



**Universidad Tecnológica
de Chihuahua**

TECNOLOGÍAS DE LA INFORMACIÓN: DESARROLLO DE SOFTWARE
MULTIPLATAFORMA

EVALUACIÓN Y MEJORA PARA EL DESARROLLO DE SOFTWARE

Reporte de investigación de tipos de pruebas de software

Docente: Dynhora Ramírez

Oscar Ariel Quintana Merino
TID41M

Tabla de contenido

Tipos de Prueba de Software	2
¿Qué es una prueba de software?	2
Objetivo del plan de pruebas.	2
Diferencias entre la pruebas manuales y automatizadas.	2
Pruebas manuales:	2
Pruebas automatizadas:	3
Tipos de prueba de software	4
Importancia del plan de pruebas.	8
Tipos de Casos de Prueba:	9
Escenarios de Prueba:	9
FUENTES DE CONSULTA	11

Tipos de Prueba de Software

¿Qué es una prueba de software?

Una prueba de software es un proceso sistemático y controlado que se realiza para evaluar y verificar la calidad y el funcionamiento de un programa o sistema de software. Estas pruebas son fundamentales para garantizar que el software cumpla con los requisitos y expectativas del cliente, sea confiable, seguro y funcione correctamente.

Objetivo del plan de pruebas.

El objetivo principal de las pruebas de software es identificar defectos, errores o problemas en el software antes de su lanzamiento o implementación en un entorno de producción. Esto implica asegurarse de que el software funcione correctamente, sea seguro y cumpla con los estándares de calidad. Además, busca evitar problemas costosos y disruptivos que puedan surgir una vez que el software esté en manos de los usuarios finales.

Diferencias entre las pruebas manuales y automatizadas.

La principal diferencia entre las pruebas manuales y automatizadas radica en la ejecución: las pruebas manuales son realizadas por seres humanos que interactúan directamente con el software, mientras que las pruebas automatizadas son realizadas por herramientas y scripts de software. Cada enfoque tiene sus ventajas y desventajas, y la elección entre ambos depende de factores como el tipo de prueba, el presupuesto, el tiempo disponible y la naturaleza del proyecto de desarrollo de software.

Algunas características son las siguientes:

Pruebas manuales:

Las pruebas manuales son llevadas a cabo por seres humanos, generalmente testers o ingenieros de pruebas, que interactúan directamente con el software para evaluar su funcionamiento. Estas pruebas involucran acciones como hacer clic en

botones, ingresar datos, navegar por menús y realizar diversas operaciones en la aplicación, emulando el comportamiento de un usuario real. La ventaja de las pruebas manuales radica en que permiten la exploración y la creatividad por parte de los testers, lo que les permite adaptarse y descubrir problemas no previstos. Son especialmente adecuadas para pruebas iniciales, exploratorias y de usabilidad. Sin embargo, son intensivas en tiempo y esfuerzo, lo que puede ser un desafío en proyectos grandes o complejos.

- Ejecución por humanos
- Interacción humana
- Exploración y creatividad
- Pruebas iniciales y exploratorias
- Tiempo y esfuerzo intensivos

(De Telecomunicación et al., 2015)

Pruebas automatizadas:

Las pruebas automatizadas, por otro lado, se realizan utilizando herramientas de software y scripts que ejecutan automáticamente las acciones de prueba y evalúan los resultados. Estas pruebas son altamente repetibles y consistentes, ya que las mismas acciones se ejecutan de la misma manera cada vez que se realizan las pruebas. Son ideales para pruebas de regresión, donde se verifica que las nuevas actualizaciones de software no introduzcan errores en funcionalidades existentes, y también son eficaces para pruebas repetitivas, como pruebas de carga y rendimiento. Sin embargo, las pruebas automatizadas son menos adecuadas para evaluar aspectos subjetivos de la usabilidad y la experiencia del usuario, ya que carecen de la intuición y la creatividad humanas. Aunque requieren tiempo y esfuerzo inicial para desarrollar scripts de prueba, pueden ahorrar tiempo significativo a lo largo del ciclo de vida del proyecto, especialmente cuando se repiten las pruebas con regularidad.

- Ejecución por herramientas y scripts
- Repetibilidad y consistencia
- Ideal para regresión y casos de uso repetitivos

- Menos adecuadas para pruebas de usabilidad
- Ahorro de tiempo a largo plazo

(De Telecomunicación et al., 2015)

Tipos de prueba de software:

- **Prueba de Unidad (Unit Testing):**

- **Descripción:** Esta prueba se centra en evaluar componentes individuales o unidades de código, como funciones o métodos, para asegurarse de que funcionen correctamente de manera aislada.
- **Pasos o Estructura:** El tester o el desarrollador crea casos de prueba para cada unidad y ejecuta las pruebas con datos de entrada específicos. Se comparan los resultados con los resultados esperados.
- **Aplicación Principal:** La prueba de unidad se utiliza principalmente para verificar que las partes más pequeñas del software estén libres de errores y cumplan con sus especificaciones.
- **Naturaleza:** Prueba automatizada.

(Types of Software Testing: Different Testing Types with Details, 2023)

- **Prueba de Integración (Integration Testing):**

- **Descripción:** Esta prueba evalúa cómo se combinan y funcionan conjuntamente varios módulos o componentes del software cuando se integran en un sistema más grande.
- **Pasos o Estructura:** Se identifican las interfaces entre los módulos y se crean casos de prueba para probar la comunicación y la interacción entre ellos. Se busca asegurar que los módulos funcionen correctamente juntos.
- **Aplicación Principal:** La prueba de integración se utiliza para garantizar que los componentes individuales se integren sin problemas y cumplan con los requisitos de diseño.

- **Naturaleza:** Puede ser manual o automatizada.

(Types of Software Testing: Different Testing Types with Details, 2023)

- **Prueba funcional (Functional Testing):**

- **Descripción:** Prueba las funciones y características del software para asegurarse de que funcionen según lo previsto.
- **Pasos o Estructura:** Su aplicación es validar que el software cumpla con los requisitos funcionales.
- **Aplicación Principal:** Prueba las funciones y características del software para asegurarse de que funcionen según lo previsto.
- **Naturaleza:** Puede ser tanto manual como automatizado, dependiendo de los casos de prueba.

(Types of Software Testing: Different Testing Types with Details, 2023)

- **Prueba no Funcional (Non-Functional Testing):**

- **Descripción:** Evalúa atributos no funcionales del software, como rendimiento, seguridad y usabilidad.
- **Pasos o Estructura:** Su aplicación es garantizar que el software cumpla con aspectos no funcionales importantes.
- **Aplicación Principal:** Evalúa atributos no funcionales del software, como rendimiento, seguridad y usabilidad.
- **Naturaleza:** Puede ser tanto manual como automatizado, según los atributos que se evalúen.

(Types of Software Testing: Different Testing Types with Details, 2023)

- **Pruebas de Cumplimiento (Compliance Testing):**

- **Descripción:** Esta prueba verifica si el software cumple con estándares y regulaciones específicas, como regulaciones gubernamentales o industriales.
- **Pasos o Estructura:** Su aplicación es garantizar que el software cumpla con requisitos legales y normativos.

- **Aplicación Principal:** Verifica si el software cumple con estándares y regulaciones específicas, como regulaciones gubernamentales o industriales.
- **Naturaleza:** Puede ser manual o automatizado, dependiendo de los requisitos y regulaciones específicas.

(Types of Software Testing: Different Testing Types with Details, 2023)

- **Prueba de Regresión (Regression Testing):**

- **Descripción:** Esta prueba se realiza para verificar que las nuevas actualizaciones o cambios en el software no afecten negativamente las funcionalidades existentes.
- **Pasos o Estructura:** Se ejecutan casos de prueba previamente establecidos que cubren las áreas del software afectadas por los cambios. Se comparan los resultados con los resultados previos.
- **Aplicación Principal:** La prueba de regresión es fundamental para asegurarse de que las nuevas versiones no introduzcan errores en funcionalidades existentes.
- **Naturaleza:** Puede ser manual o automatizada.

(Types of Software Testing: Different Testing Types with Details, 2023)

- **Prueba de Usabilidad (Usability Testing):**

- **Descripción:** Esta prueba evalúa la facilidad de uso y la experiencia del usuario en el software, incluyendo la navegación, el diseño y la eficiencia.
- **Pasos o Estructura:** Los usuarios reales o simulados realizan tareas específicas en el software mientras se observa su interacción y se recopilan comentarios y métricas de usabilidad.
- **Aplicación Principal:** La prueba de usabilidad se utiliza para garantizar que el software sea intuitivo y satisfactorio para los usuarios finales.
- **Naturaleza:** Es principalmente una prueba manual.

(Types Archives - Software Testing Fundamentals, s/f)

- **Prueba de Carga (Load Testing):**

- **Descripción:** Esta prueba evalúa el rendimiento y la capacidad del software al someterlo a cargas de trabajo y volúmenes de datos significativos.
- **Pasos o Estructura:** Se diseñan casos de prueba que simulan la carga esperada en el sistema y se mide cómo responde el software en términos de velocidad y capacidad de respuesta.
- **Aplicación Principal:** La prueba de carga se utiliza para identificar cuellos de botella y problemas de rendimiento bajo condiciones de alta carga.
- **Naturaleza:** Puede ser automatizada.

(Types Archives - Software Testing Fundamentals, s/f)

- **Pruebas de Humo (Smoke Testing):**

- **Descripción:** Una prueba rápida y superficial para verificar que las funciones básicas del software funcionen después de un cambio o una nueva construcción.
- **Pasos o Estructura:** Identificar problemas críticos antes de realizar pruebas más exhaustivas.
- **Aplicación Principal:** Una prueba rápida y superficial para verificar que las funciones básicas del software funcionen después de un cambio o una nueva construcción.
- **Naturaleza:** Puede ser automatizado para ejecutar pruebas básicas.

