

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/291344205>

# Usando Jmeter para pruebas de rendimiento

Article · January 2008

CITATIONS

3

READS

3,321

4 authors, including:



Javier Díaz

Universidad Nacional de La Plata

154 PUBLICATIONS 97 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Recommender Systems for learning resources. Educational Data Mining. Learning Analytics [View project](#)



Internet del Futuro: Ciudades Digitales Inclusivas, Innovadoras y Sustentables, IoT, Ciberseguridad, Espacios de Aprendizaje del Futuro [View project](#)

# **Usando Jmeter para pruebas de rendimiento**

**F. Javier Diaz**

**Claudia M. Tzancoff Banchoff**

**Anahí S. Rodríguez**

**Valeria Soria**

{jdiaz, cbanchoff, anahi, valeria}@linti.unlp.edu.ar

LINTI. Fac. de Informática, Universidad Nacional de La Plata.

**La Plata, 1900, ARGENTINA**

## **Abstract**

Producing high quality software [1] implies developing systems which fulfill all the specified or implicit needs established by the user.

The software process is conformed by seven stages: Requirements analysis – Specification – Design and architecture – Programming – Testing – Documentation – Maintenance.

Software testing can be made in different stages of this process. It is important to emphasize on it being performed in early stages, so as to condition the posterior development.

When dealing with a web application, there are many aspects to analyze. Apart from the functional ones. Many are related to the specification and functional customization of the system, but there are other aspects related to the performance of the server, ease of navigation, security aspects, etc.

The goal of this article is to emphasize on the testing process of the use of a tool [2] which simplifies the analysis. We took as pilot case the analysis of the performance of a content manager implemented by the UNLP.

**Key words:** Quality, Testing, Performance Tests, Web applications.

## **Resumen**

Producir software de alta calidad [1] implica desarrollar sistemas que cumplan con la totalidad de las necesidades especificadas o implícitas establecidas por el usuario.

El proceso de software consta de siete etapas: Análisis de requisitos - Especificación - Diseño y Arquitectura - Programación - Prueba - Documentación - Mantenimiento.

Las pruebas del software pueden realizarse en distintas etapas de este proceso. Se destaca la importancia de que las mismas se realicen en etapas tempranas, pudiendo esto, obviamente, condicionar el posterior desarrollo.

Cuando se trata de una aplicación web, existen varios aspectos adicionales a analizar, además de los funcionales. Muchos tienen que ver con la especificación y adecuación funcional del sistema, pero hay otros aspectos que tienen que ver con la performance del servidor, la facilidad de navegación, aspectos de seguridad, etc.

El objetivo de este artículo es destacar en el proceso de testing el uso de una herramienta [2] que simplifique el análisis. Se tomó como caso piloto, el análisis de rendimiento de un manejador de contenidos implementado por la UNLP.

**Palabras Claves:** Calidad, Testing, Pruebas de Rendimiento, Aplicaciones web.

## **Introducción**

Las pruebas de rendimiento realizados sobre computadoras, redes, software u otros dispositivos, son utilizados para determinar la velocidad y eficiencia de los mismos. Este procedimiento puede involucrar tanto tests cuantitativos, por ejemplo, medir tiempos de respuesta o cantidad en millones de líneas de código, como tests cualitativos, en los cuales se evalúa fiabilidad, escalabilidad e interoperabilidad. Estas pruebas de rendimiento pueden ser realizadas a través de herramientas que proveen pruebas de estrés, que permiten determinar la estabilidad del sistema. [3]  
[4]

Las limitaciones en los tiempos de respuesta de un sitio web y una aplicación de escritorio son similares, y no han cambiado en el transcurso de los años. Cabe aclarar que en el caso de los sitios web el tiempo está muy relacionado a la velocidad del enlace donde se esté “navegando”.

Según el autor Jakob Nielsen, en el libro “Usability Engineering”[5] existen tres límites importantes en el tiempo de respuesta: [6] [7]

- *0,1 segundo*: es el límite en el cual el usuario siente que esta “*manipulando*” los objetos desde la interfaz de usuario.
- *1 segundo*: es el límite en el cual el usuario siente que está navegando libremente sin esperar demasiado una respuesta del servidor.
- *10 segundos*: es el límite en el cual se pierde la atención del usuario, si la respuesta tarda más de 10 segundos se deberá indicar algún mecanismo por el cual el usuario pueda interrumpir la operación.

