Análisis de capacidad

Entrega 4 – Desarrollo de Software en la Nube

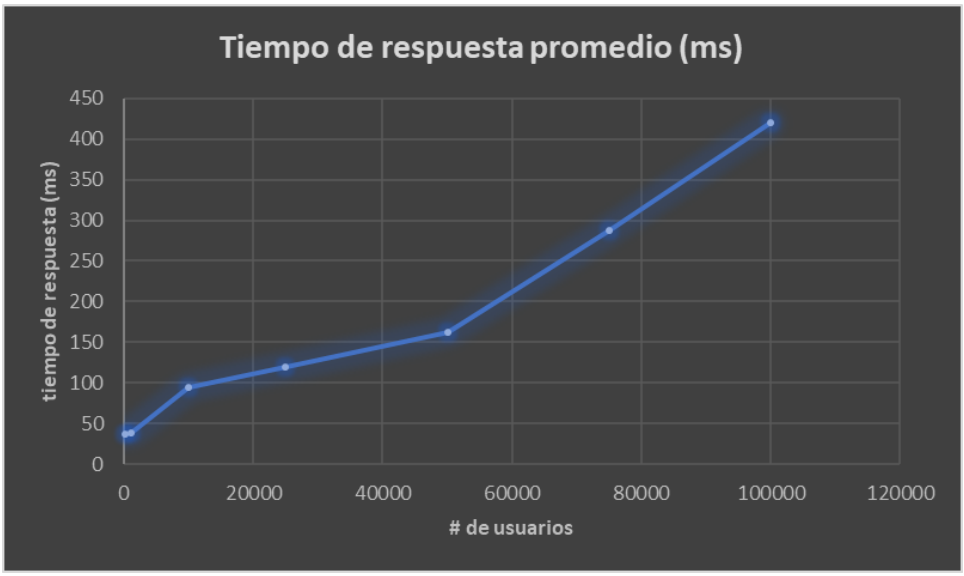
Diego Alejandro Camelo Giraldo, Juan Sebastián Alegría Zúñiga, María Camila Gómez Hernández, Andrés Felipe Lugo Saavedra

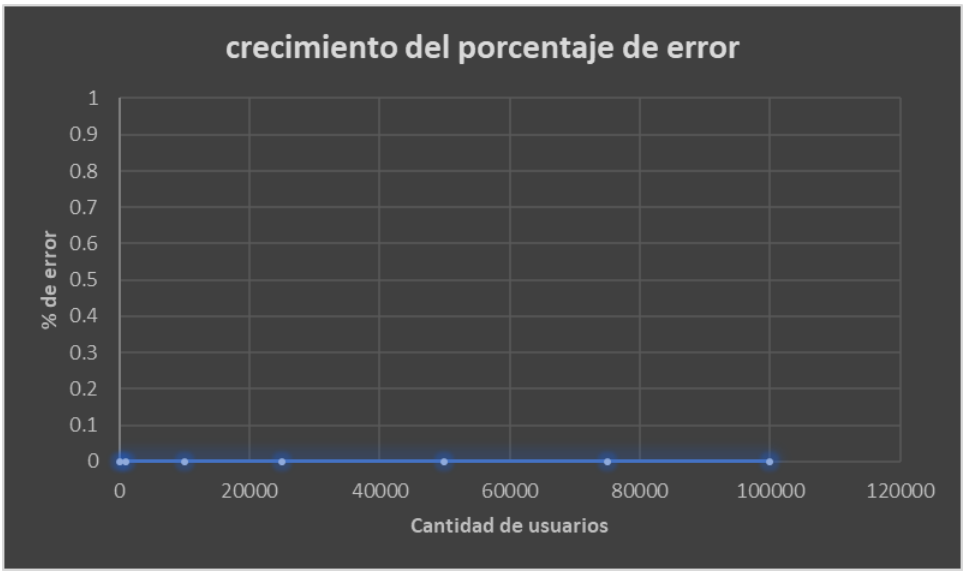
Grupo 6

Nota: Antes de ejecutar las pruebas de estrés, ajuste el documento del plan de análisis de capacidad con base a la retroalimentación realizada por los tutores del curso. 1. Pruebas de estrés sobre el Modelo de Despliegue.

