

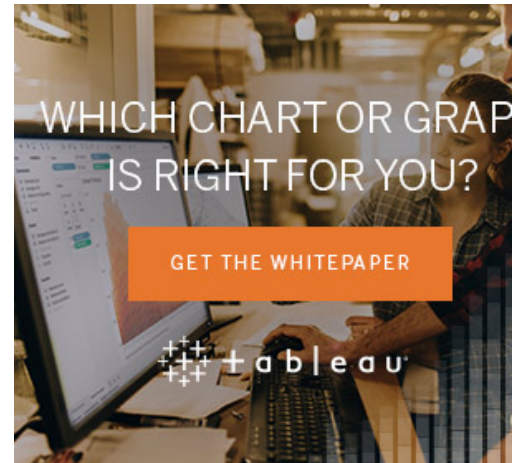


« [Tutorial 5: ¿Por qué TestComplete es la herramienta más adecuada para automatizar el sistema operativo Android?](#)

## Prueba disruptiva

**Disruptive testing** is a type of testing which is carried out to make a software application crash or fail in order to determine the behavior of the application when different amounts of load are applied to it. It is the best suitable for the products which are produced in large quantities since the loss for destroying a small number of quantities would be less than the cost of destroying the larger one. For example, mobile testing. When a new version of a mobile software is developed, it will be used in a large number of handsets and the risk of not performing load or stress test can affect the business extensively.

Las pruebas disruptivas son comparativamente más fáciles de realizar y brindan más información que otros tipos de pruebas no funcionales. Determina la solidez del software. Comprendamos este tipo de pruebas en detalle comenzando con los tipos de pruebas disruptivas.



### Tipos de pruebas disruptivas:

*Las pruebas disruptivas* son de los siguientes tipos:



### Pruebas destructivas:

Es la forma tradicional de probar la solidez de la aplicación de software. Como se discutió anteriormente, esta prueba se realiza para determinar el comportamiento de la aplicación cuando se le aplican diferentes cantidades de carga. **Las pruebas destructivas** también se utilizan para verificar que el software funcione correctamente incluso cuando acepta entradas inválidas o inesperadas, lo que ayuda a establecer la solidez de la validación de entrada y las rutinas de gestión de errores.

#### Prueba de manejo de excepciones:

El manejo de excepciones forma parte de las pruebas destructivas cuando el sistema falla debe existir un determinado proceso para respaldar la ocurrencia, durante el cálculo de excepciones, de condiciones irregulares o excepcionales que requieren un procesamiento especial que altera el flujo normal de ejecución del programa. Tal manejo excepcional es proporcionado por las construcciones del lenguaje de programación o los mecanismos de hardware de la computadora.

#### Pruebas de recuperación:

La prueba de recuperación es el tipo de prueba para probar el comportamiento de la aplicación de software para recuperarse de fallas o fallos de hardware o software. Las pruebas de recuperación se facilitan mediante la falla forzada del software de numerosas maneras para verificar que la recuperación se realiza correctamente o no. Es diferente de las pruebas de confiabilidad, donde solo nos enfocamos en probar el punto de falla. Las pruebas de recuperación son una prueba básica para verificar qué tan rápido puede recuperarse una aplicación de software contra varios tipos de falla, fallas de hardware u otros problemas catastróficos. etc.

### Ventajas de realizar pruebas disruptivas:

Ayuda a determinar el comportamiento de la aplicación de software cuando el software está sujeto a un uso inadecuado.

Ayuda a verificar la solidez del producto de software.



#### CUBA Platform - Open Source Java Web Framework

Ad Visual design, code generation, enterprise components out of the box  
cuba-platform.com

[Learn more](#)

### Desventajas de realizar pruebas disruptivas:

Un costo adicional para la organización.

### Estrategia de prueba destructiva:

Como cualquier prueba tradicional, la prueba destructiva tiene casos de prueba que diseñan, prueban el desarrollo de scripts, la ejecución de scripts de prueba, el aumento de errores, el cierre de bugs y el informe de prueba a los interesados después de la finalización de las pruebas. Las siguientes son algunas de las formas de realizar las pruebas destructivas.

**Métodos para analizar el punto de falla:** en esta estrategia, el analista de negocios podría jugar un papel clave en el recorrido del sistema para realizar una evaluación de lo que podría salir mal en varios puntos.

**Revisión por pares:** los casos de prueba deben ser revisados por un compañero de pruebas o un compañero que no esté muy familiarizado con las operaciones del sistema.