En nuestro caso particular, este tiempo está condicionado a los siguientes puntos:

- El servidor testeado se encuentra en la misma red en la cual se realizaron las pruebas.
- Velocidad de conexión del servidor.
- Velocidad de conexión del cliente.
- Tiempo en el cual el navegador web tarda para dibujar la página (tiempo muy pequeño).
- Rendimiento de la red en el momento de la prueba.

## **Características de la Prueba**

En esta sección se describirá tanto la herramienta utilizada como las pruebas realizadas.

### **La herramienta**

Para analizar el tiempo de respuesta del servidor se utilizó la herramienta Jmeter [8]. La versión utilizada de Jmeter durante este trabajo es la 2.3.1.

Jmeter es una herramienta open source muy completa, implementada en Java que permite realizar test de comportamiento funcional y medir el rendimiento. También se puede utilizar para realizar pruebas de estrés, por ejemplo, en un servidor, y poner a prueba su rendimiento [9].

Para estas pruebas, se configuró un servidor Proxy que provee Jmeter, para poder construir un camino de navegación aleatorio, y así simular la visita de un usuario.

Para poder evaluar los resultados se utilizaron 3 (tres) componentes provistos por la herramienta[10] [11].

- Summary Report: Permite visualizar los resultados del test realizado, en una tabla. Los datos que presenta son:
  - *Label*: etiqueta de la muestra
  - *#Muestras*: cantidad de thread utilizados para la URL.
  - *Media*: tiempo promedio en milisegundos para un conjunto de resultados.
  - *Min*: tiempo mínimo que demora un thread en acceder a una página.
  - *Max*: tiempo máximo que demora un thread en acceder a una página
  - *Rendimiento*: rendimiento medido en los requerimiento por segundo / minuto / hora.
  - *Kb/sec*: rendimiento medido en Kbytes por segundo.
  - *Media en bytes*: tamaño medio de respuesta del servidor (en bytes).
- Agregate Graph: Esta componente es similar a la anterior, pero permite obtener resultados más precisos. Utiliza más memoria, ya que calcula la mediana y la línea al 90%, la cual requieren que todos los datos estén almacenados. Los datos que se presentan son:
  - *URL* : etiqueta de la muestra
  - *#Muestras*: cantidad de Thread utilizados para la URL.
  - *Media*: tiempo promedio en milisegundos para un conjunto de resultados.
  - *Mediana*: valor en tiempo del percentil 50.
  - *Línea de 90%*: máximo tiempo utilizado por el 90% de la muestra, al resto de la misma le llevo más tiempo.
  - *Min*: tiempo mínimo de la muestra de una determinada URL.
  - *Max*: tiempo máximo de la muestra de una determinada URL.
  - *%Error*: porcentaje de requerimientos con errores.
  - *Rendimiento*: rendimiento medido en los requerimiento por segundo / minuto / hora.
  - *KB/sec*: rendimiento medido en Kbytes por segundo.

- Grafico de Resultados: Esta componente permite visualizar gráficamente los siguientes datos: media, mediana, dispersión y el rendimiento (representado como el número actual de requerimientos/minutos que el servidor maneja).

### **La Prueba**

La prueba realizada consistió en definir 3 tests de 100, 50 y 25 threads cada uno, los cuales simulan 100, 50 y 25 accesos de usuarios respectivamente.

Se definieron una lista de enlaces a los que se simuló el acceso aleatorio y a partir de ahí, se recolectaron los datos necesarios para su interpretación.

### **El análisis**

Se analizaron los resultados a través de un intervalo de confianza<sup>1</sup> con un nivel de confianza al 95%.

Para un primer análisis, se supone que la población tiene una distribución Normal.

Para un segundo análisis, dado que la muestra es grande, no se requiere hacer la suposición de que la muestra tiene una distribución Normal ya que por el Teorema Central del Límite (TCL), para **n** grande implica que  $\bar{X}$  tiene una distribución aproximadamente Normal sin importar la naturaleza de la distribución poblacional [12].

---

<sup>1</sup> Se llama *intervalo de confianza* a un intervalo de valores alrededor de un parámetro muestral en los que, con una probabilidad o nivel de confianza determinado, se situará el parámetro poblacional a estimar. [13].

## Resultados Detallados

### Prueba Nro. 1

La primera prueba fue realizada el día 10/06/08 a las 16:07 hs, Se configuraron 50 threads, cada "1 seg". Los valores totales obtenidos por la componente "Aggregate Graph" se muestran en la Tabla 1.

