzo del 2022

* Fundamentos de las pruebas de software
* Pleanecion y metodología
* Diseñor de pruebas funcionales
* Modelos tdd – bdd
* Automatizacion de pruebas funcionalesdd

Sabado 26 de Marzo Del 2022

Pricipio1 : Las pruebas revelan la presencia de bugs, no la ausencia de ellos

Principio 2: Es imposible probarlo todo

Principio 3: cuando antes se comience a probar… mejor

Principio 4: las aglomeración de defectos. Los bugs siempre van en pandilla.

Principio 5: es necesario ajustar continuamente el plan de pruebas.

Principio 6: Las pruebas se deben adaptar a necesidades especificas

Principio 7: Software 100% libre de errores NO Existe

Analsis funcional -> Revision – P estáticas ---- Diseño Pruebas funcionales aceptación (arreglos ortogonales, y covering array).

Fundamentos de Pruebas

Perfil de conocimientos y habilidades del tester

Objeticos y aportes para proyectos de desarrollo

Importancia de las pruebas

Transversalidad de las pruebas

Principios, tipos de pruebas, proceso de pruebas

Producto

Backend -> \*Unit

Front-End -> Movimineto del producto

Tipos de pruebas:

1.Estrategia:

**Pruebas de unidad**

“vefiricar en la aplicación la funcionalidad de registro de usuarios.”

**Pruebas de integración**

“Se ofrece una solución a un cliente el cual tiene 3 componentes:

1. Un portal de acceso
2. Una herramienta para E-learning.
3. Un sistema de indicadores de gestión para medir la utilización del portal y del E-learning

Una de las funcionalidades compartidasd entre aplicaciones es:

Cuando se crea un nuevo curso automáticamente se creara un foro en el portal para que los integrantes del curso participen de el.

Fuera de verificar que los modulos(creación de curso y foro) de cada componente funcionen correctamente de acuerdo a las especificaciones,se debe probar que la integración de los componentes(modulos y productos) operen adecuadamente como solucion”.

**Pruebas de sistemas**

“Llegada camión a desembarque.”

**Pruebas de regression**

“Una vez que se ha encontrado un defecto en el Modulo de indicadores y el area de Desarrollo lo ha solucionado, se verifica la NC y se realizan las pruebas entre los otros modulos(Portal y e-leraning) para asegurar que no se ingresaron mas NC,s por la corrección del BUG reportado”.

**Pruebas alpha y beta**

Alpha -> realizadas antes de producción,

Beta -> pruebas realizadas normalmente con una muestra poblacional para probar la app.

**Pruebas de aceptación**

Realizadas con los clientes y define su aceptación del software, En este nivel de pruebas se debe especificar los siguientes ítems:

Quien define el nivel de la prueba de aceptación

Quien plantea y escribe los instrumentos de pruebas

Quien eejcuta las pruebas

Cual es el criterio de éxito/falla para la prueba de aceptación

2. Objetivos:

Pruebas funcionales

**Pruebas de seguridad**

La prueba de seguridad intenta verificar que los mecanismos de protección incorporados en el sistema lo protegerán, de hecho, de accesos impropios.

Durante la prueba de seguridad, el responsable de la prueba desempeña de la prueba desempeña el papel de un individuo que desea entrar al sistema.

**Pruebas de desempeño (carga)**

Estas pruebas son dirigidas a evaluar la conformidad de un sistema o componente con requerimientos de desempeño especifico.

Observan el comportamiento de una aplicación al realizar un numero especifico de transacciones durante el tiempo que dura la prueba

Normalmente esto se lleva a cabo usando una herramienta de prueba automática para simular un gran numero de usuarios.

**Pruebas Resistencia (stress)**

La prueba de resistencia ejecuta un sistema de forma que demande recursos en cantidad, frecuencia o volúmenes anormales.

Es el tipo de prueba que enfrenta el sistema a situaciones anormales de ejecución y determina la resistencia del sistema de acuerdo al tope o un liminte

Estrés

Por ejemplo:

Diseñar preubas especiales que genera diez iterrupciones por segundo, cuando las normales son una o dos;

Incrementar las frecuencias de datos de entrada en un orden de magnitud con el fin de comprobar como responde las funciones de entrada;

Ejecutar casos de prueba que puedad dar problemas con un sistema

**Pruebas Volumen**

Prueba con la cual se confirma que cualquier valor que pudiera llegar a ser grande en un cierto plazo(como por ejemplo logs, archivos de datos, etc…)

Puede ser acomodado por el programa y no causra que el programa se reviente.

**Pruebas Concurrencia.**

Prueba multiusuario encaminada a determinar los efectos de acceder el mismo código de aplicación, modulo o registros de una base de datos.

Identifica y mide:

El nivel de bloqueo, locking

Bloque mutuo entre usuarios, deadlocking

Uso de hilos simples, single.threaded

Semáforos de bloqueo, locking shemaforos