**Nombre:** José Manuel Vargas Poma

**Investigar 8 tipos de pruebas que existen en la Ingeniera de Software**

1. **Pruebas funcionales**

Se llevan a cabo para comprobar las características críticas para el negocio, la funcionalidad y la usabilidad. Las pruebas funcionales garantizan que las características y funcionalidades del software se comportan según lo esperado sin ningún problema.

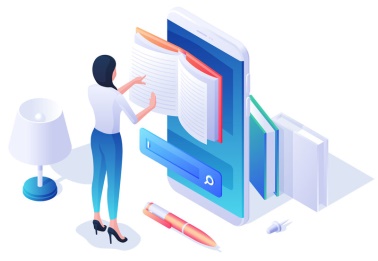


Ventajas

* Se asegura de que el sitio web, aplicación está libre de defectos.
* Garantiza el comportamiento esperado de todas las funcionalidades.
* Garantiza que la arquitectura sea correcta con la seguridad necesaria.
* Mejora la calidad y las funcionalidades generales.
* Minimiza los riesgos empresariales asociados con el sitio web/aplicación.

1. **Pruebas unitarias**

Las pruebas unitarias se centran en probar piezas/unidades individuales de una aplicación de software al principio del SDLC. Cualquier función, procedimiento, método o módulo puede ser una unidad que se someta a pruebas unitarias para determinar su corrección y comportamiento esperado.



Ventajas

* Detección temprana de errores en las nuevas funcionalidades o características desarrolladas.
* Minimiza los costos de las pruebas a medida que se detectan problemas desde el principio.
* Mejora la calidad del código con una mejor refactorización del código.
* Simplifica la integración y permite una buena documentación.

1. **Pruebas de integración**

Las pruebas de integración implican probar diferentes módulos de una aplicación de software como grupo. Una aplicación de software se compone de diferentes submódulos que trabajan juntos para diferentes funcionalidades. El propósito de las pruebas de integración es validar la integración de diferentes módulos juntos e identificar los errores y problemas relacionados con ellos.

Ventajas

* Se asegura de que todos los módulos de aplicación estén bien integrados y funcionen juntos según lo esperado.
* Detecta problemas y conflictos interconectados para resolverlos antes de crear un gran problema.
* Valida la funcionalidad, fiabilidad y estabilidad entre diferentes módulos.
* Detecta excepciones ignoradas para mejorar la calidad del código.

1. **Pruebas no funcionales**

Las pruebas no funcionales son como pruebas funcionales; sin embargo, la principal diferencia es que esas funciones se prueban bajo carga para el rendimiento de los observadores, fiabilidad, usabilidad, escalabilidad, etc. Las pruebas no funcionales, como las pruebas de carga y esfuerzo, normalmente se llevan a cabo mediante herramientas y soluciones de automatización, como [LoadView](https://www.loadview-testing.com/es/). Además de las pruebas de rendimiento, los tipos de pruebas no funcionales incluyen pruebas de instalación, pruebas de confiabilidad y pruebas de seguridad.



1. **Prueba de Performance Testing**

Las pruebas de rendimiento son un tipo de pruebas no funcionales, realizadas para determinar la velocidad, estabilidad y escalabilidad de una aplicación de software. Como su nombre indica, el objetivo general de esta prueba es comprobar el rendimiento de una aplicación con respecto a los diferentes puntos de referencia del sistema y de la red, como la utilización de la CPU, la velocidad de carga de la página, el control de tráfico máximo, la utilización de recursos del servidor, etc.