	# Muestras	Media	Mediana	Línea de 90%	Min	Máx	% Error	Rendimiento	Kb/sec
<b>TOTAL</b>	3850	5405	1078	15000	0	125360	1.3%	8,3/sec	79

Tabla 1: Valores correspondientes a la Prueba 1.

Como puede verse, el tiempo promedio para acceder a una página es 5,405 segundos, realizándose un total de 3850 requerimientos al servidor.

El tiempo total utilizado para los 50 threads se puede calcular con la siguiente fórmula:

$$\text{Tiempo Total} = \# \text{Muestras} * \text{Media} = 3850 * 5405 = 20809250 \text{ milisegundos}$$

El tiempo promedio total requerido por cada thread, se puede calcular de la siguiente manera:

$$((\text{Tiempo Total} / 1000) / 60) / \text{cantidad de Thread} = ((25535400 / 1000) / 60) / 50 = 6,936416 \text{ minutos}$$

### Análisis realizado

En primer lugar se evaluaron los resultados obtenidos a través de un intervalo de confianza para una distribución Normal al 95%. La fórmula del mismo es la siguiente:

$$[ TP - Z_{0.95} * D/\sqrt{n}, TP + Z_{0.95} * D/\sqrt{n} ]$$

Donde:

- Tiempo promedio (TP) de respuesta es: 5405

- Desviación (D) es: 11443.
- Tamaño de la muestra (n) es de: 3850
- $Z_{0,95}$  : 1,96

El intervalo resultante es el siguiente:

$$[ 5405 - 1,96 * 11443/\sqrt{3850}, 5405 + 1,96 * 11443/\sqrt{3850} ]$$

= [5043,535539 ; 5766,464461 ] en milisegundos.

= [5,043535539 ; 5,766464461 ] en segundos.

Por lo tanto, se puede esperar que el tiempo de respuesta promedio esté entre 5 y 5,7 segundos para una cantidad de 50 usuarios simultáneos realizando 3850 solicitudes.

En segundo lugar, se evaluaron los resultados obtenidos a través de un intervalo de confianza al 95% para muestras grandes. La fórmula del mismo es la siguiente:

$$[ TP - Z_{0,95} * S/\sqrt{n}, TP + Z_{0,95} * S/\sqrt{n} ]$$

Donde:

- Tiempo promedio (TP) de respuesta es: 5405
- Estimador del Desvío (S) es:  $\sqrt{(\sum x^2 - (\sum x)^2/n)/n-1} = 85971,17312$
- Tamaño de la muestra (n) es de: 3850
- $Z_{0,95}$  : 1,96

El intervalo resultante es el siguiente:

$$[ 5405 - 1,96 * 85971,17312/\sqrt{3850}, 5405 + 1,96 * 85971,17312/\sqrt{3850} ]$$

= [ 2689,320215 ; 8120,679785 ] en milisegundos.

= [ 2,689320215 ; 8,120679785 ] en segundos.

En este caso, se puede esperar que el tiempo de respuesta promedio esté entre 2 y 8 segundos para una cantidad de 50 usuarios simultáneos realizando 3850 solicitudes.



## **Prueba Nro. 2**

La segunda prueba fue realizada el día 11/06/08 a las 11:12 hs, se configuraron 25 threads, cada "1 seg". Se utilizó la misma lista de enlaces utilizada en la Prueba Nro. 1. Los valores totales obtenidos por la componente "Aggregate Graph" se muestran en la Tabla 2.

	#Muestras	Media	Mediana	Línea de 90%	Min	Máx	% Error	Rendimiento	Kb/sec
<b>TOTAL</b>	1925	1294	15	1531	0	123360	1.3%	11,7/sec	111,8

Tabla 2: Valores correspondientes a la Prueba 2.

Como puede verse en la tabla anterior, el tiempo promedio para acceder a una página es 1,294 segundos, realizándose un total de 1925 requerimientos al servidor.

El tiempo total utilizado para los 25 threads y el requerido para cada thread se calcularon de la misma manera que en la Prueba 1:

$$\text{Tiempo Total} = \# \text{Muestras} * \text{Media} = 1925 * 1294 = 2490950 \text{ milisegundos}$$

$$((\text{Tiempo Total} / 1000) / 60) / \text{cantidad de Thread} = ((2490950 / 1000) / 60) / 50 = 1,66063 \text{ en minutos}$$

## **Análisis realizado**

El análisis para la segunda prueba, fue realizado de manera similar que en la primera, usando las mismas fórmulas.

El intervalo resultante, utilizando un intervalo de confianza para una distribución Normal al 95%, es:

$$[ 1294 - 1,96 * 8783 / \sqrt{1925}, 1294 + 1,96 * 8783 / \sqrt{1925} ]$$

$$= [ 902,5390776 ; 1687,460922 ] \text{ en milisegundos}$$

$$= [ 0,902539077 ; 1,687460922 ] \text{ en segundos}$$

Esto significa que, puede esperarse un tiempo de respuesta promedio entre 0,9 y 1,7 segundos para una cantidad de 25 usuarios simultáneos realizando 1925 solicitudes.

Con un intervalo de confianza al 95% para muestras grandes, el intervalo resultante es el siguiente:

$$[ 1295 - 1,96 * 56413,6792/\sqrt{1924}, 1295 + 1,96 * 56413,6792/\sqrt{1924}]$$

= [ -1225,797515 ; 3815,797515 ] en milisegundos.

= [ -1,225797515 ; 3,815797515 ] en segundos.

Como el límite inferior es negativo, y la medida del tiempo nunca puede ser negativa, se puede esperar que el tiempo de respuesta promedio esté entre 0 y 3,8 segundos para una cantidad de 25 usuarios simultáneos realizando 1295 solicitudes.

### **Prueba Nro. 3**

La tercera prueba fue realizada el día 11/06/08 a las 12:08 hs, se configuraron 100 threads, cada "1 seg". Se utilizó la misma lista de enlaces utilizada en la Prueba Nro. 1. La Tabla 3 muestra los valores obtenidos de la componente "Aggregate Graph".

	#Muestras	Media	Mediana	Línea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
TOTAL	7300	12526	250	29265	0	696594	0,42 %	5,7/sec	102,5

Tabla 3: Valores correspondientes a la Prueba 3.

Como puede verse en la tabla anterior, el tiempo promedio para acceder a una página es 12,526 segundos, realizándose un total de 7300 requerimientos al servidor.

El tiempo total utilizado para los 100 threads es de 91439800 milisegundos y el tiempo promedio total requerido por cada thread es de 15,23996 en minutos.

### **Análisis realizado**

Como en las pruebas anteriores, se calcularon los intervalos en base a un intervalo de confianza para una distribución Normal al 95% y en base a un intervalo de confianza al 95% para muestras grandes.

Los datos obtenidos son:

- [ 11,41498972 ; 13,63701028 ] en segundos para el primer caso y,
- [ 7,79171985 ; 17,26028015 ] en segundos para el segundo.

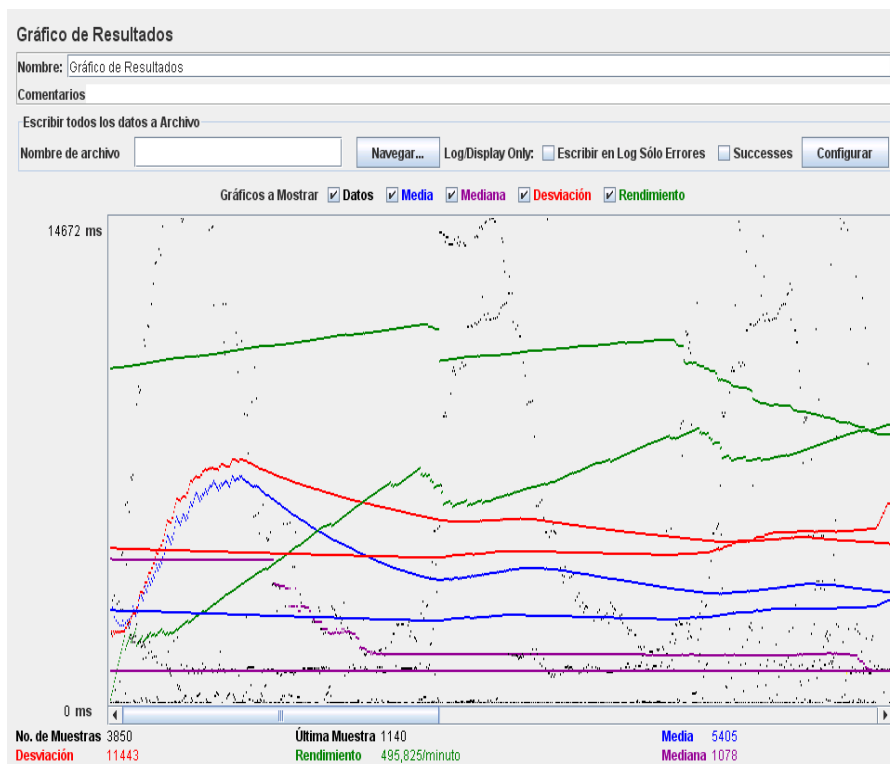
Por lo tanto, se puede esperar que el tiempo de respuesta promedio esté entre 11 y 13 segundos para una cantidad de 100 usuarios simultáneos realizando 7300 solicitudes y que el tiempo de respuesta promedio esté entre 7,8 y 17,2 segundos para la misma cantidad de muestras.

