

Simulación y análisis de condiciones de carga extrema

Calidad de Software - CSY4111



CONTENIDO

01
INTRODUCCIÓN A LAS
PRUEBAS DE RENDIMIENTO

02
JUSTIFICACIÓN Y
BENEFICIOS

02
HERRAMIENTAS

Refrescando conocimiento

DuocUC[®]



REFRESCANDO CONOCIMIENTO

Las pruebas de inspección de código son un proceso de revisión sistemática y exhaustiva del código fuente de un programa para identificar errores, vulnerabilidades, malas prácticas y oportunidades de mejora.

Estas pruebas se basan en el análisis estático del código, lo que significa que se examina el código sin ejecutarlo.

El objetivo principal de las pruebas de inspección de código es garantizar la calidad y la robustez del software.

REFRESCANDO CONOCIMIENTO

Mejora de la calidad del software:

Al detectar y corregir errores y problemas en el código, las pruebas de inspección de código contribuyen a un software más confiable, estable y de alta calidad.

Ahorro de costos:

Identificar y solucionar problemas en etapas tempranas del desarrollo evita que se propaguen y se conviertan en errores costosos de corregir en etapas posteriores.





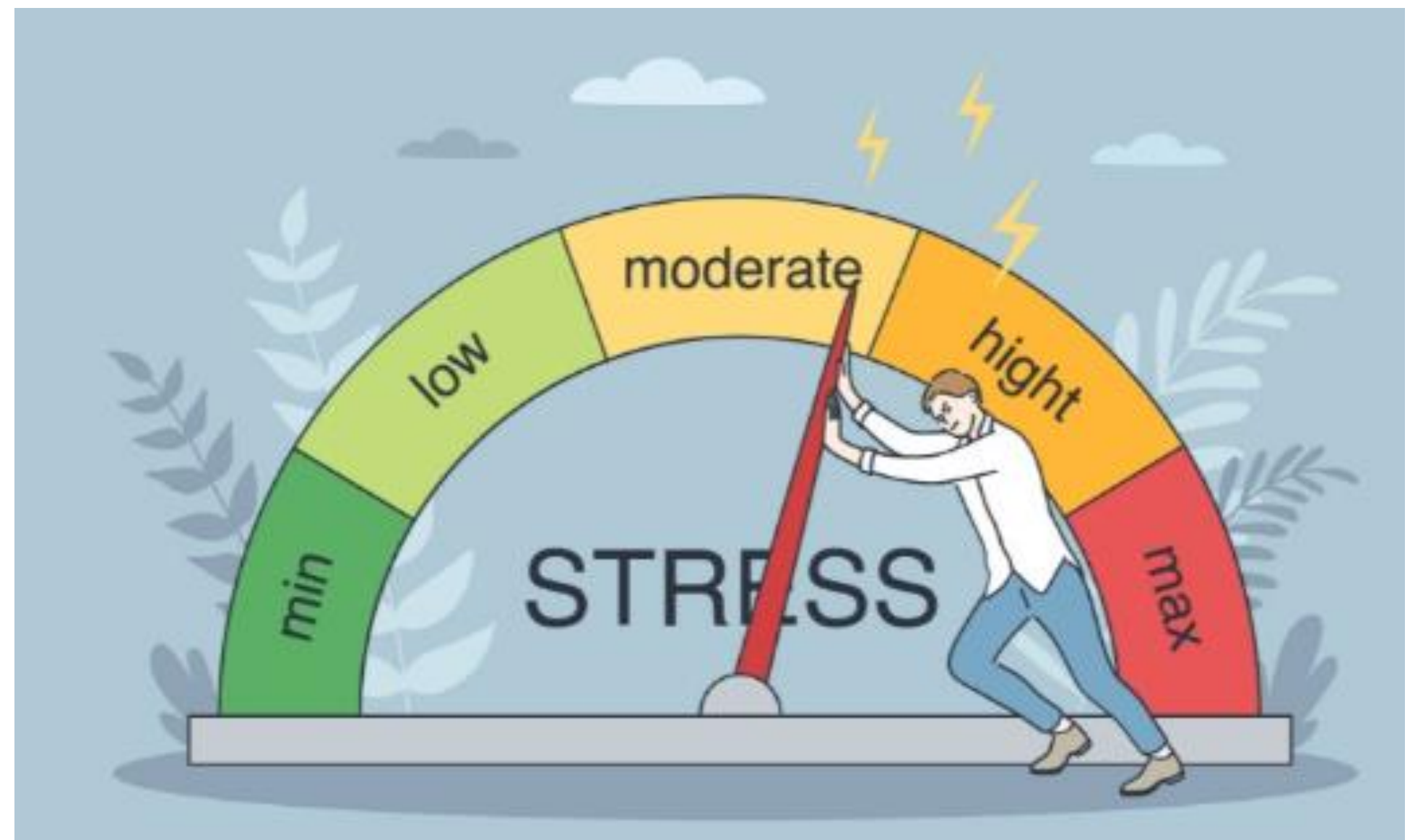
01

INTRODUCCIÓN A LAS PRUEBAS DE RENDIMIENTO

INTRODUCCIÓN A LAS PRUEBAS DE RENDIMIENTO

Las pruebas de rendimiento son un tipo de pruebas de software que se utilizan para evaluar el rendimiento, la capacidad y la estabilidad de una aplicación bajo diferentes condiciones de carga y estrés.

Estas pruebas se centran en medir la respuesta, la velocidad, la eficiencia y la escalabilidad del sistema, identificando posibles cuellos de botella, problemas de rendimiento y limitaciones.



