

# **Evaluación de la Eficiencia en Productos de Software. Experiencia desde el Departamemto de Evaluación de Productos de Software de CALISOFT**

**MsC. Delvis Echeverría Perez**

Centro Nacional de Calidad del Software(CALISOFT), La Habana, Cuba, decheverria@uci.cu

**Ing. Ariannis Abella Paumier**

Centro Nacional de Calidad del Software(CALISOFT), La Habana, Cuba, abella@uci.cu

## **ABSTRACT**

The research is related to the evaluation of the characteristics of "efficiency" of ISO 9126, in the Software Products Evaluation Department (DEPSW) in CALISOFT, responsible for providing testing services to national and international companies in Cuba. Specifically within the characteristics in question we have identified a set of testing: containment, performance, load and stress. Errors or incidents which are generated in these tests have ratings provided by the DEPSW through experience these tests. For the execution of these testing was defined a procedure, that including automated tools as JMeter, technical and methods, also been proposed test case and base case with possible corrective actions found bug fixes. The proposal describes a close relationship to teaching undergraduate and graduate of the Informatic Sciences University in the training of students and professionals on topics related to the testing efficiency.

**Keywords:** Efficiency, Software Testing, Tools

## **RESUMEN**

La investigación está relacionada con la evaluación de la característica de "Eficiencia" de la Norma ISO 9126, en el Departamento de Evaluación de Productos de Software (DEPSW) de CALISOFT, encargado de brindar servicios de pruebas a empresas nacionales e internacionales en Cuba. Específicamente dentro de la característica en cuestión se han identificado un conjunto de pruebas: contención, rendimiento, carga y estrés. Los errores o incidencias que se generan de estas pruebas han sido clasificaciones dadas por el DEPSW, a través de la experiencia de estas pruebas. Para la ejecución de estas pruebas se definió un procedimiento donde se incluye herramientas automatizadas como el JMETTER, técnicas y métodos, además se han propuesto casos de pruebas y una base de caso con acciones correctivas y posibles soluciones a errores encontrados. Se describe en la propuesta la estrecha relación del DEPSW con la docencia de pregrado y postgrado con la Universidad de las Ciencias Informáticas en la capacitación de estudiantes y profesionales en temas relacionados con las pruebas de eficiencia.

**Palabras claves:** Eficiencia, Pruebas de Software, Herramientas

## **1. INTRODUCCIÓN**

En la ingeniería del software, las pruebas de eficiencia son las pruebas que se realizan, desde una perspectiva, para determinar lo rápido que realiza una tarea un sistema en condiciones particulares de trabajo. Las pruebas de rendimiento son un subconjunto de la ingeniería de pruebas, una práctica informática que se esfuerza por mejorar el rendimiento, englobándose en el diseño y la arquitectura de un sistema, antes incluso del esfuerzo inicial de la codificación.

Las pruebas de performance permiten conocer y mitigar los riesgos relacionados con el mal desempeño de las aplicaciones en los entornos de producción y realizar las correcciones necesarias antes de salir al mercado. Se

cuantifica la capacidad de la infraestructura, se validan los requerimientos de performance, la escalabilidad de las plataformas y del sistema a probar. De esta manera, la empresa puede conocer qué cantidad de clientes simultáneos soporta su producto, con tiempos y datos razonables sobre la infraestructura y las plataformas propuestas. Asimismo, puede saber si es suficiente el hardware para soportar el nivel propuesto de transacciones y qué expectativa de crecimiento soporta. Es relevante la selección de la herramienta adecuada, la definición de los escenarios de prueba y la configuración del entorno. Una vez ejecutadas las pruebas, es importante la evaluación de resultados para identificar los posibles problemas en la performance actual y las posibilidades de mejora sobre el desempeño del sistema. Se trabaja en conjunto con expertos de tecnología e infraestructura de la empresa. Hasta que un sistema no es sometido a carga, no podremos conocer su comportamiento, ya que las pruebas funcionales no suelen incluir niveles altos de concurrencia. El elemento diferenciador de las pruebas de rendimiento es la existencia de concurrencia en el sistema. Desde este punto de vista, no se consideran pruebas de rendimiento:

- Aquellas pruebas encaminadas a optimizar procesos que se ejecutan de manera aislada, aunque sean procesos lentos o pesados que se alarguen en el tiempo
- Las pruebas de volumen en las que únicamente se prueba cómo se comporta el sistema cuando funciona o procesa un volumen alto de datos.

