

# CODEPROJECT

## Pruebas de carga y estrés - Nube.

A continuación, se realizarán las pruebas de carga y estrés al aplicativo para el control de inventarios realizado por CodeProject. Estas pruebas intentan medir cuán alto es el porcentaje de error que tiene el aplicativo al momento de recibir cierto número de peticiones en un tiempo de respuesta determinado.

La herramienta que se empleará para realizar las pruebas será **JMeter** y este examen se realizará al sitio web almacenado en la **Azure Cloud**.

### Primera prueba.

- Tomaremos como formulario de prueba el **dashboard de marcas** del aplicativo. La primera prueba consiste en trabajar con 6 grupos de hilos distintos. Cada uno de ellos tendrá un periodo de subida de 1 segundo.

Número de Hilos	Periodo de Subida (En segundos)	Porcentaje de Error	Rendimiento
1	1	0,00%	1,6/sec
10	1	0,00%	7,6/sec
100	1	0,00%	38,3/sec
1000	1	0,00%	76,4/sec
5000	1	54,58%	865,8/sec
10000	1	55,73%	62,3/sec

### Conclusión primera prueba.

De la primera prueba podemos inferir que a partir de los cinco mil usuarios, el aplicativo comienza a tener un alto porcentaje de error. Además, cada vez que aumenta el número de usuarios el aplicativo responde más lento y eventualmente ninguna petición podrá ser recibida. En base a esto, consideramos oportuno aumentar los segundos de subida a cada grupo de hilos a 3 segundos.

### Segunda prueba.

- De nuevo tomaremos el dashboard de ventas como objeto de prueba. En este segundo testeo seguiremos trabajando con los mismos 6 grupos de hilos, pero ahora el periodo de subida será de 3 segundos.

Número de Hilos	Periodo de Subida (En segundos)	Porcentaje de Error	Rendimiento
1	3	0,00%	1,5/sec
10	3	0,00%	3,2/sec
100	3	0,00%	29,3/sec
1000	3	0,00%	64,9/sec
5000	3	8,83%	41,1/sec
10000	3	57,61%	82,2/sec

### Conclusión segunda prueba.

En esta segunda prueba podemos inferir que el aumentar el periodo de subida a 3 segundos mejoró el rendimiento del aplicativo. El porcentaje de error disminuyó notablemente para el grupo de 5000 usuarios. Pasó de tener en la primera prueba un porcentaje de error del 54% a un casi que inmejorable 8,83%. Como medida ante esto, consideramos que el aumento de los segundos de subida ayudaría a que el aplicativo pueda recibir cada vez más peticiones.

---

### Tercera prueba.

- Para esta tercera prueba seguiremos haciendo uso del dashboard de ventas. En este tercer ensayo aumentaremos el periodo de subida a 5 segundos.

Número de Hilos	Periodo de Subida (En segundos)	Porcentaje de Error	Rendimiento
1	5	0,00%	1,6/sec
10	5	0,00%	2,0/sec
100	5	0,00%	18,6/sec
1000	5	0,00%	89,9/sec
5000	5	7,20%	18,2/sec
10000	5	54,61%	77,9/sec

### Conclusión tercera prueba.

Esta tercera prueba muestra un gran avance ya que el aplicativo puede recibir 5000 peticiones con un porcentaje de error de 7,20%. Además de esto, el rendimiento aumentó en muchos de los grupos. La meta ahora es hacer que a 5000 peticiones, el sistema tenga menos porcentaje de error, por lo cual aumentaremos el periodo de subida a 7 segundos.

### Cuarta prueba.

- Para esta prueba consideramos que aumentar el periodo de subida a 7 segundos mejorará el rendimiento del aplicativo. Seguiremos haciendo uso del dashboard de ventas.

Número de Hilos	Periodo de Subida (En segundos)	Porcentaje de Error	Rendimiento
1	7	0,00%	1,3/sec
10	7	0,00%	1,5/sec
100	7	0,00%	13,6/sec
1000	7	0,00%	92,8/sec
5000	7	17,82%	46,1/sec
10000	7	58,73%	62,0/sec

### Conclusión cuarta prueba.

La cuarta prueba refleja que a 10000 usuarios el aplicativo sigue comportándose de manera inestables, sin embargo, causó sorpresa que el porcentaje de error en 5000 aumentara aproximadamente un 10% con respecto a la prueba pasada. Consideramos que este cambio puede estar relacionado con la máquina en la que se están haciendo las pruebas, pero continuaremos con la quinta prueba.

### Quinta prueba.

- La quinta prueba consiste en aumentar el periodo de subida a 10 segundos. Utilizaremos el dashboard de ventas de nuevo para hacer esta prueba.

Número de Hilos	Periodo de Subida (En segundos)	Porcentaje de Error	Rendimiento
1	10	0,00%	1,6/sec

10	10	0,00%	1,1/sec
100	10	0,00%	9,7/sec
1000	10	0,00%	93,7/sec
5000	10	4,16%	41,8/sec
10000	10	76,85%	37,1/sec

### Conclusión quinta prueba.

En esta prueba hubo una disminución de errores considerable al momento de recibir 5000 peticiones. El aplicativo se mostró más estable y pasó a una tasa de error muy aceptable.

## Sexta prueba.

- La sexta y última prueba consiste en aumentar el periodo de subida de 10 segundos a 15. Esperamos que con este aumento en el periodo de subida el aplicativo tenga menos de 50% de errores a la hora de recibir peticiones.

Número de Hilos	Periodo de Subida (En segundos)	Porcentaje de Error	Rendimiento
1	15	0,00%	1,6/sec
10	15	0,00%	43,1/min
100	15	0,00%	6,5/sec
1000	15	0,00%	66,8/sec
5000	15	77,62%	22,3/sec
10000	15	45,18%	41,8/sec

## Conclusión final.

Consideramos que elaborar estas pruebas nos permitieron, de cierta manera, dimensionar la complejidad de hacer de un sistema una aplicación con un perfecto balance entre la rapidez y el rendimiento. En conclusión, debemos seguir aumentando el periodo de subida para que el aplicativo a las 5000 peticiones tenga una tasa de 0% de error.