

Análisis de Capacidad. Entrega No. 3

Desarrollo de Soluciones Cloud

Grupo 5: Jairo Céspedes Plata, María Catalina Ibáñez, José Manuel Moreno, David Saavedra Martínez

Contexto del proyecto

El proyecto sobre el cual se adelantarán los análisis de capacidad y pruebas, descritos a continuación tiene como principal propósito la conversión de archivos de distintos formatos mediante un servicio web. En consecuencia, resulta necesaria la ejecución de un análisis de capacidad. El siguiente análisis de capacidad hace parte de la tercera entrega de documentación acerca de la herramienta desarrollada hasta la fecha.

Herramientas para el análisis:

Para llevar a cabo el análisis de capacidad exhaustivo se utilizó una máquina local en Bogotá, Colombia que ejecute la herramienta de carga JMeter.

La selección de la máquina y la herramienta se dio debido a la facilidad que ofrece la interfaz de JMeter en Windows y la conexión por cable ethernet (300Mb con servicio de fibra óptica) que posee la máquina local; las especificaciones se ilustran en la Imagen 1. Por otra parte, vale mencionar que se ha decidido utilizar la herramienta JMeter, ya que es un framework ampliamente conocido, así como por la gran cantidad de documentación disponible.



Imagen 1. Capacidad de la máquina local.

La configuración de los parámetros de JMeter, consideró el ajuste del número de hilos, el tiempo de rampa y otras variables relevantes para simular cargas realistas en el sistema. Adicionalmente, se diseñaron escenarios de prueba detallados y realistas para abordar diferentes aspectos críticos del sistema.

Criterios de medición de desempeño

Como marco general para la construcción de las pruebas se tuvo la metodología GSM (Goal, Signal, Metric) desarrollada por Google [1]. Dicho enfoque permite construir mediciones basadas en las necesidades del proyecto que podrán variar a lo largo del tiempo y no realizar el proceso inverso de fijar métricas arbitrarias para luego ajustar las pruebas.

Como principales criterios están la carga y la conversión explicados a detalle en la tabla 1.

Tabla 1. Criterios para la construcción de pruebas

	GOAL	SIGNAL	METRIC
--	------	--------	--------

Carga	Usuarios podrán acceder a la carga de archivos de manera eficiente.	Cuando un usuario intenta cargar un archivo, la plataforma está disponible y funciona de manera rápida.	Respuesta de las peticiones en menos de 25 seg (Throughput) y nivel del error del 0.1%.
Conversión	Los documentos convertidos deberán ser correctos y procesados de manera rápida.	Cuando un usuario realiza una conversión recibe el documento correcto y de manera rápida.	99.9% de los archivos deben ser correctos y procesar al menos 0.5 archivos por min.

Tabla 1. Fijación de métricas estimadas.

Adicionalmente y buscando brindar mayor robustez se presentan criterios de aceptación para ejecutar los escenarios de carga con características y técnicas mejor definidas.

Escenarios de prueba:

1. Carga de Archivos:

- Se simulará la carga de archivos en el sistema mediante solicitudes HTTP utilizando datos de prueba que reflejen la diversidad de archivos que podrían ser cargados por los usuarios en la aplicación (archivos con extensión .docx, .xlsx y .pptx). Se evaluará la capacidad del sistema para manejar archivos de 1 MB, así como la concurrencia de varios archivos cargándose simultáneamente (con 0 seg de diferencia entre cada uno).
- Estos *requests* deberían responderse en promedio, en menos de 25 segundos. En el caso que el tiempo establecido sea superado, se considerará que el sistema perdió rendimiento y la aplicación llegó a su límite. La cantidad máxima de archivos será aquella en la que no degrade ese tiempo de respuesta. Por lo tanto, se espera que, en promedio, se logre subir 1 archivo por segundo, aunque esto también podrá depender, para efectos de la prueba. Además, se espera que la carga de archivos no supere una utilización de RAM de alrededor del 50% en el file-server; esto dado que se puede cargar un archivo, mientras se está convirtiendo otro a PDF, por lo que se necesita capacidad disponible para esta operación.
- Finalmente, se espera que el 99.9% (% de error del 0.1%) de los archivos logren ser cargados exitosamente
- **Parámetros de Jmeter:** La configuración final para las pruebas de carga de archivos implicará realizar ajustes en la cantidad de hilos (threads) para encontrar un equilibrio entre el rendimiento y la tasa de error. Se realizarán varias pruebas con diferentes cantidades de hilos (200, 190, 180, 170, 160), hasta que arrojen una tasa de error aceptable (del orden del 0.1%).