Las pruebas de rendimiento son ejecutadas por medio de scripts automatizados, éstos se encargan de emular las acciones que realizaría un usuario final sobre la aplicación bajo pruebas. Los scripts se ejecutan en paralelo, cada uno de ellos emulando un “usuario virtual”, de esta forma, anticipando la carga esperada cuando el sistema pase a producción. Durante la ejecución de las pruebas, nuestros consultores se encargan de vigilar el sistema, que recibe la carga por medio de indicadores de rendimiento, esta acción es comúnmente llamada monitorización del sistema. En base a las métricas obtenidas, Globe Testing sugiere mejoras para optimizar el rendimiento del sistema y, de esta forma, mejorar los tiempos de respuesta de la aplicación.

La noche antes de un paso a producción suele estar caracterizada por pocas horas de sueño y mucha tensión, causada por la incertidumbre de no saber cómo se va a comportar el sistema una vez estén todos los usuarios trabajando en el entorno. Las pruebas de rendimiento tienen como objetivo anticipar los problemas que puedan ocurrir una vez la aplicación esté en producción. Hacer pruebas de rendimiento significa dormir bien, sabiendo que su sistema está preparado para la carga esperada. No realizar pruebas de rendimiento supone, en muchos casos, una pérdida económica, no solo causada por la falta de disponibilidad de sus sistemas y el impacto que esto tiene en su producción, sino también por el impacto que la falta de servicio tiene en el usuario final (ya sea un cliente que no volverá a usar su E-Commerce por falta de confianza).

## **2. DEPARTAMENTO DE EVALUACIÓN DE PRODUCTOS DE SOFTWARE**

CALISOFT es el centro responsable de la verificación y validación de productos, procesos y organizaciones según normas nacionales e internacionales. Además facilita la implementación de las mejores prácticas en el proceso de desarrollo y/o mantenimiento de software. Dentro de su estructura organizativa se encuentra el Departamento de Evaluación de Productos de Software (DEPSW). El DEPSW brinda servicios de pruebas de software a empresas nacionales e internacionales, en su procedimiento realiza diferentes tipos de pruebas que se han identificado dentro de las características de calidad de la Norma ISO 9126. Este es un estándar internacional para la evaluación de la calidad del software. El estándar está dividido en cuatro partes las cuales dirigen, realidad, métricas externas, métricas internas y calidad en las métricas de uso y expendido. El modelo de calidad clasifica la calidad del software en un conjunto estructurado de características y subcaracterísticas, para el caso de la eficiencia específica:

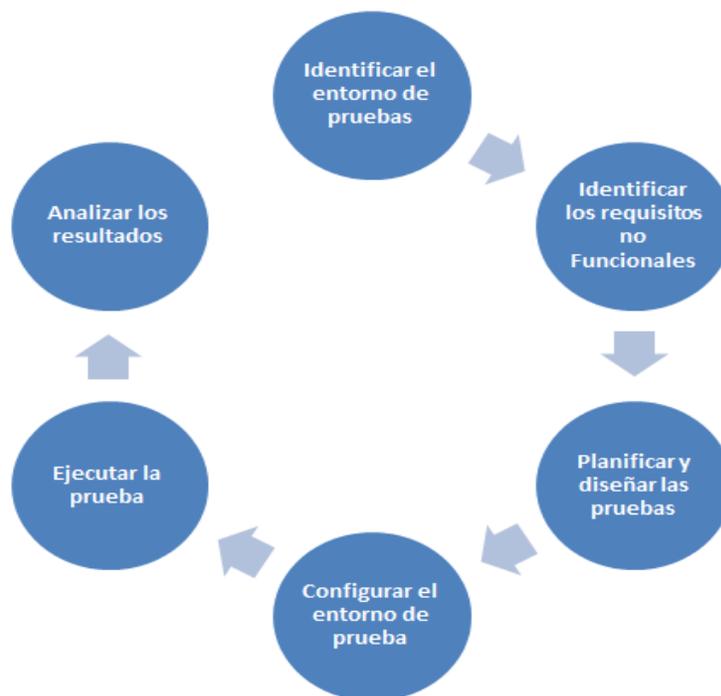
Eficiencia: Conjunto de atributos relacionados con la relación entre el nivel de desempeño del software y la cantidad de recursos necesarios bajo condiciones establecidas.

