*Trabajo en clases*

# Testing Exploratorio

* + ¿Que es el testing exploratorio?
  + ¿Como se implementa?
  + ¿Cuando se debería usar y cuando no se debería usar el testing exploratorio?
  + ¿Que importancia tienen las pruebas exploratorias para CI/CD?

***Alumnos:*** Milagro Villafañe, Bouso Gonzalo, Isaías Gimenez, Octavio Biancotti, Milagros Flores, Viviana Avila, Quevedo Diego

Desarrollo

El **testing exploratorio** es un enfoque en el que se aprende sobre la aplicación, se diseñan y ejecutan pruebas simultáneamente con un objetivo específico en mente. A diferencia de otras metodologías, no se siguen casos de prueba predefinidos, sino que el tester decide el camino a seguir en función de lo que va descubriendo sobre la aplicación.

***Elementos clave del testing exploratorio*:**

1. **Objetivo Definido**: Antes de comenzar, se establece un objetivo claro para la sesión, como probar flujos de usuario, integraciones con software externo, o identificar vulnerabilidades de seguridad.
2. **Límite de Tiempo**: La actividad se realiza en sesiones limitadas de tiempo, por ejemplo, 1 hora o 25 minutos.
3. **Flexibilidad y Adaptación**: Aunque se define un objetivo, el tester es libre de explorar diferentes caminos y ajustar su enfoque conforme aprende más sobre la aplicación durante la sesión.
4. **Habilidades del Tester**: Este enfoque requiere testers con habilidades analíticas, pensamiento crítico, creatividad, y curiosidad para detectar fallos complejos y evaluar riesgos.
5. **Documentación**: Aunque no es exhaustiva, es recomendable documentar el objetivo de la sesión, los casos de prueba explorados, y cualquier bug encontrado. La documentación debe ser suficiente para reproducir los errores identificados.

**Beneficios:**

* **Motivación para Testers Experimentados**: Presenta un reto intelectual y pone a prueba las habilidades del tester, lo que puede ser más motivador que seguir casos de prueba predefinidos.
* **Mejora Continua**: A medida que avanza la sesión, el conocimiento adquirido sobre la aplicación permite diseñar y ejecutar pruebas de mayor calidad.

El testing exploratorio, cuando se realiza correctamente, puede ofrecer excelentes resultados en la detección de fallos, pero depende en gran medida de la experiencia y habilidades del tester.

Escala de tiempo

Descripción generada automáticamenteç

***Cómo ejecutar las pruebas exploratorias***

Las pruebas exploratorias no deben ejecutarse de forma descuidada. Existen pasos específicos a seguir para lograr los mejores resultados. La forma de ejecutar las pruebas también se conoce como Ciclo SBTM (siglas en inglés de “Gestión de Pruebas Basadas en Sesiones”). El siguiente proceso es una guía paso a paso sobre cómo realizar las pruebas:

***Paso 1:*** ***Inducir una categoría de errores***

Crear una clasificación de errores categorizando las fallas o problemas comunes experimentados en proyectos anteriores. Por ejemplo, si una página se cae continuamente, hay que clasificar ese problema.

Examinar y revisar la raíz del problema a través de un análisis completo de causa raíz de la falla.

Ubicar los riesgos y formular diseños e ideas para probarlos.

***Paso 2:*** ***Test Charter***

El Test Charter consiste en saber en qué hay que centrarse, qué hay que probar y cómo probarlo.

Las ideas de prueba marcan el inicio de todo el proceso de prueba y dan significado a lo que es la prueba exploratoria y cómo puede utilizarse.

***Paso 3:*** ***Time Box***

La duración recomendada es de entre 45 y 90 minutos.

Durante este tiempo el tester no puede ser interrumpido.

El Time Box puede superar los 45 minutos o ser inferior a 45 minutos.

Este método permite a los testers observar la respuesta del sistema y obtener un resultado preciso.

***Paso 4: Revisión de Resultados***

* Los resultados son comprobados y revisados detalladamente en esta sesión.
* Los defectos se evalúan y se comprueban adecuadamente.
* Las áreas de cobertura son analizadas y escaneadas completamente.
* Los testers aprenden de las pruebas en base a los resultados obtenidos.

***Paso 5: Reporte de pruebas y resumen***

* Se reporta y recopila el resultado de la prueba.
* Exploración para confirmar si es necesario realizar otra prueba o no.
* Exploración para comprobar si se han identificado todos los errores.
* Correlacionar los resultados con el Test Charter.

***Cuándo se deben usar***

* Son adecuadas cuando se necesita aprender sobre un producto o aplicación de forma rápida y brindar comentarios inmediatos. También son útiles cuando no se sabe cuál será la próxima prueba que se debe realizar, o cuando se quiere aprender y aislar un defecto en particular.

***Cuándo no se deben usar***

* No son recomendables cuando aplicarlas sería redundante y no aportaría información extra ni un impacto significativo en la calidad del trabajo. En estos casos, una prueba regular basada en scripts podría ser más efectiva. También puede ser más útil esperar hasta que el software tenga su funcionalidad principal.

El testing exploratorio no es recomendable en pruebas de cumplimiento, ya que en este caso las pruebas planificadas son más relevantes. En las pruebas de cumplimiento, los testers deben seguir necesidades específicas, como mandatos gubernamentales.

cuándo no se deben usar pruebas exploratorias

Las pruebas exploratorias por sí solas no pueden ofrecer una cobertura adecuada y los equipos no deben intentar realizarlas, a menos que hayan alcanzado unos hitos iniciales

***¿Qué es el testing en CI/CD?***

El testing en CI/CD puede definirse como la práctica de probar o testar el software de manera automatizada durante sus procesos de desarrollo para garantizar su calidad y funcionamiento.

Es importante aclarar la importancia de la automatización de las pruebas o test en las labores de CI/CD, debido a que esta opción es ideal para los procesos repetitivos, llegando a alcanzar mejores resultados en menos tiempo y sin los errores humanos que pueden darse cundo estas actividades se llevan a cabo de forma manual.

De modo que, si se quieren aprovechar las ventajas de la implementación de procesos de integración y entrega continua, es necesario realizar labores de testing en CI/CD de manera automatizada para los proyectos de software.

Las pruebas exploratorias son fundamentales en el contexto de CI/CD (Integración Continua y Entrega Continua) por varias razones clave:

**1. Detección de Errores:** Permiten identificar errores que pueden no ser capturados por pruebas automatizadas. Esto es crucial, ya que los errores pueden surgir en escenarios no previstos durante el desarrollo[7][8].

**2. Mejora de la Cobertura de Pruebas:** Estas pruebas complementan las pruebas automatizadas al ofrecer una cobertura más amplia, ya que los testers pueden explorar la aplicación de manera creativa y adaptativa, lo que ayuda a descubrir problemas en la lógica o en la interfaz que no se habrían considerado en pruebas predefinidas[7][8].

**3. Aceleración del Ciclo de Desarrollo:** Al integrarse en el pipeline de CI/CD, las pruebas exploratorias permiten a los equipos reaccionar rápidamente ante problemas, lo que reduce el tiempo de comercialización y mejora la calidad del software lanzado[1][5].

**4. Enfoque en la Experiencia del Usuario:** Estas pruebas se centran en la usabilidad y la experiencia del usuario, lo que es vital para asegurar que el producto final no solo funcione, sino que también sea intuitivo y satisfactorio para los usuarios finales[8].

En resumen, las pruebas exploratorias son una herramienta valiosa que complementa las pruebas automatizadas, mejorando la calidad y la eficiencia del proceso de desarrollo en entornos CI/CD.