### **Resultados Generales**

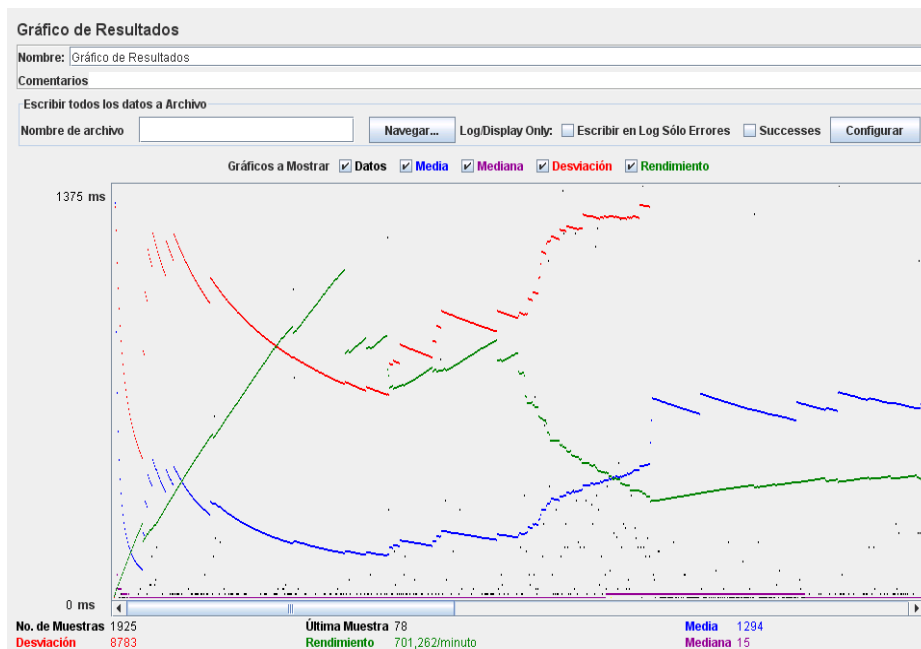
Como se mencionó al comienzo de este artículo, Jmeter provee una componente que permite visualizar los datos obtenidos en forma gráfica.

### **Gráficos Obtenidos**

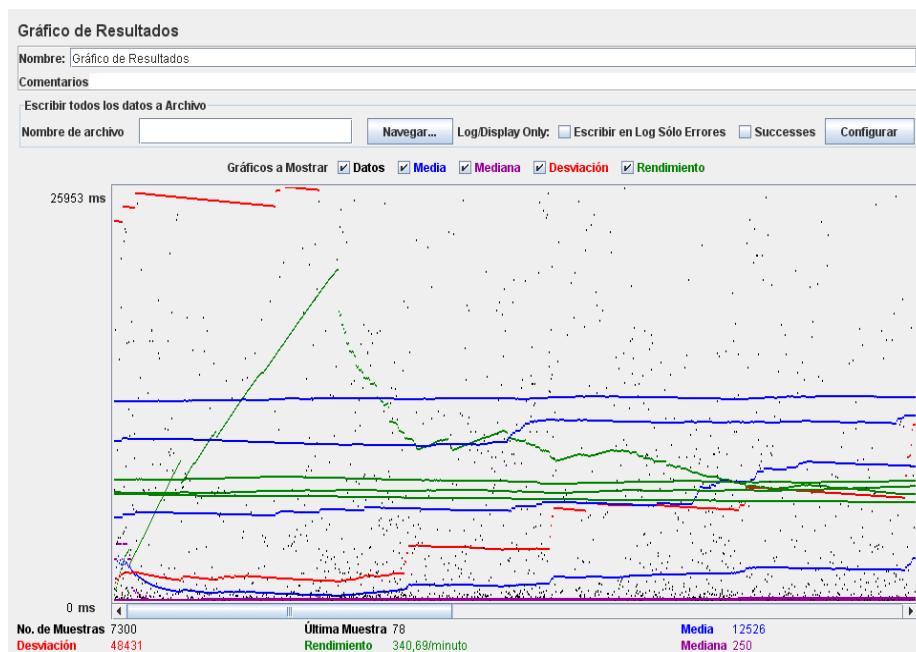
Las Figuras 1, 2 y 3 muestran los datos obtenidos para la prueba 1, 2 y 3 respectivamente.