- Comportamiento en el tiempo - Atributos del software que se relacionan con los tiempos de respuesta y procesamiento y en las tasas de rendimientos en desempeñar su función.
- Comportamiento de recursos - Usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas.

Cabe destacar que en ocasiones algunos términos relacionados con estas pruebas tienden a confundir. El termino eficiencia especifica la característica de Calidad de la Norma ISO 9126. Dentro de esta característica se han identificado un conjunto de pruebas como son:

- **Carga:** Usada para validar y valorar la aceptabilidad de los límites operacionales de un sistema bajo carga de trabajo variable, mientras el sistema bajo prueba permanece constante. La variación en carga es simular la carga de trabajo promedio.
- **Estrés:** Enfocada a evaluar cómo el sistema responde bajo condiciones anormales. (Extrema sobrecarga, insuficiente memoria, servicios y hardware no disponible, recursos compartidos no disponible).
- **Contención:** Enfocada a la validación de las habilidades del elemento a probar para manejar aceptablemente la demanda de múltiples actores sobre un mismo recurso (registro de recursos, memoria, etc).
- **Rendimiento:** Enfocadas a monitorear el tiempo en flujo de ejecución, acceso a datos, en llamada a funciones y sistema para identificar y direccionar los cuellos de botellas y los procesos ineficientes.

El departamento de Pruebas de Software cuenta con un grupo que se dedica a llevar a cabo las pruebas dentro de la característica de calidad de Eficiencia. Este grupo presenta un procedimiento que se divide en las siguientes actividades:



**Figura 1: Procedimiento para Pruebas de eficiencia**

- **Identificar el entorno de pruebas:** Identificar el entorno físico de pruebas y el entorno de producción, así como las herramientas y recursos de que dispone el equipo de prueba. El entorno físico incluye hardware, software y configuraciones de red.
- **Identificar los requisitos no Funcionales:** Determinar el tiempo de respuesta, el rendimiento, la utilización de los recursos y los objetivos y limitaciones.
- **Planificar y diseñar las pruebas:** Identificar los principales escenarios, determinar la variabilidad de los usuarios y la forma de simular esa variabilidad, definir los datos de las pruebas, y establecer las métricas a recoger.
- **Configurar el entorno de prueba:** Preparar el entorno de prueba, herramientas y recursos necesarios para ejecutar cada una de las estrategias, así como las características y componentes disponibles para la prueba.
- **Ejecutar la prueba:** Ejecutar y validar las pruebas, los datos de las pruebas, y recoger los resultados.
- **Analizar los resultados:** Analizar y compartir los resultados de la prueba tanto individualmente, como con un equipo multidisciplinario.

### 3. HERRAMIENTAS AUTOMATIZADAS

El proceso de elaboración de pruebas automáticas no es un proceso complejo y sí necesario en cada uno de las líneas de producción. Varias son las ventajas por las cuales el equipo de calidad y de desarrollo del software, necesitan adoptar esta política.

- **Confiable:** Las pruebas realizan exactamente las mismas operaciones cada vez que se ejecutan, de tal modo se elimina en gran medida el error humano.
- **Repetible:** Se puede probar cómo el software reacciona bajo la ejecución repetida de las mismas operaciones.
- **Programable:** Se pueden programar las pruebas de manera sofisticada y automatizada.
- **Entendible:** Se puede construir un test de pruebas que cubra cada característica del uso de la aplicación.
- **Reutilizable:** Se puede reutilizar pruebas en diversas versiones, aun realizando cambios en la interfaz de usuario.
- **Software de una calidad mejor:** Porque se pueden funcionar más pruebas en menos tiempo y con pocos recursos.
- **Rápido:** Usuarios humanos son mucho más lentos que usuarios automatizados, por tanto el funcionamiento de las pruebas en herramientas automatizadas se realiza de manera más rápida.
- **Reducción de costos:** Como el número de los recursos para la prueba se reduce. Las inversiones asociadas disminuyen.

El procedimiento del DEPSW define un conjunto de herramientas automatizadas como apoyo. Dentro de ellas se encuentran:

**Jmeter:** Es un proyecto de Apache que puede ser utilizado como una herramienta de prueba para analizar y medir el desempeño de una variedad de servicios, con énfasis en aplicaciones web. Utilizar JMeter en aplicaciones web para la comprobación de los recursos del sistema, supone una mayor efectividad en el proceso y en la fiabilidad de los resultados. JMeter como herramienta de prueba dispone de varios componentes que facilitan la elaboración de los escenarios de prueba con la ventaja de simular para cada uno de esos escenarios miles de usuarios.

**Mantis:** Es un gestor de incidencias completo que te permite crear proyectos, asignarles incidencias, hacer seguimiento de las incidencias. Esta herramienta es una de las plataformas más extendidas en todo el mundo. Se trata de una herramienta libre donde los usuarios pueden administrar con facilidad el registro de bugs por proyectos asignando roles y prioridades. Es una herramienta OpenSource desarrollada en php y mysql, fácil de

instalar y muy flexible en su configuración. Podéis especificar un número indeterminado de estados para cada tarea (abierta, encaminada, testeada, devuelta, cerrada, reabierta) y tantos perfiles como necesitéis (programador, probador, coordinador, visualizador).

#### 4. GESTIÓN DE LOS ERRORES O INCIDENCIAS GENERADAS.

A través de la ejecución de las pruebas son generados los errores o las incidencias, los cuales se han clasificado de la siguiente manera:

- **Concurrencia de Usuarios.**
- **Tiempos de Respuesta.**
- **Errores HTTP:** Los códigos HTTP describen el estado de un URL cuando un visitante intenta accederlo. Esto incluye mensajes de error que aparecen cuando un visitante encuentra problemas en un sitio web. Los códigos son números de tres dígitos.

Hay cinco clases de códigos HTTP de error. El primer dígito determina su categoría:

- a) Un primer dígito de 1, 2 ó 3 representa una solicitud funcional. Puede encontrar información sobre estos códigos en este artículo.
- b) Un primer dígito de 4 representa un error del lado del cliente. Los códigos más comunes se extienden de 400 a 404.
- c) Un primer dígito de 5 representa un error del lado del servidor. Los códigos comunes se extienden de 500 a 510.

Estos errores o incidencia son gestionadas a través de un sistema automatizado “Mantis” descrito en el capítulo anterior.

#### 5. APORTES DEL DEPARTAMENTO DE EVALUACIÓN DE PRODUCTOS DE SOFTWARE.

##### 5.1 BASE DE CASOS DE POSIBLE SOLUCIONES DE ERRORES HTTP.

Esta Base de caso se confeccionó debido a que la mayoría de los clientes en ocasiones no presentan la solución a estos errores. Se confeccionó con los especialistas del DEPSW y los programadores de varios equipos de desarrollo. Se describe cada error y se enuncia como pudiera ser eliminado el error. A continuación algunos de los errores HTTP:

**Tabla 1: Descripción de los Errores HTTP**

Número	Error HTTP	Descripción
200	OK.	Es la respuesta estándar para una petición HTTP exitosa. La respuesta dependerá del método utilizado.
201	Created.	La petición ha sido completada y el resultado es que un nuevo recurso ha sido creado.
202	Accepted	La petición ha sido aceptada para procesarla, pero aún no ha sido completada.
300	Multiple Choices.	Indica múltiples opciones para el recurso que el cliente podría seguir. Por ejemplo, puede ser usado para presentar diferentes opciones de formatos de videos, listar archivos con diferentes extensiones, etc.
400	Bad Request,	El servidor web no pudo entender la solicitud debido a errores de sintaxis.
404	Not Found.	Este error puede deberse a un error de ortografía o de sintaxis

		en la URL o que se trate de un recurso que ya no existe en el servidor. Errores comunes que producen un código 404 en los navegadores, como por ejemplo el incluir la dirección de un archivo local o el omitir el protocolo en una URL absoluta
405	Method Not Allowed.	Se solicitó un recurso con un método no soportado por ese recurso. Por ejemplo, usar el método GET en un formulario que requiere que la data sea presentada vía POST.

## 5.2 DOCENCIA DE PREGRADO Y POSTGRADO.

En la actualidad el DEPSW, es el encargado de impartir la asignatura de Práctica de Investigación y Desarrollo, al segundo año de la carrera de Ingeniería en Ciencias Informáticas de la Universidad de las Ciencias Informáticas, dentro del programa de la asignatura se imparte la conferencia y laboratorio practica relacionado con la evaluación de la eficiencia. Además la Dirección de Postgrado de la misma universidad brinda el curso de postgrado “Evaluación de la eficiencia en productos de software” que también es preparado por el DEPSW. En ambas asignatura se imparten conocimientos teóricos y prácticos.

## CONCLUSIONES

Las pruebas relacionadas con la eficiencia en el departamento proporcionan una ayuda durante la fase de desarrollo y optimizan la construcción de componentes críticos en el rendimiento del sistema Pudiéndose lograr una mejor retroalimentación entre los equipos de desarrollo y el equipo de prueba. En la actualidad se cuenta con una Base de Casos de ayuda a solucionar los errores o incidencias generadas lográndose realizar varias iteraciones de pruebas y evaluar un producto listo para ser desplegado coherentemente con los requisitos no funcionales de eficiencia descritos al comienzo de las pruebas. Los involucrados en estas pruebas saldrán con una completa capacitación tanto a nivel de pregrado como postgrado, presentando conocimientos prácticos y teóricos.

## REFERENCES

- Almenares L..(2008). Cómo realizar Pruebas de Carga y Estrés en JMeter. Tesis de grado. Universidad de las Ciencias Informáticas. La Habana Cuba.
- Diaz F. (2012). Usando jmeter para pruebas de rendimiento.  
[http://www.linti.unlp.edu.ar/uploads/docs/usando\\_jmeter\\_para\\_pruebas\\_de\\_rendimiento.pdf](http://www.linti.unlp.edu.ar/uploads/docs/usando_jmeter_para_pruebas_de_rendimiento.pdf). (2013)
- Cpanel, Errores HTTP,  
(2010)[http://docs.cpanel.net/twiki/bin/view/11\\_28/Es/GuiaUsuarioCpanel/HTTPCodesES](http://docs.cpanel.net/twiki/bin/view/11_28/Es/GuiaUsuarioCpanel/HTTPCodesES). ( 2013).
- QATecnico, <http://qatecnico.blogspot.com/2012/03/pruebas-de-rendimiento-tipos-y.html>. ( 2013).
- Globe Testing, <http://www.globetesting.com/pruebas-de-rendimiento/>, ( 2013).
- QAustral, Mantis (2010), <http://qaustral.com.ar/sitio/mantis-manual-de-usuario/>, ( 2013).
- Apache Jmeter, <https://jmeter.apache.org>, (2013).