2. Procesamiento de Archivos a PDF y métricas de desempeño:

- Se simulará el procesamiento de archivos para convertirlos a formato PDF, una operación crítica en el flujo de trabajo de la aplicación. Se medirá el tiempo necesario para procesar cada archivo y se evaluará la capacidad del sistema para manejar cargas de trabajo intensivas

considerando factores como el tamaño y la complejidad de los archivos, así como la disponibilidad de recursos del sistema.

- Se estimará la cantidad de archivos con extensión .docx, .pptx y .xlsx que fue capaz de convertir a pdf, después de que se lancen las tareas en 1 minuto, mediante una cola de mensajes, cada documento cargado debe pesar 1 Mb.
- Se espera que la tasa de error sea menor al 0.1% y que se logre procesar 0.5 archivos por minuto. Si alguno de estos SLOs es superado se considerará que el sistema perdió rendimiento y la aplicación llegó a su límite
- **Parámetros de Jmeter:** La configuración final para las pruebas de carga de archivos implicará realizar ajustes en la cantidad de hilos (threads) para encontrar un equilibrio entre el rendimiento y la tasa de error. Se realizarán varias pruebas con diferentes cantidades de hilos (50, 10, 5), hasta que arrojen una tasa de error aceptable (0%) y una tasa de 0.5 archivos por minuto, de forma ideal.
- Cada prueba se va a ejecutar al menos 3 veces para evitar el sesgo estadístico y se anotarán los resultados de cada una, finalmente sacar conclusiones más específicas como:
 - Cantidad de archivos subidos al API sin afectar el tiempo de respuesta de menos de 5 segundos.
 - Cuántos archivos es capaz de convertir el sistema en menos de 1 minuto siendo de diferentes tipos (.docx, .pptx y .xlsx) y de un tamaño de 1Mb.
 - Comportamiento de las métricas en un escenario de estrés (utilización CPU y RAM, capacidad de procesamiento y tiempo de respuesta).

3. *Desarrollo de las pruebas:*

3.1. Prueba #1 - Carga de Archivos

La configuración final para las pruebas de carga de archivos implicó realizar ajustes en la cantidad de hilos (threads) para encontrar un equilibrio entre el rendimiento y la tasa de error. Al realizar varias pruebas con diferentes cantidades de hilos (200, 190, 180, 170, 160), se determinó que utilizar unos 160 hilos para cada tipo de archivo, arrojaba la menor tasa de error para ese conjunto de tamaños de muestras, dicha tasa fue del orden del 0.21%. Sin embargo, vale resaltar que no se logró una tasa de error inferior al límite preestablecido del 0.1%.

Los resultados promedio de estas pruebas indicaron un rendimiento con un promedio throughput de 20.46/secs, con lo cual como se advirtió, **no se logró el objetivo de tener una tasa de error inferior al 0.1%**. Sin embargo, como se explica en el desarrollo de las pruebas, este resultado se tendría en parte o podría atribuirse a una aparente limitación de GCP, donde ésta última, tendría una limitación

para más de 150 hilos, no porque la máquina virtual tuviera algún problema de procesamiento o capacidad, si no por alguna condición de no permitir subir más de 150 hilos.

A continuación, se detalla el paso a paso realizado para lograr estos resultados:

- Pruebas utilizando 200 hilos (threads) por cada tipo, para 3 tipos diferentes de archivos (docx, xlsx, pptx); sin embargo, el % de error subió del 0.1% (15.5%) así que se descarta, debido a que el mismo Google Cloud Storage niega tantas peticiones al mismo tiempo.

Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: ☐ Errors ☐ Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Carga de Archi...	200	5779	368	14235	3533.09	0.00%	14.0/sec	2.48	14273.29	181.5
Carga de Archi...	200	7735	171	14172	3723.68	1.00%	9.2/sec	1.64	9913.29	182.6
Carga de Archi...	200	3665	166	12890	3986.48	45.50%	9.5/sec	2.12	10337.51	227.4
TOTAL	600	5726	166	14235	4103.92	15.50%	26.5/sec	5.10	28101.32	197.1

Text

☒ Carga de Archivos xlsx
☒ Carga de Archivos xlsx
☒ Carga de Archivos xlsx
☒ Carga de Archivos xlsx
☒ Carga de Archivos pptx
☒ Carga de Archivos xlsx
☒ Carga de Archivos pptx
☒ Carga de Archivos pptx
☒ Carga de Archivos xlsx
☒ Carga de Archivos pptx
☒ Carga de Archivos pptx

Sampler result Request Response data

Response Body Response headers

Find Case

```
<!doctype html><meta charset="utf-8"><meta name=viewport content="width=device-width, initial-scale=1"><title>403</title>403 Forbidden
```

- Pruebas utilizando 190 hilos (threads) por cada tipo, para 3 tipos diferentes de archivos; sin embargo, el error es superior al 0.1%, se obtuvo el 10.88% así que se descarta, debido a que el mismo Google Cloud Storage niega tantas peticiones al mismo tiempo.

Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: ☐ Errors ☐ Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Carga de Archivos docx	700	20625	693	45668	9142.89	0.00%	15.3/sec	2.34	15573.03	157.0
Carga de Archivos xlsx	700	37830	162	92891	15465.76	0.57%	6.5/sec	1.09	6930.55	172.7
Carga de Archivos pptx	700	33384	155	77967	12368.35	0.00%	6.2/sec	0.95	6715.52	157.0
TOTAL	2100	31280	155	92891	14708.67	0.19%	10.5/sec	2.93	19545.99	162.2

- Pruebas utilizando 180 hilos (threads) por cada tipo, para 3 tipos diferentes de archivos; sin embargo, el error es superior al 0.1%, se obtuvo un % de error que subió del 0.10% así que se descarta (6.48%), debido a que el mismo Google Cloud Storage niega tantas peticiones al mismo tiempo.

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

☐ Successes

Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Carga de Archi...	180	5184	470	12064	2929.29	0.00%	14.9/sec	2.64	15193.38	181.4
Carga de Archi...	180	8068	827	14081	3670.20	0.00%	7.8/sec	1.39	8454.23	181.4
Carga de Archi...	180	4685	170	13850	3656.06	19.44%	7.4/sec	1.46	8044.11	201.1
TOTAL	540	5979	170	14081	3745.71	6.48%	21.2/sec	3.88	22433.62	188.0

- Pruebas utilizando 170 hilos (threads) por cada tipo, para 3 tipos diferentes de archivos; sin embargo, el error es superior al 0.1%, se obtuvo como % de error que subió del 0.10% así que se descarta (2.16 %), debido a que el mismo Google Cloud Storage niega tantas peticiones al mismo tiempo.

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only: ☐ Errors ☐ Successes

Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Carga de Archi...	170	5204	499	12316	3048.17	0.00%	13.7/sec	2.43	14018.65	181.4
Carga de Archi...	170	7821	674	14588	3848.32	0.00%	7.5/sec	1.33	8087.73	181.4
Carga de Archi...	170	5163	169	12646	3590.44	6.47%	6.7/sec	1.23	7297.50	187.5
TOTAL	510	6063	169	14588	3725.23	2.16%	19.3/sec	3.45	20416.25	183.4

- Finalmente se realizaron pruebas utilizando 160 hilos (threads) por cada tipo: Se hizo para 3 tipos diferentes de archivos. Al ejecutar 3 iteraciones, el error se mantiene por debajo del 0.1%. Al finalizar, se registró un error % promedio del 0.21% y un promedio de throughput= 20.46/seg

- Iteración 1:

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

☐ Successes

Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Carga de Archi...	160	4578	222	10310	2410.89	0.00%	15.4/sec	2.73	15745.24	181.3
Carga de Archi...	160	6866	1135	11510	2490.93	0.00%	8.3/sec	1.48	8997.32	181.3
Carga de Archi...	160	5308	946	10188	2578.30	0.00%	8.1/sec	1.44	8819.94	181.3
TOTAL	480	5584	222	11510	2670.66	0.00%	21.4/sec	3.79	22687.49	181.3

- Iteración 2:

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

☐ Successes

Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Carga de Archi...	160	4814	353	12105	2966.00	0.00%	13.2/sec	2.34	13462.88	181.3
Carga de Archi...	160	7120	100	13231	3276.35	0.62%	7.4/sec	1.31	7966.88	181.8
Carga de Archi...	160	4842	224	10927	2693.32	0.00%	7.3/sec	1.29	7913.37	181.3
TOTAL	480	5592	100	13231	3177.48	0.21%	20.2/sec	3.58	21402.48	181.5

- Iteración 3:

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display O

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec
Carga de Archivos d...	160	4939	415	12358	3210.19	0.00%	12.9/sec	2.30
Carga de Archivos.xlsx	160	6563	815	13324	3532.31	0.00%	7.6/sec	1.34
Carga de Archivos p...	160	4721	92	12296	3281.67	1.25%	7.0/sec	1.25
TOTAL	480	5408	92	13324	3443.70	0.42%	19.8/sec	3.52

En este punto, la configuración asociada a las pruebas considerando 160 hilos fue:

Thread Group

Name: Archivos

Comments:

Action to be taken after a Sampler error

☒ Continue
 ☐ Start Next Thread Loop
 ☐ Stop Thread
 ☐ Stop Test
 ☐ Stop Test Now

Thread Properties

Number of Threads (users): 160

Ramp-up period (seconds): 0

Loop Count: ☐ Infinite ☒ 1

☒ Same user on each iteration
☐ Delay Thread creation until needed
☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

HTTP Request

Name: Carga de Archivos.docx

Comments:

Basic: Advanced

Web Server

Protocol (http): Server Name or IP: 34.110.178.166 Port Number:

HTTP Request

POST Path: /files/uploadfile Content encoding:

☐ Redirect Automatically
 ☒ Follow Redirects
 ☒ Use Keepalive
 ☒ Use multipart/form-data
 ☐ Browser-compatible headers

Parameters Body Data Files Upload

File Path	Parameter Name	MIME Type
C:/Users/genia/OneDrive/Desktop/Trabajos 2024-1/Cloud/archivos/prueba1docx_S(loop_counter).docx	file	application/vnd.openxmlformats-officedoc...

Vale mencionar que la VM-web server, no excedió el 50% de uso de CPU y se comportó de la siguiente manera:



3.2. Prueba #2 - Procesamiento de Archivos en la conversión a PDF y métricas de desempeño

Para las pruebas en el procesamiento de archivos durante la conversión, la configuración final también requirió ajustes en la cantidad de hilos a considerar. En primera instancia se evaluaron 50 hilos, obteniendo como resultado una tasa de error superior al 98% (solo 3 tareas se transformaron de 150), esto debido a que al parecer el worker no deja crear más tareas de procesamiento y solo realiza esas 3 y para el resto simplemente no les realiza procesamiento.

Ante este resultado, se consideró un número inferior de hilos para mejorar el desempeño en la prueba. Sin embargo, al considerar 10 hilos por tipo de archivo, la herramienta lograba una transformación de 9 archivos de 141, con una tasa de error del 94%. Lo cual sigue siendo una tasa muy alta de error.

Se volvió a ajustar el número de hilos a 5, donde al cabo de 1 minuto se presentaron 0 trabajos finalizados – transformados.

A continuación, se detalla el paso a paso realizado para lograr estos resultados:

- Pruebas utilizando 50 hilos (threads) por cada tipo, para 3 tipos diferentes de archivos; sin embargo, pasado 1 minuto, se transformaron muy pocas tareas.

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

☐ Successes

Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
CrearTarea docx	50	389	117	613	151.84	0.00%	79.9/sec	37.24	50.20	477.5
CrearTarea pptx	50	359	143	563	124.42	0.00%	49.8/sec	23.22	31.30	477.5
CrearTarea xlsx	50	267	86	496	103.72	0.00%	46.1/sec	21.49	28.97	477.5
TOTAL	150	338	86	613	138.39	0.00%	110.4/sec	51.46	69.38	477.5

- Al cabo de 1 minuto se arroja un XML el cual muestra que, de las 150 tareas, unas 3 se transformaron en ese periodo de tiempo:

```

<?xml version="1.0" encoding="UTF-8"?>
<testResults version="1.2">
  <httpSample>
    <responseData class="java.lang.String">{"input_file_path":"uploads/prueba1docx_3.docx";"converted_file_ext":"pdf";"name":&#
  </httpSample>
</testResults>
  
```

- Mientras que 147 siguen en Uploaded.

```

<?xml version="1.0" encoding="UTF-8"?>
<testResults version="1.2">
  <httpSample>
    <responseData class="java.lang.String">{"input_file_path":"uploads/prueba1docx_3.docx&
  </httpSample>
</testResults>
  
```

- Se observa que el worker escaló:



- Se procede a bajar el número de hilos a 0 threads por cada tipo de archivo. Dado que se obtuvo que $3/150 = 0.02$ tareas procesadas por minuto y el objetivo es lograr que se tenga: 0.5 tareas por minuto. Aunque, en este caso, al esperar 10 minutos, se obtuvo que 9 fueron exitosos y 141 fallaron, por lo tanto, hay un % de error del 94%, un valor bastante alto. Vale resaltar que lo que sí se cumple es que de esos 3, al obtener el estado processed y el resto ser uploaded implica que el 100% de los archivos que se transformaron en ese tiempo lo hicieron exitosamente.