Escenario 1. Se deberá definir un escenario donde se pueda probar cuál es la máxima cantidad de peticiones HTTP por minuto que soporta el API con 30 archivos publicados. Para hacer pruebas de estrés se sugiere utilizar la herramienta Apache Bench (ab) o JMeter que debe instalarse en el ambiente de nube (para no sesgar los resultados de la prueba con la latencia que introduce la red, la máquina de pruebas debe ubicarse en el mismo segmento de red de la aplicación). El escenario y los resultados de las pruebas de estrés deberán ser documentados con gráficas que ilustran cómo se comporta el sistema a medida que el número de usuarios virtuales accediendo a la aplicación se incrementa hasta llegar al punto de degradar completamente el rendimiento del sistema.

Restricciones del escenario. En las pruebas de stress el tiempo de respuesta promedio de la página debe ser de máximo 1.500 ms, si este tiempo no se cumple, se concluye que el sistema NO soporta la cantidad de peticiones de la prueba. En caso de que durante una prueba se generen más de un 1% de errores en las peticiones de la prueba, se concluye que el sistema NO soporta la cantidad de peticiones.

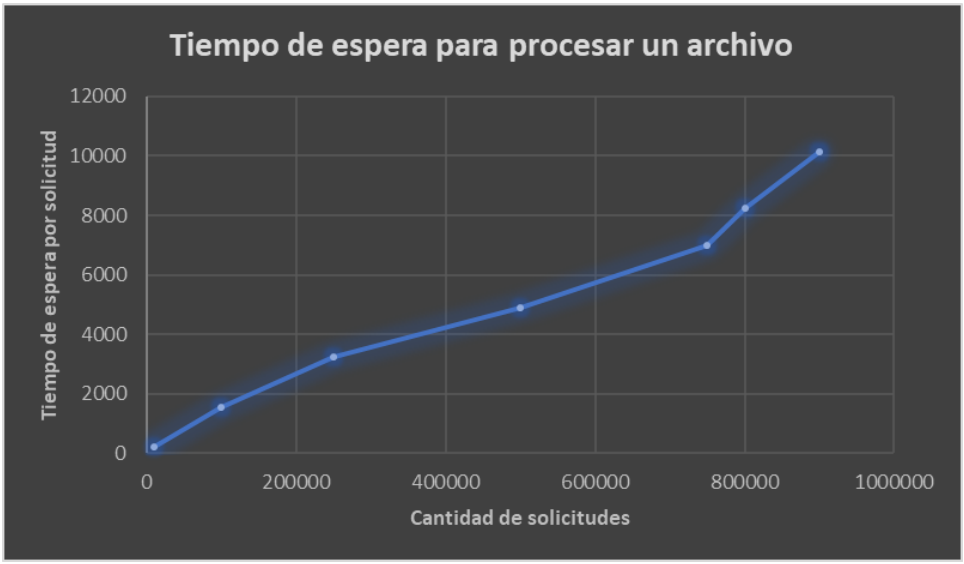




En este escenario, llevamos a cabo una prueba de carga en la que fuimos aumentando gradualmente el número de usuarios que enviaban solicitudes, mientras registramos los tiempos de respuesta esperados y el porcentaje de error en cada solicitud. Durante toda la prueba, vimos que el comportamiento de la prueba se mantenía muy estable conforme al aumento de solicitudes, dando como resultado un aumento lineal en el tiempo de respuesta por solicitud, teniendo un pequeño cuello de botella entre las 10000 y 50000 solicitudes. Comparando estos resultados con la entrega anterior, vemos una mejoría en los tiempos de respuesta y un porcentaje del 0% de errores incluso al llegar a las 100000 solicitudes, lo cual muestra que las mejoras recientes fueron efectivas. Al tratar de probar más allá de este

número de solicitudes, el programa en la máquina virtual empezó a congelarse por lo que fue imposible seguir con las pruebas, sin embargo, consideramos que los resultados son más que satisfactorios para probar el rendimiento actual del programa.

Escenario 2. Se deberá definir un escenario donde se pueda probar cuál es la máxima cantidad de archivos que pueden ser procesados por minuto en la aplicación. Para hacer pruebas de estrés se debe utilizar la herramienta Apache Bench (ab) o JMeter que debe instalarse en el ambiente de nube (para no sesgar los resultados de la prueba con la latencia que introduce la red, la máquina de pruebas debe ubicarse en el mismo segmento de red de la aplicación). Las pruebas de estrés deberán realizarse desde otros equipos diferentes a los utilizados para ejecutar el servidor web y el servidor de base de datos. El escenario y los resultados de las pruebas de estrés deberán ser documentados con gráficas que ilustran cómo se comporta el sistema a medida que el número de usuarios convirtiendo archivos se incrementa, hasta llegar al punto en que el tiempo para iniciar el procesamiento de un archivo enviado por un usuario supere los 10 minutos (600 segundos).



En este escenario podemos ver como el cambio en el tiempo de espera entre solicitudes se mantiene con un crecimiento lineal bastante estable. Antes de las 700000 solicitudes el tiempo de espera sigue la tendencia y no parece presentar anormalidades mientras aumentan las solicitudes. No obstante, hay un crecimiento más prolongado luego de las 750000 solicitudes, por lo que se puede entender que ahí se encuentra el cuello de botella del servicio. Si comparamos esto con el resultado anterior, vemos que el umbral antes del cuello de botella creció considerablemente, por lo que se puede verificar que las mejoras lograron hacer el servicio más eficaz. Se tiene que tener en cuenta 2 factores importantes para entender mejor los resultados conseguidos. El primero es que la falta de recursos en la máquina virtual pudo tener que ver con el comportamiento de las pruebas, pues a veces jmeter puede dar ciertos fallos. Lo segundo es que no se superó el 1% de errores en ninguno de los casos anteriores, mostrando que todas las solicitudes se atendían sin importar que tanto se esperaban.

Cuellos de botella:

Al probar los servicios de la aplicación con la herramienta jmeter, logramos descubrir la cantidad de solicitudes que estas son capaces de soportar antes de empezar a presentar fallas y que el rendimiento de la aplicación comience a fallar.

Servicio de Sign-Up: Soporta poco más de 100000 solicitudes en un minuto, con un rendimiento aproximado de 420 ms por solicitud. El cuello de botella no es muy claro en este servicio, no obstante, se puede suponer que luego de las 2000000 empiezan a existir problemas para atender y aparecen porcentajes de error superiores al 0.1%.

Servicio de Login: Soporta un aproximado de 95000 solicitudes en un minuto, con un rendimiento aproximado de 647 ms por solicitud. El cuello de botella comienza cerca de las 100000 solicitudes subiendo los tiempos de respuesta a 1079 ms y creciendo los porcentajes de error a 0,07 %

Servicio de carga de archivos: Soporta un aproximado de 700000 solicitudes en un minuto, con un rendimiento máximo de 7002 ms por solicitud (este fue el tiempo de espera más largo para esta cantidad de solicitudes). El cuello de botella en esta situación comienza cuando se sobrepasa el numero anterior, no obstante, el auto escalado permite que se regule y el crecimiento se mantenga lineal a pesar de las exigencias. Porcentaje de errores entre 0.31% y el 0.97%

Servicio de acceso a archivos: Soporta un aproximado de 100000 solicitudes en un minuto, con un rendimiento aproximado de 478 ms por solicitud. El cuello de botella comienza cerca de las 220000 solicitudes subiendo los tiempos de respuesta a 1096 ms y creciendo los porcentajes de error a 0,67 %

Conclusiones:

Estas pruebas de carga muestran que la aplicación está en capacidad de recibir cientos de miles de solicitudes sin sufrir en el proceso de atenderlas constantemente. Adicionalmente, los porcentajes de solicitudes con errores se redujeron drásticamente. Por lo que aumentar la disponibilidad por medio del escalado automático fue una integración crucial al proyecto.