

CLASE 4

REPORTE DE BUGS

REPASAMOS CASOS DE PRUEBAS:

Con lo que vimos la clase pasada, pensemos por qué es importante hacer casos de prueba de forma clara y precisa.

REPASAMOS CASOS DE PRUEBAS:

Con lo que vimos la clase pasada, pensemos por qué es importante hacer casos de prueba de forma clara y precisa.

¿Qué es un caso de prueba?

Es una condición de uso con la que se puede verificar si un desarrollo funciona correctamente

Estructura de un caso de Prueba

A	B	C	D	E	F	G
Nro de Caso	Título	Pre Condición	Datos de Entrada	Paso a Paso	Resultado Esperado	Resultado Obtenido
1	Numero Ingresado Mayor a Cero	Ingresar a la app de CelFar	Numero mayor a Cero	Abrir la app Celfar e ingresar un número mayor a cero en el combo vacío y seleccionar convertir	Que convierta el número ingresado de celsius a fahrenheit	OK

TIPOS DE APLICACIONES WEB

Como ya saben, hay distintos tipos de aplicaciones Web, y cada una tiene su particularidad:

E-Commerce: El sentido de ésta app es la venta de un producto o servicio. La prueba principal está orientada a que un usuario pueda completar el flujo para adquirir dicho producto o servicio

Búsqueda: Son aplicaciones utilizadas para realizar búsquedas de distintos tipos. Lo que tenemos que garantizar es que devuelve correctamente resultados relacionados con la búsqueda que hicimos

Redes Sociales: Son utilizadas para subir fotos, relacionarse, interactuar. Las pruebas realizadas deben poner foco en sus funcionalidades principales

LEAMOS LAS ESPECIFICACIONES DE LA APP:

“El Provinciano”

Provinciano - Especificación

- ¿Cual es la necesidad / Objetivo de la App según la especificación?
- Diseñemos todos los casos de prueba que se nos ocurran (Nos tomamos 10 minutos)
- Hacemos puesta en común de los casos de prueba planteados

VEAMOS DOS CONCEPTOS NUEVOS:

Casos Positivos y Casos Negativos

¿Alguno se anima a decir que piensa que es un caso positivo y un caso negativo?

CASOS DE PRUEBA

POSITIVO

Intenta mostrar que **el producto satisface** sus requerimientos.

- ¿Cómo hago un test positivo?

Armo mi test con el requerimiento que aparece en la especificación.

Ejemplo: Para el carrito pide que se puede ingresar el nombre y el apellido del cliente, hago mi caso para ingresar el nombre Juan y apellido Pérez (un nombre y apellido cualquiera).

CASOS DE PRUEBA

NEGATIVO

El objetivo es **romper el sistema**.

- ¿Cómo hago un test negativo?

Trato de hacer algo contrario a lo que el sistema espere que se haga.

Ejemplo: Para el mismo requerimiento del carrito de nombre y apellido, tratar de poner un número en lugar de un nombre y un apellido. El resultado debería ser que se muestre un aviso de error (nunca debería caerse el sistema e impedir que siga registrando el usuario).

Actividad:

Agrupemos los casos de “**el provinciano**” en
Tests Positivos y Negativos

ESTRUCTURA PARA REPORTAR ERRORES

- ¿Cómo se documenta?
- ¿Qué datos debemos registrar al momento de reportar un Bug?
- ¿Qué datos son esenciales? y ¿cuáles no?

VEAMOS QUÉ DATOS TENEMOS TENEMOS QUE REGISTRAR AL MOMENTO DE REGISTRAR UN BUG

Título: que describa el error

El nombre del tester: En caso que haya preguntas de como reproducir el problema, es importante que los desarrolladores puedan identificar a la persona que reportó el incidente.

Fecha: Es importante capturar la fecha en que el incidente se reportó para poder saber cuánto tiempo se demoró en resolver.

VEAMOS QUÉ DATOS TENEMOS TENEMOS QUE REGISTRAR AL MOMENTO DE REGISTRAR UN BUG

Aplicación y versión: Es importante detallar que aplicación estamos probando, y que versión de la misma se está probando.

Ambiente del Tester: En caso de reportar un error web diríamos que navegador se estaba usando, en caso de ser un reporte de un incidente en un dispositivo móvil diríamos que teléfono/modelo/versión del sistema operativo.

Los pasos: necesitamos comunicar cómo reproducir ese error, algo así como la receta para reproducir el problema. Es necesario detallar bien los pasos para poder reproducir el error y que no ocurra el caso del teléfono descompuesto.

VEAMOS QUÉ DATOS TENEMOS TENEMOS QUE REGISTRAR AL MOMENTO DE REGISTRAR UN BUG

Captura de Pantalla: Al mostrar los pasos de como reproducir, o el resultado obtenido, ayuda mucho poder tener capturas de pantalla que le permitan al desarrollador ver el problema más claramente.

Resultado Obtenido: El error de lo que obtuvimos en la prueba

Resultado Esperado: Lo que deberíamos obtener según la especificación.

Importancia: se suele tomar una escala de valores como bajo, medio, alto, crítico, ... y con esto se registra la importancia que se le debería dar al error, en base a qué tanto impacto puede tener en el usuario si encuentra el error, y en base a la probabilidad que el error aparezca (es un error en algo muy usado, entonces habrá mayor posibilidad de que suceda).

Actividad:

Ejecutar las pruebas de versión 1 de “el Provinciano” (según los casos de pruebas que tenemos) y reportar los Bug’s utilizando la estructura mencionada recientemente (tenemos 15 minutos)

Versión 1 de App “El Provinciano”:

[Provinciano_v1](#)