(Types Archives - Software Testing Fundamentals, s/f)

- **Pruebas de Seguridad (Security Testing):**

- **Descripción:** Evalúa la seguridad del software, identificando vulnerabilidades y puntos débiles que podrían ser explotados por atacantes.

- **Pasos o Estructura:** Proteger el software contra posibles amenazas de seguridad.
- **Aplicación Principal:** Evalúa la seguridad del software, identificando vulnerabilidades y puntos débiles que podrían ser explotados por atacantes.
- **Naturaleza:** Puede ser manual o automatizado, con herramientas de seguridad especializadas.

(Types Archives - Software Testing Fundamentals, s/f)

Importancia del plan de pruebas.

El plan de pruebas es esencial en el desarrollo de software. Ayuda a identificar errores tempranamente, asegura que el software cumpla con los requisitos y mejora la comunicación entre equipos. Además, proporciona una metodología para pruebas repetibles y datos objetivos para tomar decisiones informadas. En última instancia, garantiza la calidad del producto y satisface a los clientes, ahorrando tiempo y recursos.

Algunas características son las siguientes:

- Garantiza la calidad del software
- Ahorra tiempo y recursos
- Mejora la comunicación
- Facilita la trazabilidad
- Permite la repetibilidad
- Facilita la toma de decisiones
- Asegura la satisfacción del cliente
- Reduce riesgos

Tipos de Casos de Prueba:

- **Casos de Prueba Funcionales:** Estos casos evalúan si el software cumple con los requisitos funcionales especificados. Se centran en las acciones que un usuario debe realizar y en las salidas esperadas. Por ejemplo, un caso de prueba funcional para un sistema de registro de usuarios puede verificar si un nuevo usuario puede registrarse correctamente y acceder a su cuenta.
- **Casos de Prueba No Funcionales:** Estos casos evalúan aspectos no funcionales del software, como rendimiento, seguridad, usabilidad y escalabilidad. Por ejemplo, un caso de prueba no funcional podría medir el tiempo de respuesta del sistema bajo una carga específica de usuarios.
- **Casos de Prueba de Regresión:** Estos casos se utilizan para verificar que las actualizaciones o cambios en el software no afecten negativamente las funcionalidades existentes. Se ejecutan repetidamente para garantizar que las características previamente probadas sigan funcionando correctamente después de las modificaciones.

(What is Software Testing and How Does it Work? | IBM, s/f)

Escenarios de Prueba:

- **Escenarios de Prueba de Flujo de Trabajo:** Estos escenarios representan una serie de pasos interconectados que un usuario realiza para completar una tarea o un flujo de trabajo específico en el software. Por ejemplo, un escenario de prueba de flujo de trabajo en una aplicación de compra en línea podría incluir la búsqueda de un producto, la selección de artículos y el proceso de pago.
- **Escenarios de Prueba de Estrés:** Estos escenarios se diseñan para evaluar la capacidad del software para manejar cargas extremas o situaciones límite. Por ejemplo, un escenario de prueba de estrés podría simular miles de

usuarios concurrentes en un sitio web para medir su rendimiento bajo una carga pesada.

- Escenarios de Prueba de Seguridad: Estos escenarios se enfocan en identificar vulnerabilidades y evaluar la resistencia del software a ataques maliciosos. Por ejemplo, un escenario de prueba de seguridad podría intentar vulnerar la autenticación de usuarios o verificar la seguridad de los datos almacenados.
- Escenarios de Prueba de Usabilidad: Estos escenarios evalúan la experiencia del usuario y la facilidad de uso del software. Se centran en la interacción del usuario con la interfaz y pueden incluir tareas como la navegación por menús, la búsqueda de información y la retroalimentación del usuario.

(What is Software Testing and How Does it Work? | IBM, s/f)

FUENTES DE CONSULTA

- PITTET, S. (n.d.). Los distintos tipos de pruebas en software | Atlassian. Retrieved September 10, 2023, from <https://www.atlassian.com/es/continuous-delivery/software-testing/types-of-software-testing>
- de Telecomunicación, S., Politécnica, U., Madrid, D. E., Manuel, J., & Peño, S. (2015). ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA Y Pruebas de Software. Fundamentos y Técnicas.
- Types of Software Testing: Different Testing Types with Details. (2023). <https://www.softwaretestinghelp.com/types-of-software-testing/>
- Types Archives - Software Testing Fundamentals. (n.d.). Retrieved September 10, 2023, from <https://softwaretestingfundamentals.com/category/types/>
- What is Software Testing and How Does it Work? | IBM. (n.d.). Retrieved September 10, 2023, from <https://www.ibm.com/topics/software-testing>