Testing

Proceso destructivo, obj: encontrar defectos para posteriormente resolverlos y contribuir a la calidad del sw

Asumimos que los defectos existen.

Cuanto testing es suficiente?

- Umbral
- Criterio económico

Error vs Defecto (Pregunta de final)

Son fallas en un producto de sw. Depende en que momento se encuentra o se produce

Cuando encontramos la falla en la misma etapa que la origino, es un error

Cuando trasciende la etapa, es un defecto

El testing busca encontrar DEFECTOS

El obj del testing no es encontrar errores, el testing se ejectua posteriormente a la implementacion y brinda una retroalimentacion.

La idea es encontrar las fallas cuando son errores que son mas baratos.

Proceso de Testing

Planificacion y control -> Analisis y diseño -> Ejecucion -> Evaluacion y Reportes

Niveles de testing

1- Testing unitario: busca proveer un mecanismo de verificacion de comprension de la funcionalidad. Garantizo que la funcionalidad cumple con el requerimiento dado. Lo lleva adelante el DESARROLLADOR

2- Testing de integración: buscamos encontrar defectos. prueba las interfaces entre los componentes del sw. bottom up, top down, sandwich. Lo lleva adelante el TESTER

3- Testing de Sistema: Probamos una funcionalidad en su totalidad. Ejecutamos un escenario con ciertas caracteristicas para encontrar si el resultado coincide con el esperado. Lo lleva adelante el TESTER

4- Testing de Aceptación: No deberiamos encontrar defectos. Es validar que lo que entregamos corresponde a el requerimiento planteado. Es de alguna forma capacitar al cliente en la funcionalidad. Es lo mas similar posible al productivo. Lo lleva adelante el PRODUCT OWNER (en agil) o el USER/CLIENTE

Estrategias

Criterio económico: Maximizar la cantidad de defectos minimizando el esfuerzo requerido

Casos de prueba: Artefacto del testing. Contiene condiciones y pasos que se deben ejecutar para garantizar que el resultado esperado sea o no igual al resultado obtenido y de esa forma encontrar defectos.

Diseñamos casos de prueba economica e inteligentemente (Minimizar casos de prueba y maximizar defectos encontrados)

2 ESTRATEGIAS

Caja blanca: podemos ver el detalle de la implementacion de la funcionalidad (codigo). Diseñar caso de prueba para garantizar la cobertura.

Caja negra: analizo en terminos de entradas y salidas, no conozco la fucionalidad interna.

Elijo los valores con los que ejecuto la funcionalidad y analizo las salidas (resultado obtenido)

Métodos (dentro de las estrategias)

Caja negra

-Basado en especificaciones:

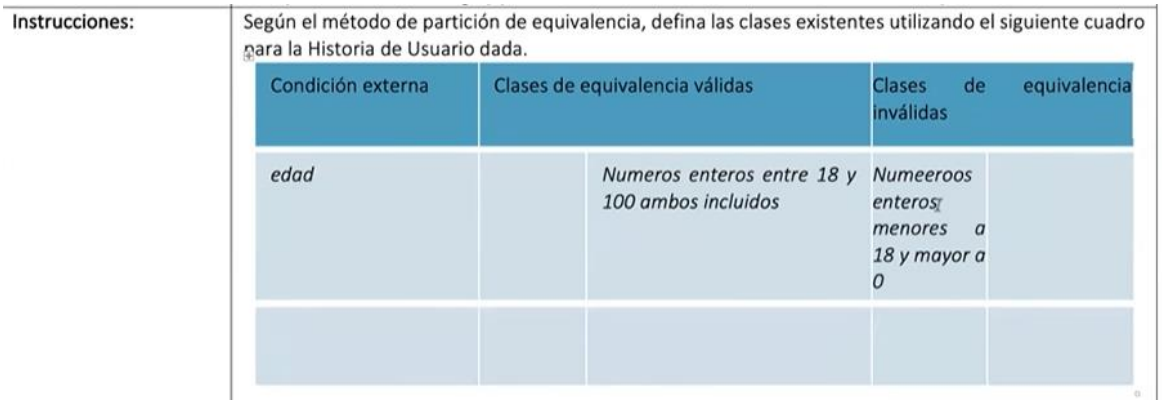
.Partición de equivalencias: analizar las condiciones externas involucradas en la funcionalidad (entradas y salidas).

1) Identificar las clases de equivalencia: Para c/u analizo los subconjuntos de valores posibles que pueden tomar que producen un resultado equivalente. Casos: Rango de valores continuos, valores discretos, selección simple, selección múltiple.

2) Identificar los casos de prueba: tomo de cada condición externa una clase de equivalencia y elijo un valor representativo para formar el caso de prueba. Cualquier valor que tome por clase de equivalencia me va a dar un resultado equivalente por eso se toma un solo valor.

Ejemplo: web de MAE bebidas. te pregunta cuál es tu edad. Condiciones externas: Edad de la persona (entrada), Ingreso/No ingreso (salida). A la edad la podemos dividir en subconjuntos de valores posibles donde cada uno va a producir un resultado equivalente. Por lo menos 2 clases de equivalencia validas -> mayor igual 18 y menor 18. Hay clases de equivalencia no validas, en este caso, texto, vacío, números no enteros, símbolos, etc. La división se basa en los resultados posibles/mensajes de error asociados. Un caso de prueba seria: Ingresar a la web con una edad que supere los 18. Resultado esperado: Ingreso valido. El único valor que elegí fue 18 que está dentro de la clase de equivalencia "mayor igual a 18", no necesito usar más valores dentro de esa clase porque se supone que el resultado es equivalente -> me deja ingresar a la web

Ejemplo del práctico



Instrucciones:	Según el método de partición de equivalencia, defina las clases existentes utilizando el siguiente cuadro para la Historia de Usuario dada.				
	Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas	
	edad		*Numeros enteros entre 18 y 100 ambos incluidos*	*Numeeroos enteros menores a 18 y mayor a 0*	

Id del Caso de Prueba	Prioridad (Alta, Media, Baja)	Nombre del Caso de Prueba	Precondiciones	Pasos	Resultado esperado
			El usuario "juan" esta logueado con permisos de admninistrador. La fecha actual es 15/5/2020	1. El cliente ingresa a la opción "Ingresar" 2. El cliente ingresa "18" en el campo de "edad" 3. El cliente selecciona l aopción ingresar	1. Se muestra el mensaje "Bienvenido al sitio web"

Prioridad (orden que le voy a dar a la ejecución de los casos de prueba para encontrar defectos)
Alta: Camino feliz.
Media: Todo lo que está al medio (lo dijo textual asi xd). Combinaciones de valores que pueden producir fallas/caminos infelices.
Baja: Validaciones, valores no ingresados, no corresponden con el formato.

Nombre: Describe el escenario que estamos probando

Precondiciones: conjunto de valores/características que tiene que tener el contexto para llevar adelante el caso de prueba. Ejemplo: el usuario "juan" esta logueado con permisos de admin. TIENEN QUE SER SIGNIFICANTES PARA EL RESULTADO DEL CASO DE PRUEBA. Tienen que tener VALORES CONCRETOS.

Pasos: conjunto de operaciones numeradas cuyo formato es: 1- El ROL ingresa la opción "x" 2- El ROL ingresa "x" en el campo de "y"… etc

Resultado esperado: salida esperada.

Que es una clase/partición de equivalencia? (Pregunta de final)

Subconjunto de valores que puede tomar una condición externa para el cual, si yo tomo cualquier miembro de ese subconjunto el resultado de la ejecución de la funcionalidad es EQUIVALENTE.

.Análisis de valores límites: Es una variante de la partición de equivalencias. La mayor cantidad de defectos se encuentran en los extremos de los intervalos. A la hora de escribir el caso de prueba, en vez de tomar CUALQUIER valor de la clase de equivalencia, tomamos uno que se encuentra en los límites.

-Basados en la experiencia

.Adivinanza de defectos

.Testing exploratorio