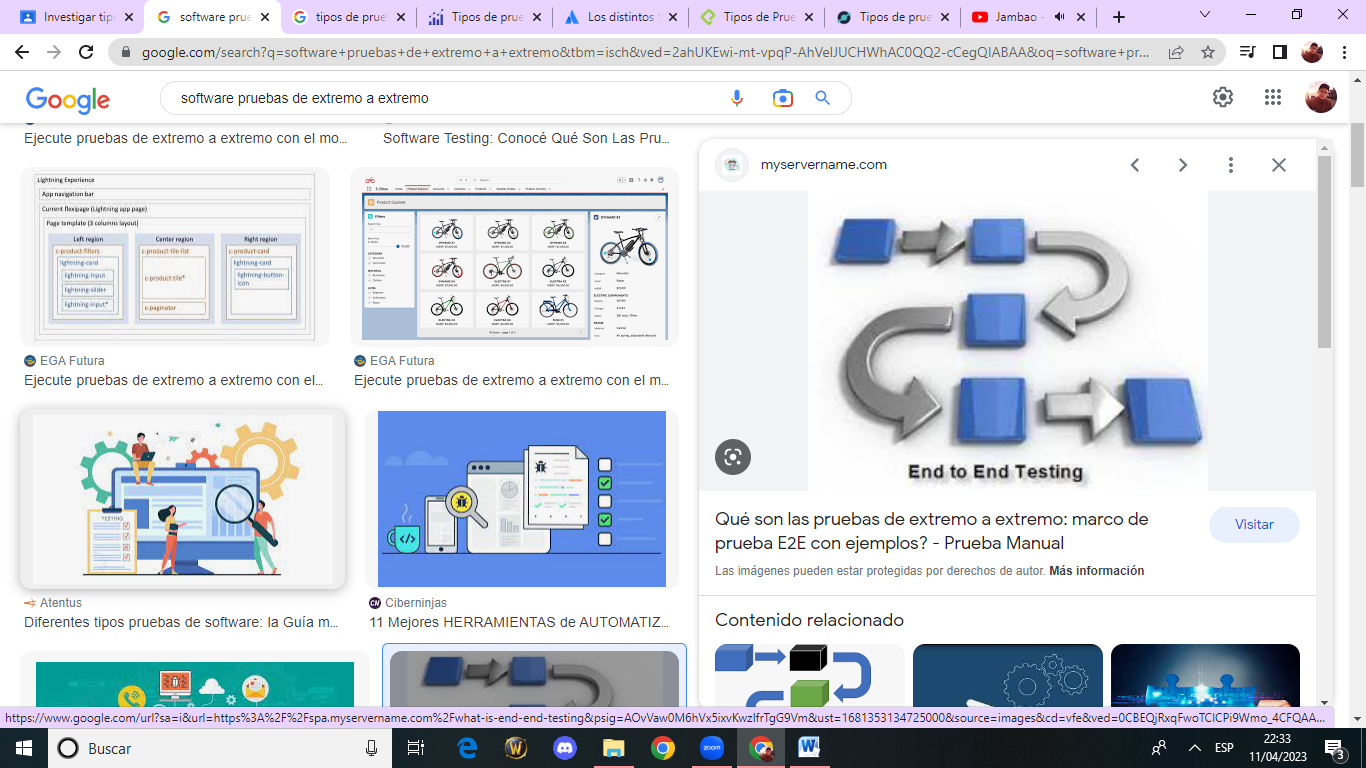


Ventajas

* Evalúa la velocidad y escalabilidad del sitio web/aplicación.
* Identifica los cuellos de botella para las mejoras de rendimiento.
* Detecta errores que se pasan por alto en las pruebas funcionales.
* Optimización del sistema y mejoras de características
* Garantiza la fiabilidad del sitio web bajo una gran carga.

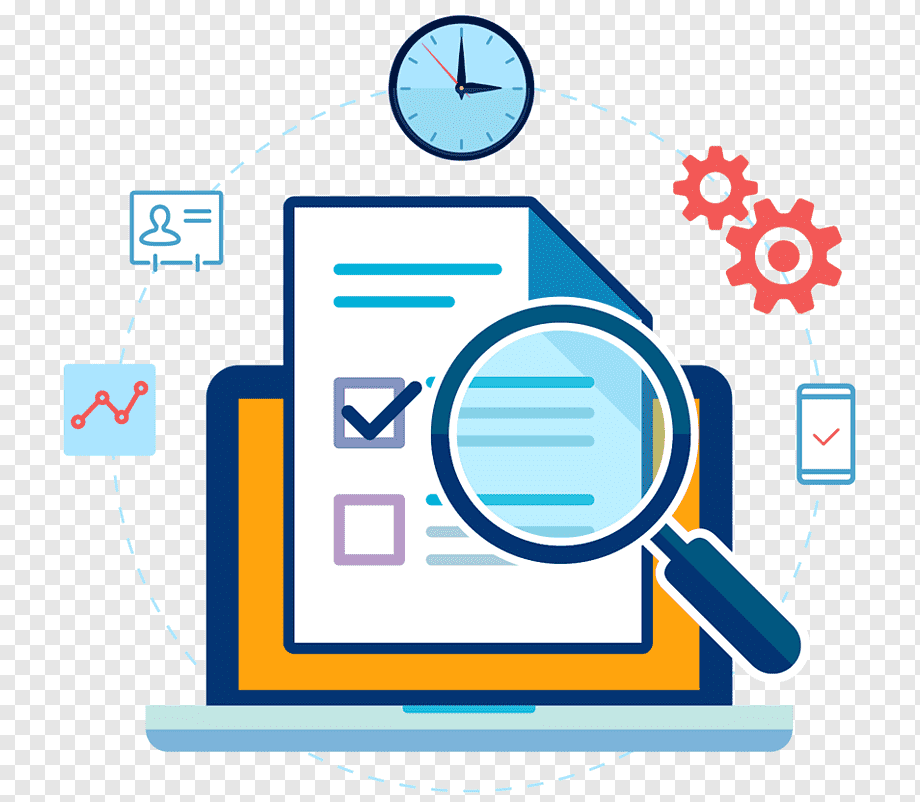
1. **Pruebas de extremo a extremo**

Las pruebas integrales replican el comportamiento de un usuario con el software en un entorno de aplicación completo. Además, verifican que diversos flujos de usuario funcionen según lo previsto, y pueden ser tan sencillos como cargar una página web o iniciar sesión, o mucho más complejos, como la verificación de notificaciones de correo electrónico, pagos en línea, etc.



1. **Pruebas de rendimiento**

Las pruebas de rendimiento evalúan el rendimiento de un sistema con una carga de trabajo determinada. Ayudan a medir la fiabilidad, la velocidad, la escalabilidad y la capacidad de respuesta de una aplicación. Por ejemplo, una prueba de rendimiento puede analizar los tiempos de respuesta al ejecutar un gran número de solicitudes, o cómo se comporta el sistema con una cantidad significativa de datos. Puede determinar si una aplicación cumple con los requisitos de rendimiento, localizar cuellos de botella, medir la estabilidad durante los picos de tráfico y mucho más.



1. **Pruebas de humo**

Las pruebas de humo son pruebas básicas que sirven para comprobar el funcionamiento básico de la aplicación. Están concebidas para ejecutarse rápidamente, y su objetivo es ofrecerte la seguridad de que las principales funciones de tu sistema funcionan según lo previsto.

Las pruebas de humo pueden resultar útiles justo después de realizar una compilación nueva para decidir si se pueden ejecutar o no pruebas más caras, o inmediatamente después de una implementación para asegurarse de que la aplicación funciona correctamente en el entorno que se acaba de implementar.