INTRODUCCIÓN A LAS PRUEBAS DE RENDIMIENTO

Una prueba de carga es un tipo de prueba de rendimiento que verifica cómo funcionan los sistemas bajo una gran cantidad de usuarios virtuales simultáneos que realizan transacciones durante un cierto período de tiempo.

El objetivo de las pruebas de carga es encontrar dónde están los cuellos de botella de rendimiento y garantizar la estabilidad y el buen funcionamiento del canal digital antes del evento.

INTRODUCCIÓN A LAS PRUEBAS DE RENDIMIENTO

Cuándo usar la pruebas de carga y estrés

Las pruebas de carga deben realizarse continuamente para garantizar que su sistema esté siempre preparado para acoger muchos usuarios, por lo que debe implementarse en sus ciclos de mejora continua (utilizando la metodología de Atentus)

La siguiente figura representa una prueba de carga y estrés que realizamos para una web.

Esta prueba analiza agregar 500 usuarios cada 30 segundos hasta llegar a los 1.000 usuarios. Todo el proceso se dio ingresando a las 23:59.

INTRODUCCIÓN A LAS PRUEBAS DE RENDIMIENTO

Lo que identifica las pruebas de carga:

Determinar si la infraestructura actual es suficiente para ejecutar la aplicación

Sostenibilidad de la aplicación con respecto a la carga máxima de usuarios

Número de usuarios simultáneos que una aplicación puede admitir y escalabilidad para permitir que más usuarios accedan a ella.

En general, el rendimiento de los componentes que interactúan para la entrega del servicio.



