

# **TALLER SOBRE CODIFICACION DE MODULOS DEL SOFTWARE**

**APRENDIZ**

**BRAYAN STIVEN PEÑA QUINAYAS**

**ADSO**

**SENA**

**2024**

## **INTRODUCCION**

Una solución de software es un conjunto de elementos diseñados para el tratamiento y administración eficiente de datos e información. En este taller, exploraremos los diferentes tipos de pruebas de software, su importancia y como aplicarlas en el contexto del desarrollo de un proyecto de software.

**¿Qué tipos de pruebas de software existen? Explique sus características y beneficios.**

## **TIPOS DE PRUEBAS DE SOFTWARE**

### **1. PRUEBAS DE UNIDAD.**

#### **Características:**

- Se centran en verificar el funcionamiento individual de unidades o componentes de código, como funciones, métodos o clases.
- Suelen realizarse a nivel de código y se ejecutan de manera aislada, sin depender de otros módulos o componentes del sistema.
- Utilizan técnicas como la caja blanca (pruebas estructurales) o la caja negra (pruebas funcionales).

#### **Beneficios:**

- Identifican errores en etapas tempranas del desarrollo, lo que facilita su corrección y reduce el costo asociado.
  - Mejoran la calidad del código al garantizar que cada unidad funcione correctamente según los requisitos especificados.
- Facilitan la reutilización del código al proporcionar una base sólida y probada para futuras implementaciones.

### **2. PRUEBAS DE INTEGRACION**

#### **Características:**

- Verifican la correcta interacción entre diferentes módulos, componentes o subsistemas del sistema.
- Se realizan después de las pruebas unitarias y antes de las pruebas de sistema para asegurar que los componentes se integren sin problemas.
- Pueden ser de dos tipos, ascendentes (de abajo hacia arriba) o descendentes (de arriba hacia abajo).

#### **Beneficios:**

- Detectan problemas de interfaz comunicación entre los módulos antes de la fase de pruebas de sistema, lo que ayuda a evitar errores costosos.
- Aseguran que las diferentes partes del sistema funcionen juntas como se espera, garantizando la coherencia y la integridad del sistema completo.
- Permiten identificar y resolver conflictos entre módulos antes de que afecten al funcionamiento del sistema en su conjunto.

### **3. PRUBAS DE SISTEMA**

#### **Características:**

- Evalúan el comportamiento del sistema completo según los requisitos funcionales y no funcionales específicos.
  - Se realizan después de las pruebas de integración y antes de las pruebas de aceptación para validar el sistema en su totalidad.
- Incluyen pruebas de rendimiento, seguridad, usabilidad y compatibilidad entre otras.

#### **Beneficios:**

- Verifican si el sistema cumple con los objetivos y las expectativas del usuario antes de su implementación.
- Identifican problemas de rendimiento, seguridad o usabilidad que podrían afectar la experiencia de usuario final.
- Proporcionan una visión global de la calidad del sistema y permiten tomar decisiones informadas sobre su lanzamiento.

#### **4. PRUEBAS DE ACEPTACION**

##### **Características:**

- Validan si el sistema cumple con los criterios de aceptación definidos por el cliente o los stakeholders.
- Se realizan después de las pruebas de sistema y suelen ser realizados por usuarios finales o representantes del cliente.
- Pueden incluir pruebas funcionales, de usabilidad de accesibilidad, entre otras.

##### **Beneficios:**

- Garantizan la satisfacción del cliente al verificar que el sistema cumpla con sus expectativas y requisitos específicos.
- Proporcionan retroalimentación directa del usuario final, lo que ayuda a identificar áreas de mejora y realizar ajustes finales antes del lanzamiento.
  - Establecen un criterio para determinar si el sistema está listo para ser implementado y utilizado en producción.

**Según la consulta que realizó, ¿qué tipos de pruebas se adaptan mejor al proyecto de software que está desarrollando?**

Para mi proyecto que es de una tienda online de videojuegos, es crucial garantizar que el software funcione correctamente, sea fácil de usar y proporcione una experiencia satisfactoria para los usuarios. Basándonos en esta premisa, los tipos de pruebas que se adaptan mejor al proyecto son:

##### **1. Pruebas de unidad:**

- En un proyecto de tienda online de videojuegos en línea, las pruebas de unidad son fundamentales para verificar el correcto funcionamiento de cada componente del sistema, como la gestión de inventario, el procesamiento de pagos, la visualización de productos, etc.
- Las pruebas de unidad ayudaran a identificar y corregir errores en el código a nivel de funciones o módulos específicos, lo que garantiza la estabilidad y fiabilidad del sistema en su conjunto.

## 2. Pruebas de integración:

- Dado que una tienda de videojuegos en línea involucra múltiples sistemas y componentes que deben interactuar entre sí, las pruebas de integración son esenciales para asegurar que todas las partes del sistema funcionen correctamente juntas.
- Las pruebas de integración permitirán detectar y resolver problemas de comunicación entre diferentes módulos, como la integración entre el catálogo de productos y el proceso de pago, la sincronización de inventario con el sistema de gestión de almacén, etc.

**Investigue e instale unas herramientas de pruebas de software en su computador, “una de su gusto”.**

### **Investigación de herramientas:**

En mi caso investigue acerca de algunas herramientas de pruebas de software, las mas comunes por así decirlo e instale Postman.

- **Selenium:** Para pruebas de software automatizadas.
  - **JUnit:** Para pruebas unitarias en Java.
  - **Postman:** Para pruebas de API.
- **Cypress:** Para pruebas de extremo a extremo en aplicaciones web.
- **Appium:** Para pruebas automatizadas en aplicaciones móviles.

# POSTMAN:

