



Ingeniería civil en Informática
Facultad de Ingeniería

Tarea 02: Desarrollo de App Web

Proyecto: Cálculo y Optimización de rendimiento usando Jmeter

Integrantes	Felipe Herrera Moraga
	Rosa Velásquez Rojas

22 Mayo 2015

Contenido

Descripción de la Etapa Inicial	3
Consulta simple:	3
Consulta media:	3
Consulta compleja:	4
Iteraciones	4
1era Iteración (Sin bootstrap)	4
Gráfico de resultados de la consulta Simple:	5
Gráfico de resultados de la consulta media:.....	5
Gráfico de resultados de la consulta compleja:	6
2da Iteración (Paginación).....	6
Gráfico de resultados de la consulta Simple:	6
Gráfico de resultados de la consulta media:.....	7
Gráfico de resultados de la consulta compleja:	7
3era Iteración (caché).....	8
Gráfico de resultados de la consulta Simple:	8
Gráfico de resultados de la consulta media:.....	9
Gráfico de resultados de la consulta Compleja:	9
4ta Iteración (indexación DB)	10
Gráfico de resultados de la consulta Simple:	10
Gráfico de resultados de la consulta media:.....	11
Gráfico de resultados de la consulta compleja:	11
5ta Iteración (Cambio de versión en la tecnología, y sistema operativo).....	12
Gráfico de resultados de la consulta Simple:	12
Gráfico de resultados de la consulta media:.....	13
Gráfico de resultados de la consulta compleja:	13
Tabla comparativa de las iteración	14
Consulta Simple	14
Consulta Media	14
Consulta Compleja	14
Conclusión	15

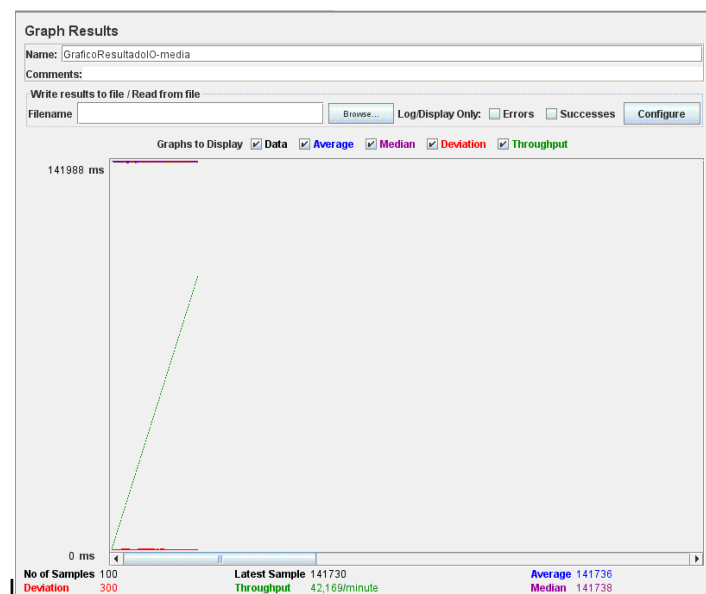
Descripción de la Etapa Inicial

Para la realización de las pruebas, se creó un proyecto realizado en el framework Codeigniter 2.2 y Bootstrap, al cual se le hicieron pruebas de carga con 100 hilos por segundo. Para mostrar el funcionamiento del proyecto a continuación se muestran las imágenes obtenidas desde Jmeter.

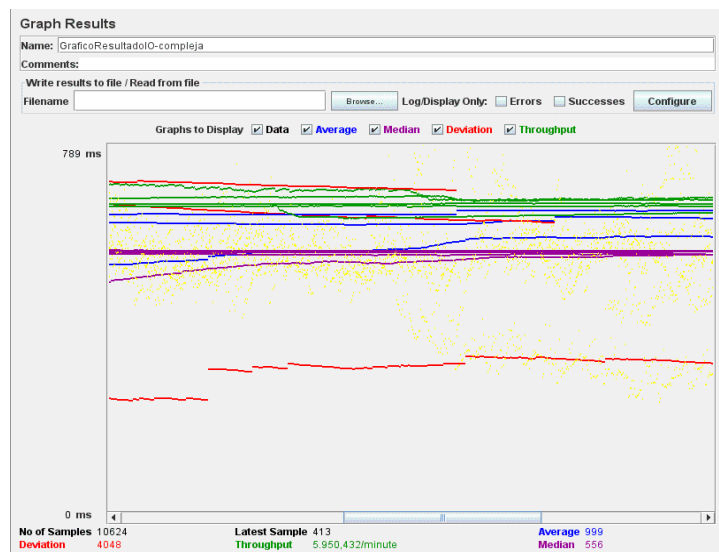
Consulta simple:



Consulta media:



Consulta compleja:



Iteraciones

Debido a los altos márgenes con respecto al porcentaje de error arrojado en cada una de las consultas por Apache Jmeter es que se piensa en optimizar este proyecto. A continuación, se detallan los procedimientos y la justificación de cada una de las iteraciones realizadas.

1era Iteración (Sin bootstrap)

En esta iteración, se decidió eliminar bootstrap como primera optimización, para que el envío de datos desde el servidor disminuyera y repartiera de mejor forma el ancho de banda entre los usuarios. A continuación se presentan los gráficos de las 3 consultas realizadas: simple, media y compleja, respectivamente.

Gráfico de resultados de la consulta Simple:

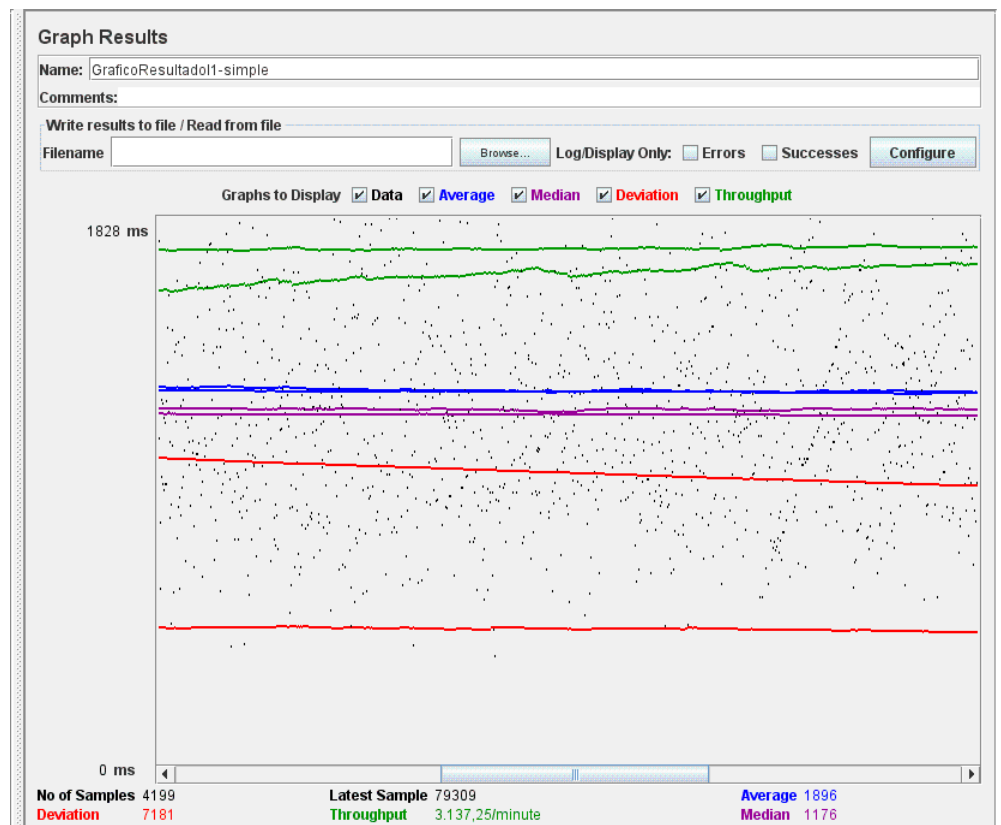


Gráfico de resultados de la consulta media:

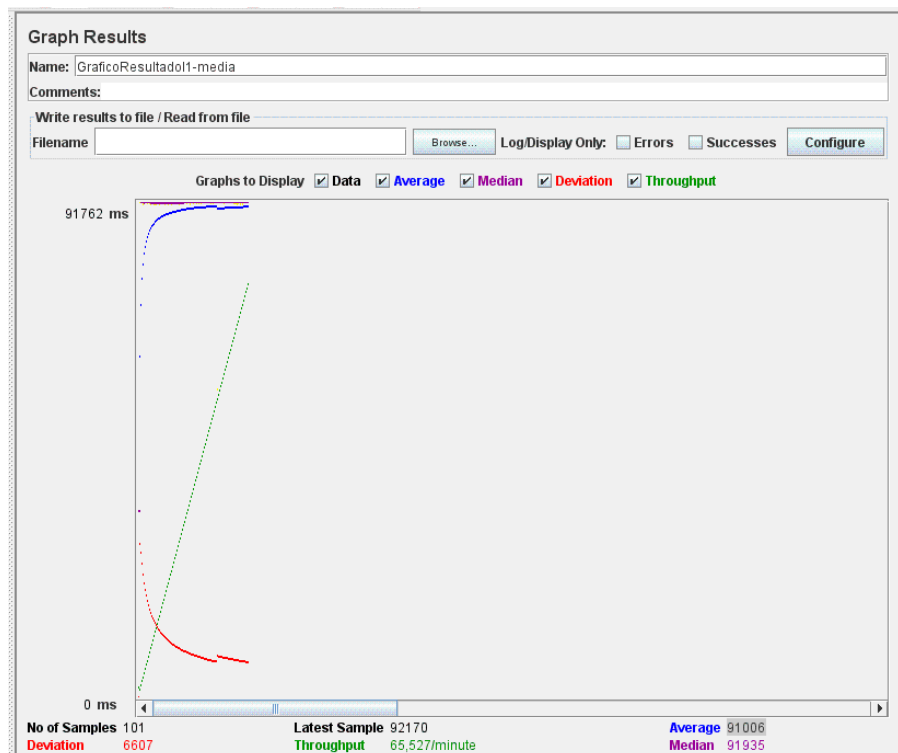
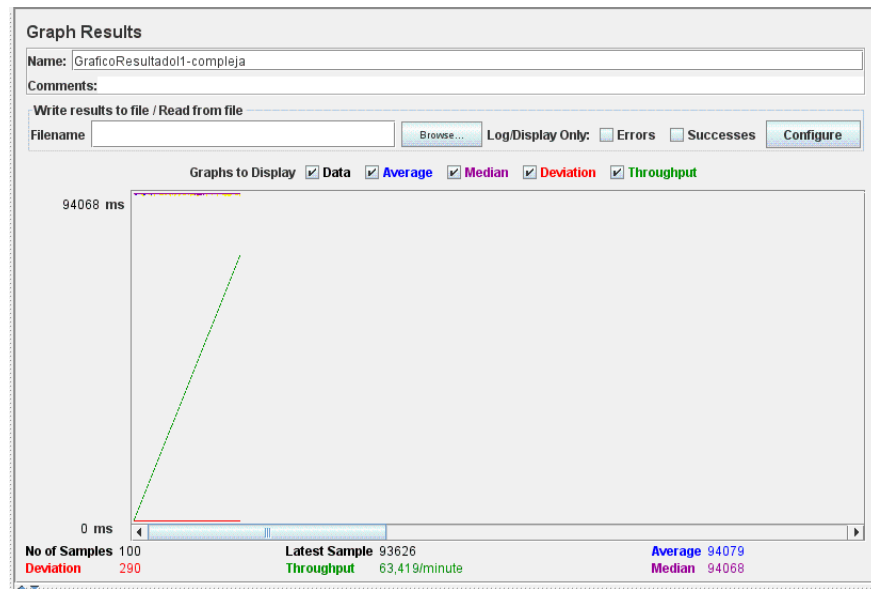


Gráfico de resultados de la consulta compleja:



2da Iteración (Paginación)

Debido a que la consulta media y la consulta compleja arrojaban demasiados datos en una vista, decidimos paginarlas, mostrando 100 filas por página para que el tiempo, ya que teóricamente cargar la vista de 100 datos, es mucho más rápido que cargar la vista de los 8600 datos.

Gráfico de resultados de la consulta Simple:

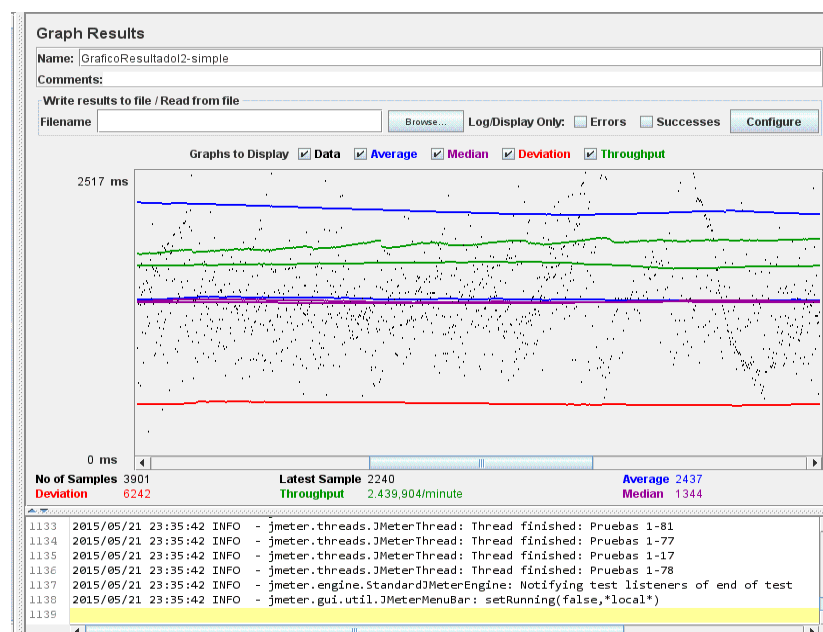


Gráfico de resultados de la consulta media:

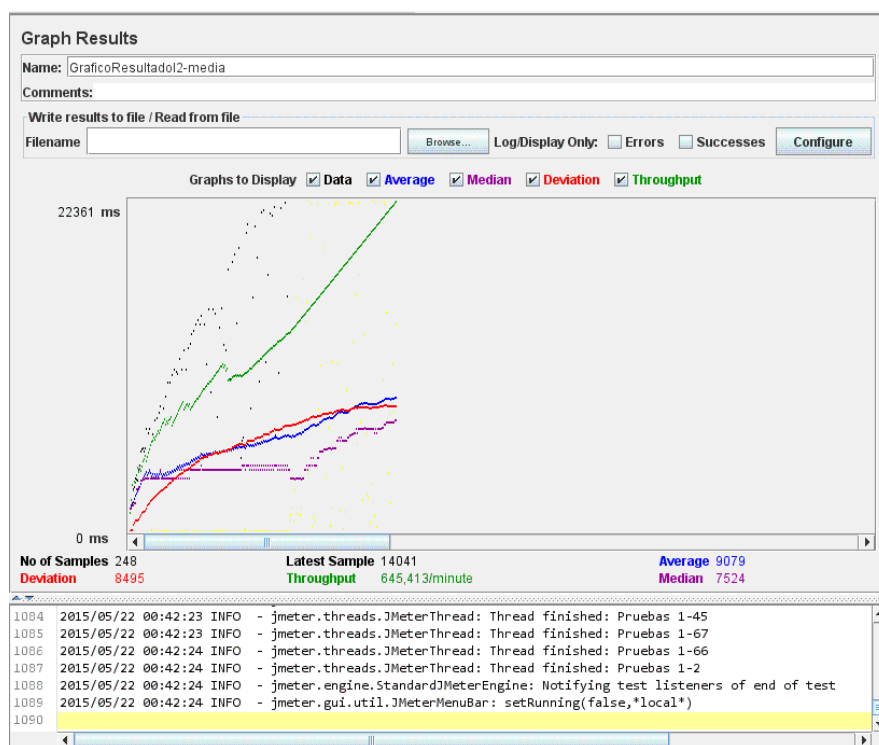
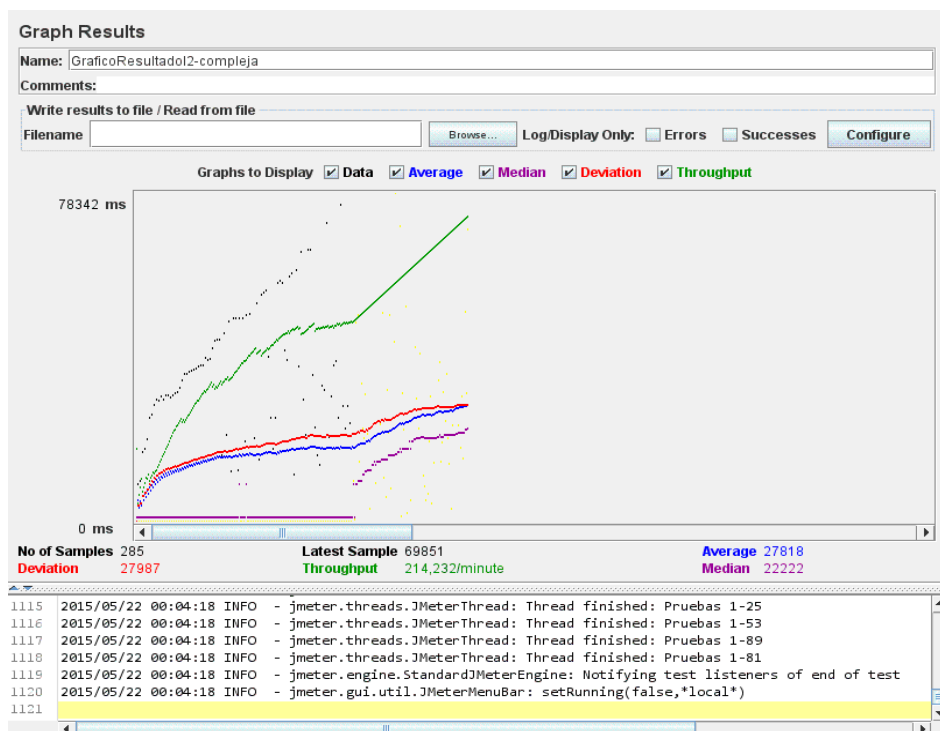


Gráfico de resultados de la consulta compleja:



3era Iteración (caché)

Para esta iteración activamos la memoria caché del controlador de codeigniter y da la base de datos (en la configuración de codeigniter en el archivo database.php) para que al realizar las consultas no se tuvieran que descargar denuevo los datos, y además guardar en RAM las últimas consultas realizadas.

Gráfico de resultados de la consulta Simple:

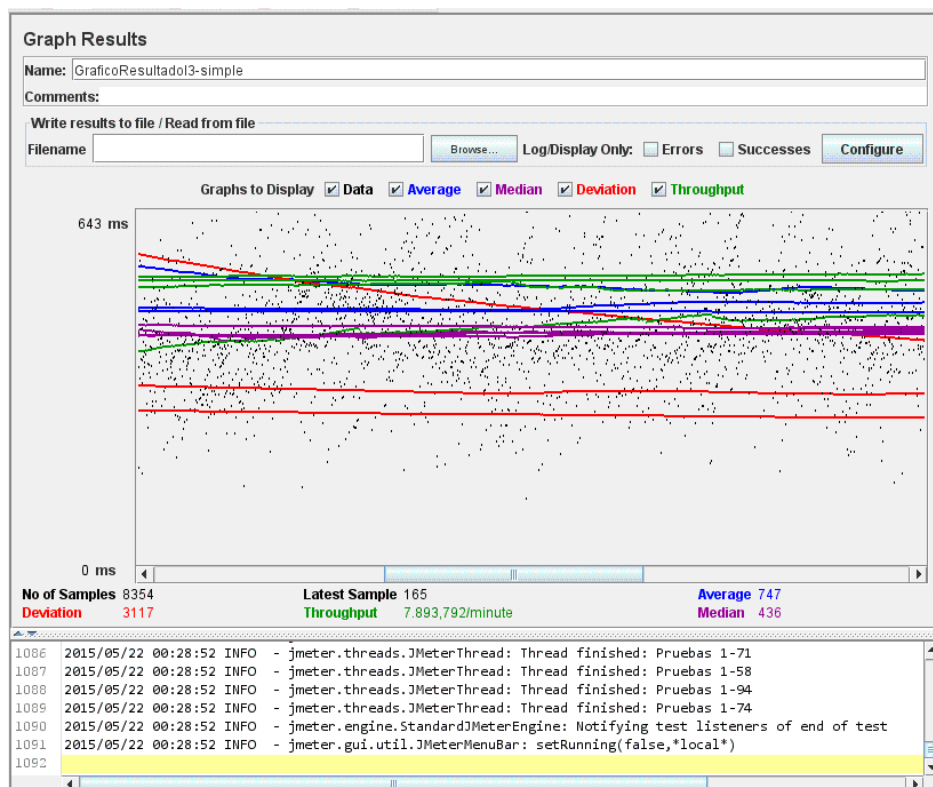


Gráfico de resultados de la consulta media:

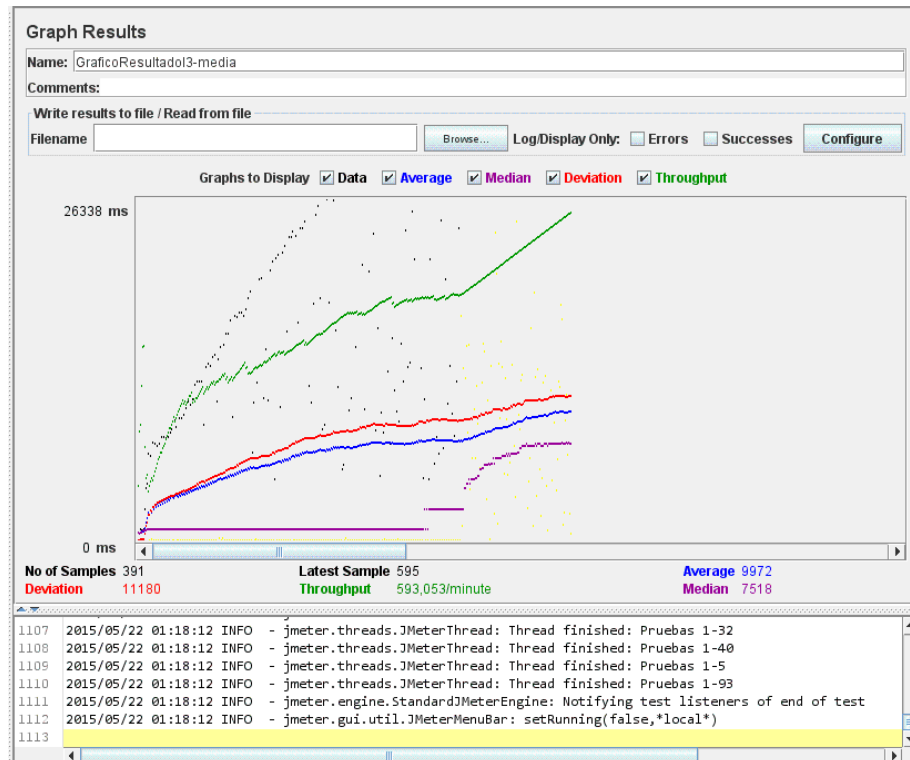


Gráfico de resultados de la consulta Compleja:



4ta Iteración (indexación DB)

Para esta iteración se consideró indexar la BD, con el fin de obtener un acceso rápido a esta, ya que al agregar un identificador a cada fila de una tabla permite aumentar la velocidad de acceso ya que permite ir directamente a la página asociada con cada entrada del índice. Para la indexación se usó el script indexación.sql que se encuentra en el directorio principal del proyecto. En los puntos 2.4.1, 2.4.2 y 2.4.3 se muestran los gráficos que resultan de las pruebas una indexado todas las tablas de la DB.

Gráfico de resultados de la consulta Simple:

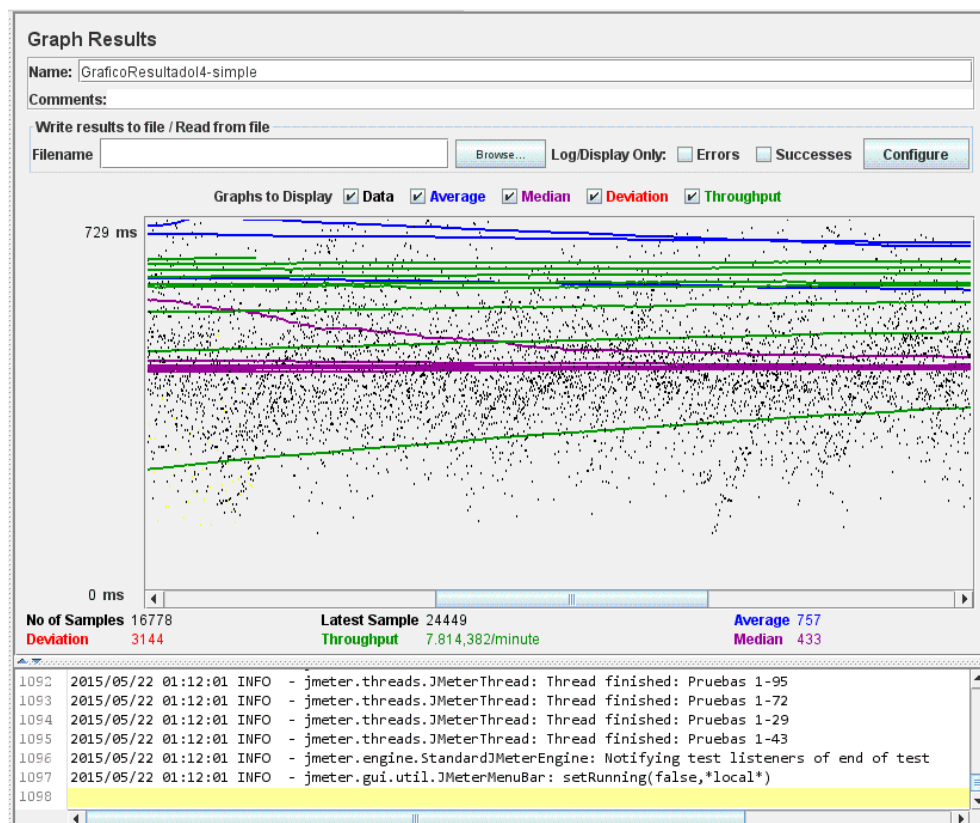


Gráfico de resultados de la consulta media:

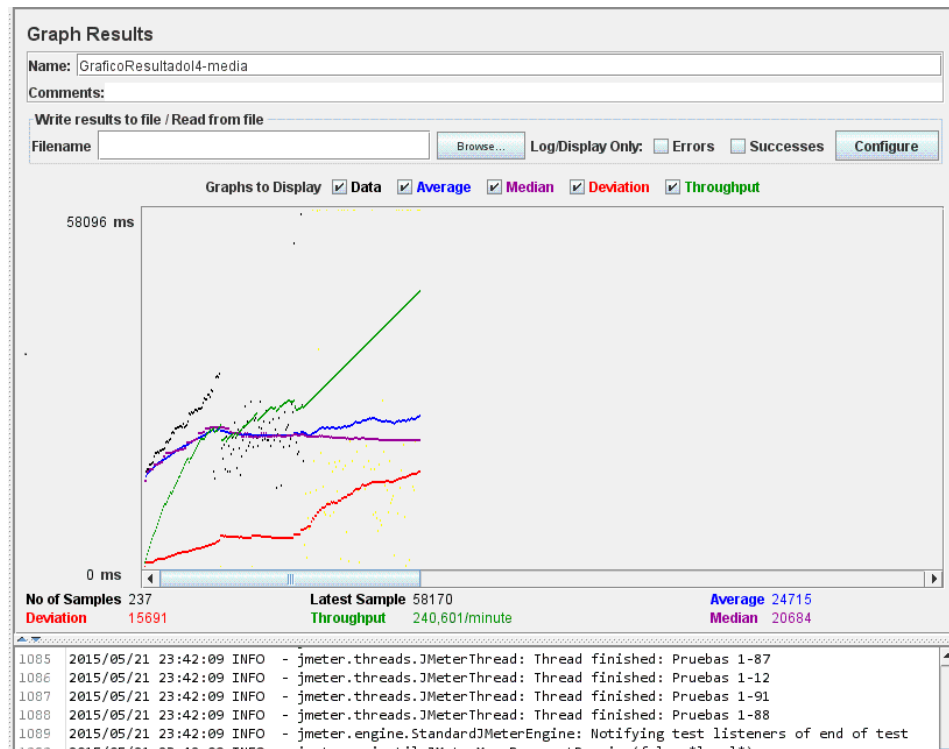
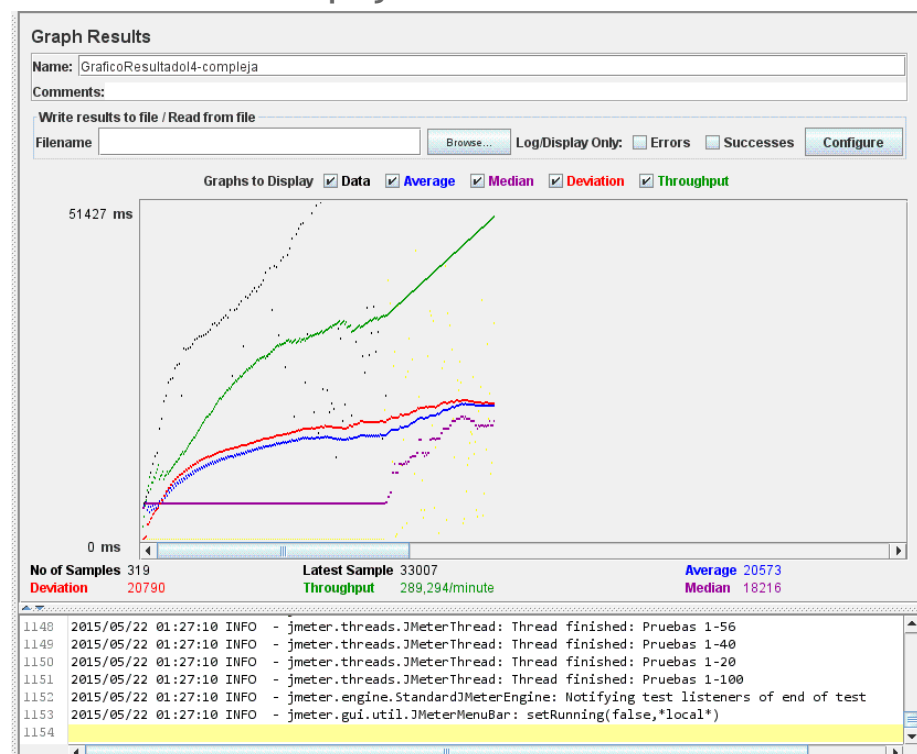


Gráfico de resultados de la consulta compleja:



5ta Iteración (Cambio de versión en la tecnología, y sistema operativo).

En esta iteración decidimos cambiar de sistema operativo a uno que tuviera menos procesos en segundo plano (en este caso ubuntu), además de ofrecernos un sistema más limpio, ligero y estable, de esta manera se puede obtener un mejor rendimiento en las consultas. Debido a que para montar el servidor utilizamos xampp, algunas incompatibilidades de codeigniter imposibilitaba su correcto funcionamiento, por lo cual lo actualizamos a la versión 3.

Gráfico de resultados de la consulta Simple:

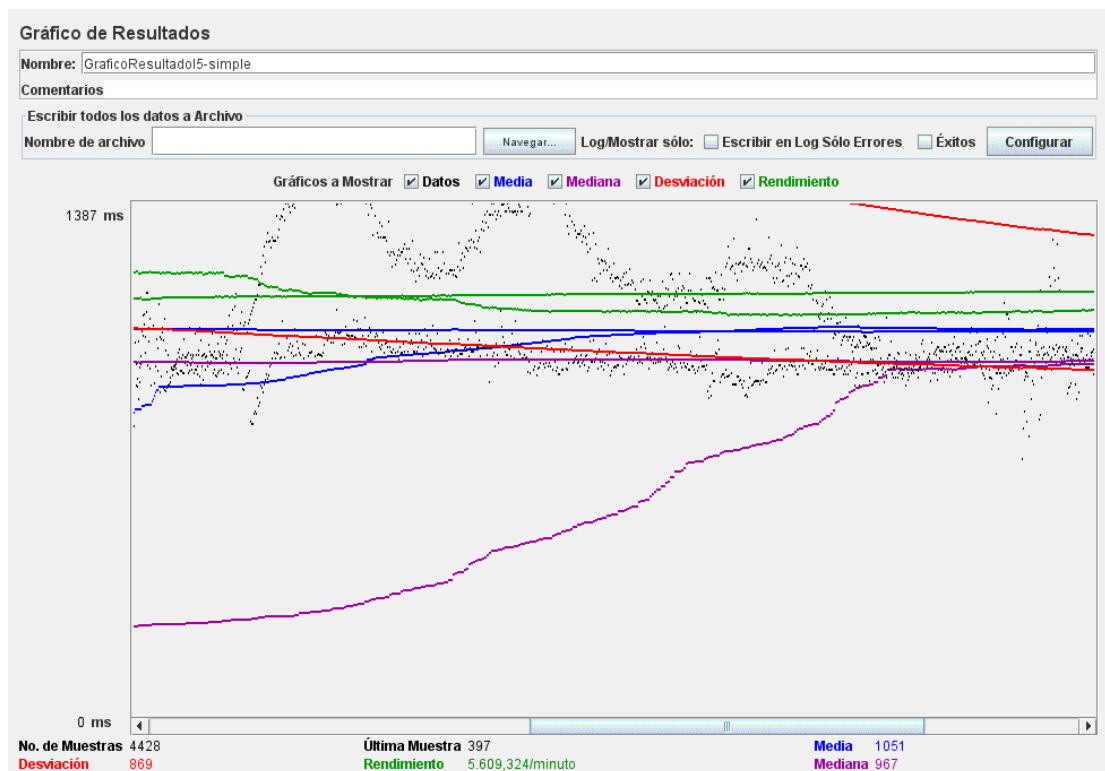


Gráfico de resultados de la consulta media:

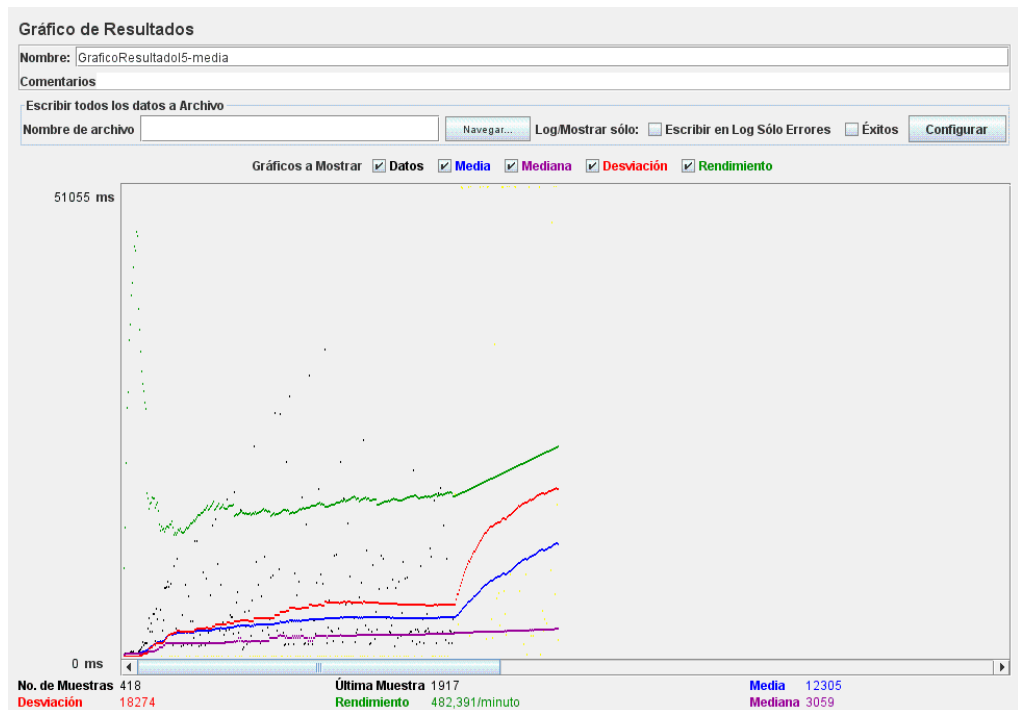


Gráfico de resultados de la consulta compleja:

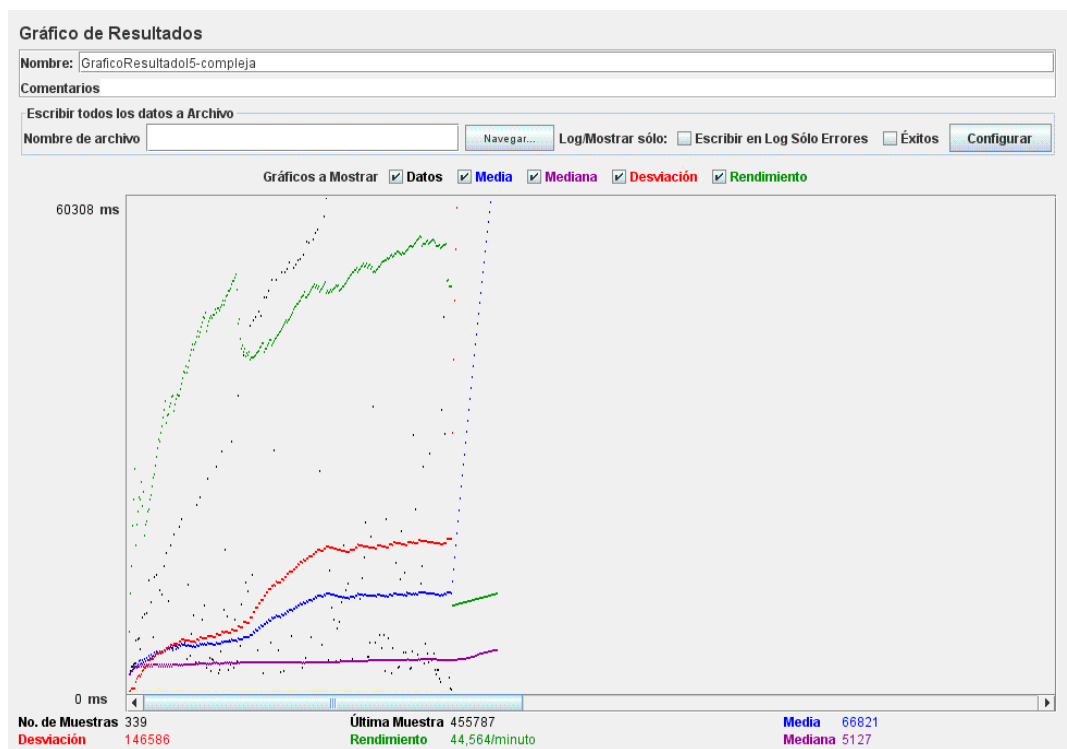


Tabla comparativa de las iteración

Consulta Simple

N° Iteración	N° Pruebas	Margen error	Throughput
1	4199	2.19%	52.3/s
2	3901	4.31%	40.7/s
3	8354	1.13%	131.6/s
4	16778	0.79%	130.2/s
5	4428	2.35%	93.5/s

Consulta Media

N° Iteración	N° Pruebas	Margen error	Throughput
1	101	100%	1.1/s
2	248	66.53%	10.8/s
3	391	61.13%	9.9/s
4	237	43.04%	4/s
5	418	42.58%	8.0/s

Consulta Compleja

N° Iteración	N° Pruebas	Margen error	Throughput
1	100	100%	1.1/s
2	285	67.37%	3.6/s
3	254	68.9%	5.0/s
4	319	64.89%	4.8/s
5	339	43.95%	44.6/m

Conclusión

Al finalizar las optimizaciones, se puede apreciar que si bien los primeros cambios no fueron significativos, ya que en las primeras optimizaciones la consulta media y compleja seguían mostrando un margen de error tendiente al 100%, sin embargo a medida que aumentaban las iteraciones se pudo lograr reducir este porcentaje. Como resultado de esta experiencia se pudo apreciar que para optimizar los tiempos de respuesta, no es necesario cambiar la tecnología utilizada, framework y versiones, ya que no muestran una mejora significativa. Por otro lado, la indexación de la BD y optimización de las consultas siguen mostrando aumentos importantes en lo que corresponde a rendimiento frente a estas.