00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00

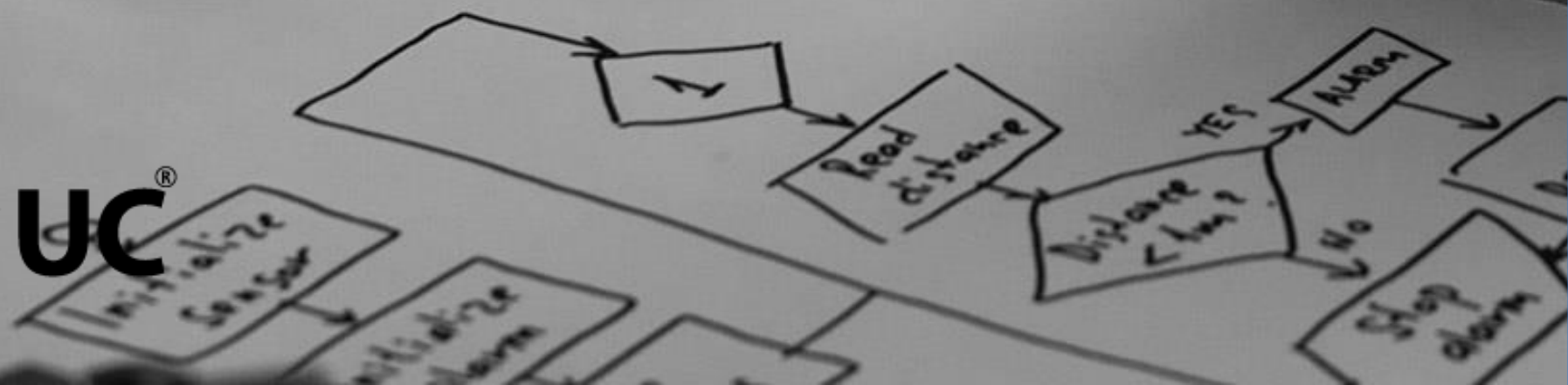
batch
Application "C:\Program Files\Microsoft
The program '[7676] Batch Calibration



02

JUSTIFICACIÓN Y BENEFICIOS

DuocUC[®]



JUSTIFICACIÓN Y BENEFICIOS

Identificación de problemas de rendimiento:

Las pruebas de rendimiento permiten identificar problemas como tiempos de respuesta lentos, alta latencia, cuellos de botella en el sistema o problemas de escalabilidad antes de que afecten a los usuarios en producción.

Optimización del rendimiento:

Al detectar y corregir los problemas de rendimiento, las pruebas de rendimiento ayudan a mejorar la eficiencia y la velocidad del sistema, lo que se traduce en una mejor experiencia del usuario.

JUSTIFICACIÓN Y BENEFICIOS

Garantía de escalabilidad:

Estas pruebas permiten evaluar cómo el sistema responde y se adapta a diferentes cargas de trabajo, asegurando que pueda manejar la demanda creciente de usuarios sin degradar el rendimiento.

Cumplimiento de requisitos no funcionales:

Las pruebas de rendimiento ayudan a verificar que el sistema cumpla con los requisitos de rendimiento establecidos en términos de tiempos de respuesta, capacidad de usuarios concurrentes, transacciones por segundo, entre otros.

JUSTIFICACIÓN Y BENEFICIOS

Optimización de recursos:

Al identificar ineficiencias y problemas de rendimiento, se pueden realizar ajustes en la configuración del sistema, en la arquitectura o en el código para aprovechar al máximo los recursos disponibles.

JUSTIFICACIÓN Y BENEFICIOS

Directrices para las pruebas de carga

Las pruebas de carga deben planificarse una vez que la aplicación sea funcionalmente estable.

Se debe decidir el número de usuarios para cada escenario o guiones

Evite la creación de registros detallados para conservar el espacio de E/S del

Trate de evitar la descarga de imágenes en el sitio

La consistencia del tiempo de respuesta durante el período transcurrido debe registrarse y lo mismo debe compararse con varias ejecuciones de prueba.

JUSTIFICACIÓN Y BENEFICIOS

Directrices para las pruebas de carga

Las pruebas de carga deben planificarse una vez que la aplicación sea funcionalmente estable.