*Figura 1: Valores correspondientes a la Prueba 1.*



*Figura 2: Valores correspondientes a la Prueba 2.*



*Figura 3: Valores correspondientes a la Prueba 3.*

## Tabla Comparativa

La Tabla 4 permite observar un resumen de los datos informados por las pruebas anteriores:

# Thread	# muestras	Rendimiento	Tiempo medio de Respuesta	Dispersión	% Error
25	1925	701,262/min	1294	8783	1,30%
50	3850	495,825/min	5405	11443	1,30%
100	7300	340,69/min	12526	48431	0,42%

*Tabla 4: Datos obtenidos en las tres pruebas*

Un resumen de los datos estadísticos analizados pueden verse en la Tabla 5.

#Thread	IC Distribución Normal al 95%	IC al 95% usando TCL
25	[0,902539077 ; 1,687460922] en seg	[-1,225797515 ; 3,815797515] en seg
50	[5,043535539 ; 5,766464461] en seg	[2,689320215 ; 8,120679785] en seg
100	[11,41498972 ; 13,63701028] en seg	[7,79171985 ; 17,26028015] en seg

*Tabla 5: Datos estadísticos en las tres pruebas*

## **Conclusión**

Si bien la etapa de pruebas funcionales de una aplicación son sumamente importantes, ya que a partir de las mismas se podrá contar con la aprobación del usuario final, el tipo de pruebas mencionadas en el presente artículo son también críticas. Una mala configuración del servidor web, o de la red donde correrá el mismo (con la aplicación residente en él) puede influenciar negativamente el uso de la misma.

En este trabajo se evaluó la utilización de la herramienta Jmeter, para la etapa de testeo de rendimiento del servidor web del sitio de la Universidad Nacional de La Plata.

Al tratarse de un portal al cual acceden muchos usuarios en forma simultánea, es indispensable estar seguros de una buena configuración del servidor donde correrá.

Las primeras pruebas realizadas dieron tiempos de respuestas poco aceptables para una cierta cantidad de usuarios concurrentes. El uso de la herramienta JMeter fue de gran ayuda para realizar estas pruebas, gracias a su flexibilidad de configuración y claridad de visualización de los resultados de las pruebas, lo que ayuda a su interpretación.

A partir del análisis de estos datos y, teniendo en cuenta los parámetros sugeridos desde el área HCI [5] se sugirió al grupo de desarrollo una reconfiguración del servidor web hasta alcanzar los valores óptimos de tiempo de respuesta.

## **Referencias**

- [1] ISO8402-1986
- [2] <http://www.opensourcetesting.org/performance.php>
- [3] [http://en.wikipedia.org/wiki/Performance\\_testing](http://en.wikipedia.org/wiki/Performance_testing)
- [4] [http://en.wikipedia.org/wiki/Stress\\_testing\\_%28software%29](http://en.wikipedia.org/wiki/Stress_testing_%28software%29)
- [5] “Usability Engineering” autor: Jakob Nielsen, publicado por: Morgan Kaufmann, San Francisco, 1994. ISBN 0-12-518406-9
- [6] <http://www.useit.com/alertbox/9703a.html>
- [7] <http://www.useit.com/papers/responsetime.html>

[8] <http://jakarta.apache.org/jmeter/>

[9] <http://www.osmosislatina.com/jmeter/basico.htm>

[10] [http://jakarta.apache.org/jmeter/usermanual/component\\_reference.html](http://jakarta.apache.org/jmeter/usermanual/component_reference.html)

[11] “Testes de Desempenho com o Tidia-Ae e o Sakai Utilizando o Benchmark Jmeter”. Tiago Caminha Gaspar; Sandro Lopes Bianchini; Flávia Linhalis; César Augusto Camillo Teixeira; Maria da Graça Campos Pimentel. <http://lince.dc.ufscar.br/home/projetos/tidia-ae%20II%20-%20Lince/relatorios/relatorio.pdf>

[12] “Probabilidades y Estadísticas para Ingeniería y Ciencias” – 6ta edición – Jay L. Devore

[13] [http://es.wikipedia.org/wiki/Intervalo\\_de\\_confianza](http://es.wikipedia.org/wiki/Intervalo_de_confianza)