**Propósito de la actividad**

Analizar los fundamentos que componen las pruebas y el proceso para su  
desarrollo.

Practica lo que aprendiste

I. Elabora un mapa mental del proceso de pruebas, en el que incluyas las  
etapas que lo componen y las principales actividades que se realizan en  
cada una de éstas.

Proceso de pruebas

* **Identificar defectos**
* **Aumentar el nivel de calidad del software**
* **Facilitar información para la toma de decisiones**
* **Evitar la aparición de defectos**

* **Revisar la base de pruebas, Requisitos e Informes de Análisis de Riesgo**
* **Identificar y dar prioridad a las condiciones de la prueba, con base en el análisis de los elementos, la especificación y comportamiento y la estructura del sistema**
* **Diseñar y ordenar los casos de prueba**
* **Identificar los datos de prueba**
* **Diseñar la configuración del entorno de las pruebas**
* Verificar que el entorno de las pruebas hayan sido debidamente configurado
* Implementar los casos de prueba
* Registrar los resultados de la ejecución

Planificación y Control

Ejecución

Análisis y Diseño

Proceso de Pruebas

Actividades de Cierre

Evaluación de los criterios de Salida

* Comprobar los resultados con los valores previstos en la planificación de la prueba
* Evaluar si se requiere más pruebas
* Elaborar un resumen de las pruebas para tu equipo de trabajo y cliente
* Comprobar que documentos han sido entregados
* Cerrar los informes de incidencias
* Documentar cuantos usuarios aceptaron el sistema
* Archivar los productos de soporte, entorno y la infraestructura para usarlo en futuras pruebas
* Utilizar la información recopilada para mejorar la madurez de las pruebas

II. De los siguientes enunciados, identifica cuáles sí son principios de las  
pruebas y anota una X al lado.

\_\_\_\_\_\_Diseñar y ordenar los casos de prueba  
\_\_\_x\_\_Las pruebas se realizan al inicio de vida de los productos  
\_\_\_\_\_\_Verificar que el entorno haya sido debidamente configurado  
\_\_\_x\_\_Actualizar las pruebas para detectar nuevos errores  
\_\_\_\_\_\_Documentar cuántos usuarios acepta el sistema

Actividad  
III. Acomoda los tipos de pruebas en el nivel correspondiente de acuerdo al  
Modelo V.  
Pruebas de sistema / Pruebas de aceptación / Pruebas de componente /  
Pruebas de integración

Nivel 1: \_\_\_\_\_Pruebas de aceptación \_\_\_\_\_  
Nivel 2: \_\_\_\_\_Pruebas de sistema\_\_\_\_\_\_\_  
Nivel 3: \_\_\_\_\_Pruebas de integración\_\_\_\_\_\_  
Nivel 4: \_\_\_\_\_Pruebas de componente\_\_\_\_\_\_\_

IV. Completa las oraciones correctamente utilizando las palabras del  
recuadro.

Pruebas de sistemas, los clientes o usuarios, pruebas de integración,  
requisitos funcionales y no funcionales del sistema, pruebas de  
aceptación, en los requisitos de los componentes, en la arquitectura del  
sistema, pruebas de componentes.

a) Las \_\_\_pruebas de componentes tienen como objetivo localizar defectos y probar el funcionamiento de los módulos del software.

b) Las \_\_\_pruebas de sistema sirven para comprobar el funcionamiento  
de un producto en general.

c) Las \_\_\_\_pruebas de integración verifican las interfaces entre los  
componentes y las interfaces entre varios sistemas.

d) Las \_\_\_\_pruebas de aceptación tienen por objetivo generar confianza en  
el sistema o producto.

e) Las pruebas de componentes están enfocadas en  
los requisitos de los componentes.

f) Las pruebas de sistemas deben incluir requisitos funcionales y no funcionales.

g) Las pruebas de integración deben planearse con base  
a la arquitectura del sistema o las tareas del sistema.

h) Las pruebas de aceptación son responsabilidad de  
los clientes o usuarios.

Actividad  
V. Elabora el siguiente cuadro comparativo, especificando las principales  
características de las siguientes pruebas.

**Tipo de prueba**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipos de pruebas** | **Cuándo se aplican** | **Qué verifican** | **¿Tienen pasos para ejecutarse?**  **¿Cuáles?** |
| Funcionales | En cualquier nivel del proceso | Que cada función del software operen según sus especificaciones. | 1. Identificar los requisitos funcionales 2. Diseñar los casos de prueba 3. Preparar el entorno de prueba 4. Ejecutar los casos de prueba 5. Registrar y reportar los resultados 6. Repetir las pruebas 7. Verificar la integridad del software |
| No funcionales | En cualquier nivel del proceso | Que el software funcione bien, así como su fiabilidad y funcionamiento. | 1. Identificar los requisitos no funcionales 2. Diseñar los casos de prueba 3. Preparar el entorno de prueba 4. Ejecutar los casos de prueba 5. Registrar y reportar los resultados 6. Repetir las pruebas   Verificar la integridad del software |
| Caja negra | En cualquier momento del ciclo de vida del software | Funcionalidad del software sin examinar la estructura del código interno. | 1. Reconocer los requerimientos y especificaciones del software. 2. Escoger valores de entrada que sean válidos e inválidos para constatar cómo son procesados por el software. 3. Determinar cuáles son las respuestas esperadas para cada uno de los valores elegidos. 4. Construye casos de prueba para los valores de entrada y ejecútalos, compara las respuestas obtenidas con las esperadas y determina si hay errores. |
| Caja blanca | En la fase de desarrollo del software | 1. Fallas en la seguridad interna. 2. Trayectorias mal estructuradas o rotas en los procesos de codificación. 3. El flujo de los valores de entrada a través del código y los resultados esperados. 4. La funcionalidad de los bucles condicionales. | 1. Identificar los componentes a probar 2. Diseñar los casos de prueba 3. Preparar el entorno de prueba 4. Ejecutar los casos de prueba 5. Registrar y reportar los resultados 6. Verificar la integridad del software |

VI. Relaciona las columnas para identificar los tipos de métricas.

Utilizan el número de defectos Métricas de seguridad  
encontrados en el producto.

Sirven para determinar la longitud del   
software, se hace contando las líneas Métricas de tamaño  
de código que lo forman.

Se usan para determinar si el sistema  
podrá resistir “ataques de acceso no Métricas de calidad   
autorizado”.