**Revisión de casos de prueba:** los usuarios finales o los expertos que conocen varios escenarios de aplicaciones deben hacerlo. Estos escenarios podrían haber pasado desapercibidos para el probador al diseñar los casos de prueba, pero los expertos pueden ayudar a incluir esos escenarios de prueba faltantes.

**Realice pruebas exploratorias:** las pruebas exploratorias, como de costumbre, pueden ayudar a determinar la robustez de la aplicación al realizar pruebas directamente en las áreas débiles de la aplicación.

**Prueba de mono:** tiene como objetivo romper la aplicación a cualquier costo. El experto puede atrapar directamente la vena yugular del software y presionar allí para interrumpir la aplicación.

## ¿Cómo realizar pruebas disruptivas?

En pruebas destructivas, verificamos las siguientes cosas:

**Comportamiento de aplicación de software adecuado:** es la verificación del comportamiento del software durante los diversos volúmenes de solicitud que comprende desde un volumen de solicitud bajo hasta el volumen de solicitud de picos. El objetivo es bloquear la aplicación y estudiar el comportamiento de la aplicación bajo prueba.

**Comportamiento incorrecto de la aplicación de software:** la prueba de confiabilidad se realiza para marcar el punto donde la aplicación puede entregar el comportamiento adecuado antes de que se descomponga. Pero, al mismo tiempo, es esencial conocer el comportamiento de la aplicación inadecuado cuando realizamos pruebas destructivas más allá del punto de corte. Estas especificaciones podrían ser útiles para el equipo de soporte de aplicaciones para el buen funcionamiento de la aplicación en el modo de producción.

**Uso incorrecto :** puede haber escenarios de uso incorrecto de la aplicación de software, como volumen máximo frecuente o volumen inesperado. Bajo tales escenarios, el comportamiento exacto de la aplicación de software debe conocerse de antemano, lo que solo puede ser posible a través de pruebas destructivas.

**Datos de entrada incorrectos:** no siempre ocurre que los datos de entrada válidos se envíen a la aplicación. Puede haber escenarios en los que se envíen datos no válidos a la aplicación que pueden provocar el bloqueo de la aplicación. Una vez más, dicho comportamiento y la forma de la aplicación podrían probarse a través de las pruebas destructivas.

**Datos de salida correctos :** todas las aplicaciones informáticas siguen la regla predeterminada de GIGO, es decir, Basura adentro y Basura fuera. Las pruebas destructivas se pueden utilizar para estudiar el comportamiento de la aplicación frente a la entrada de basura frente a la salida de basura, lo que explica el nivel de validaciones realizadas dentro de la aplicación para superar la situación de fallas o fallos del software.

## Conclusión:

En la técnica de prueba destructiva, una aplicación de software se hace a propósito para bloquear o fallar el programa con el fin de verificar la robustez. Además, para llevar a cabo las pruebas destructivas no es crucial tener el conocimiento de la aplicación de los requisitos originales.

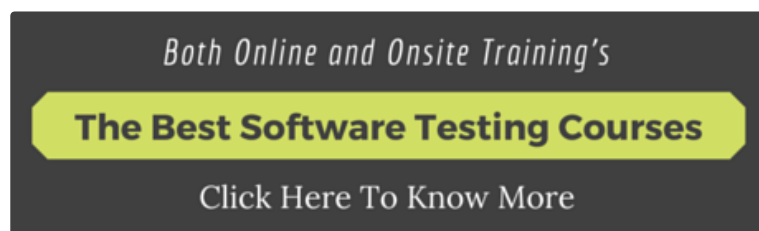
### Compartir este:



### Artículos Relacionados:

1. [¿Qué es la prueba de resistencia en las pruebas de software?](#)
2. [Prueba de recuperación](#)
3. [Conceptos básicos de las pruebas del sistema: una guía para principiantes](#)
4. [¿Qué es una prueba negativa y consejos para escribir casos de prueba negativos?](#)
5. [¿Qué es la prueba de portabilidad en el software?](#)
6. [Prueba de usabilidad: ¿Qué? ¿Por qué? & ¿Cómo?](#)
7. [¿Qué es Pruebas Exhaustivas en Pruebas de Software?](#)
8. [Cobertura de prueba en pruebas de software](#)
9. [Pruebas de automatización frente a pruebas manuales](#)
10. [¿Qué es una prueba estructural?](#)

Los mejores cursos de capacitación en pruebas de software



22 de enero de 2018 | Etiquetas: [Ventajas](#) , [Pruebas destructivas](#) , [desventajas](#) , [Estrategia](#) , [tipo de prueba](#) , [tipos de pruebas](#) | Categoría: [tipos de pruebas](#)

3