Se debe decidir el número de usuarios para cada escenario o guiones

Evite la creación de registros detallados para conservar el espacio de E/S del

Trate de evitar la descarga de imágenes en el sitio

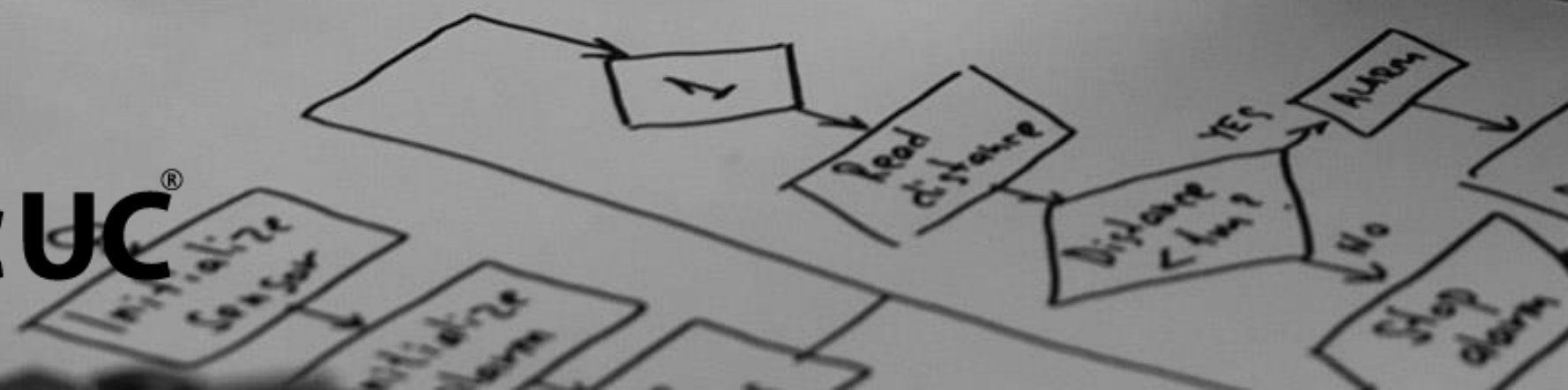
La consistencia del tiempo de respuesta durante el período transcurrido debe registrarse y lo mismo debe compararse con varias ejecuciones de prueba.



02

JUSTIFICACIÓN Y BENEFICIOS

DuocUC[®]



HERRAMIENTAS

Apache JMeter:

Es una herramienta de código abierto ampliamente utilizada para realizar pruebas de carga, rendimiento y estrés.

Permite simular diferentes escenarios de carga y generar informes detallados con métricas de rendimiento.

<https://jmeter.apache.org/>



HERRAMIENTAS

LoadRunner:

Es una herramienta comercial que permite realizar pruebas de carga y rendimiento a través de una variedad de protocolos de aplicación.

Proporciona capacidades para simular escenarios complejos y monitorear el rendimiento del sistema.

<https://www.microfocus.com/es-es/products/loadrunner-professional/overview>



HERRAMIENTAS

Gatling:

Es una herramienta de pruebas de rendimiento de código abierto basada en Scala.

Permite simular cargas de trabajo masivas y proporciona análisis en tiempo real del rendimiento del sistema.

<https://gatling.io/>



HERRAMIENTAS

Locust:

Es una herramienta de pruebas de rendimiento de código abierto escrita en Python.

Permite escribir escenarios de carga personalizados y monitorear el rendimiento en tiempo real.

<https://locust.io/>



HERRAMIENTAS

Apache Benchmark (ab):

Es una herramienta de línea de comandos que forma parte del paquete de software Apache HTTP Server.

Permite medir el rendimiento y la capacidad de respuesta de un servidor web

<https://httpd.apache.org/docs/2.4/programs/ab.html>



Conclusiones de la clase

DuocUC[®]



Conclusiones

- ✓ Las pruebas de rendimiento permiten identificar problemas como tiempos de respuesta lentos, alta latencia, cuellos de botella en el sistema o problemas de escalabilidad antes de que afecten a los usuarios en producción.
- ✓ Las pruebas de rendimiento son un tipo de pruebas de software que se utilizan para evaluar el rendimiento, la capacidad y la estabilidad de una aplicación bajo diferentes condiciones de carga y estrés

Bibliografía

- ✓ dotcom monitor. (15 de mayo 2019). Pruebas de rendimiento en línea (carga y estrés). <https://www.dotcom-monitor.com> Recuperado de <https://www.dotcom-monitor.com/wiki/es/knowledge-base/solucion-de-prueba-de-carga/>

Simulación y análisis de condiciones de carga extrema

Calidad de Software - CSY4111