- Pruebas con 10 threads por cada tipo de archivo:

Summary Report

Name:

Comments:

Write results to file / Read from file

Filename: Log/Display Only: ☐ Errors ☐ Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
CrearTarea docx	10	117	102	129	8.82	0.00%	76.3/sec	35.43	47.87	475.3
CrearTarea pptx	10	132	85	279	71.55	0.00%	33.3/sec	15.47	20.90	475.3
CrearTarea.xlsx	10	109	81	266	53.02	0.00%	33.4/sec	15.52	20.97	475.3
TOTAL	30	119	81	279	52.52	0.00%	61.7/sec	28.65	38.71	475.3

Y al cabo de 1 minuto arroja un XML el cuál muestra que de las 30 tareas, 0 se transformaron en ese periodo de tiempo:

```

Welcome  import shutil  Untitled-1  Entrega3Prueba1Tree.csv
C: > Users > genia > OneDrive > Desktop > Trabajos 2024-1 > Cloud > Entrega3Prueba1Tree.csv
> prod  Aa .ab .* No results  ↑ ↓ ≡ ×

1  <?xml version="1.0" encoding="UTF-8"?>
2  <testResults version="1.2">
3  <httpSample>
4  |  <responseData class="java.lang.String">[{"input_file_path":"uploads/prueba1docx_3.docx&
5  </httpSample>
6
7  </testResults>
8
  
```

- 30 siguen en Uploaded implicando que ninguno ha sido error.

El worker quedó así, donde se nota que no existió algún problema de capacidad o procesamiento con la Máquina Virtual:



- Al final, se obtuvo una tasa de $0/30 = 0$ archivos por minuto, por lo tanto, se evaluará el reducir la cantidad de tareas ya que el worker se está viendo saturado.
- Pruebas con 5 threads, para los 3 tipos de archivos:

- Iteración 1:

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

☐ Successes

Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
CrearTarea docx	5	135	88	269	67.41	0.00%	18.5/sec	8.56	11.57	475.0
CrearTarea pptx	5	87	79	94	6.53	0.00%	19.2/sec	8.92	12.06	475.0
CrearTarea.xlsx	5	88	76	102	9.68	0.00%	19.2/sec	8.89	12.01	475.0
TOTAL	15	103	76	269	45.27	0.00%	34.7/sec	16.11	21.77	475.0

- Y al cabo de 1 minuto arroja un XML el cual muestra que, de las 15 tareas, 0 se transformaron en ese periodo de tiempo:

```

Welcome  x  import shutil  Untitled-1  x  Entrega3Prueba1Tree.csv  x
C: > Users > genia > OneDrive > Desktop > Trabajos 2024-1 > Cloud > Entrega3Prueba1Tree.csv
> uploaded  Aa ab * 1 of 15  ↑ ↓ ≡ ×

1  <?xml version="1.0" encoding="UTF-8"?>
2  <testResults version="1.2">
3  <httpSample>
4  |   <responseData class="java.lang.String">[{"input_file_path":"uploads/prueba1docx_5.docx&
5  </httpSample>
6
7  </testResults>
8
  
```

Resultado Iteración 1: Tasa de 0 archivos transformados por minuto.

El Worker registró lo siguiente:





Donde se obtuvo que el % de error es del 100% ya que los 15 archivos fueron no procesados (al cabo de 10 min).

- Iteración 2:

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

☐ Successes

Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
CrearTarea docx	5	102	93	113	7.60	0.00%	43.5/sec	20.30	27.34	478.0
CrearTarea pptx	5	91	85	100	5.24	0.00%	44.2/sec	20.65	27.83	478.0
CrearTarea.xlsx	5	88	84	96	3.97	0.00%	41.0/sec	19.13	25.77	478.0
TOTAL	15	94	84	113	8.40	0.00%	49.5/sec	23.11	31.13	478.0

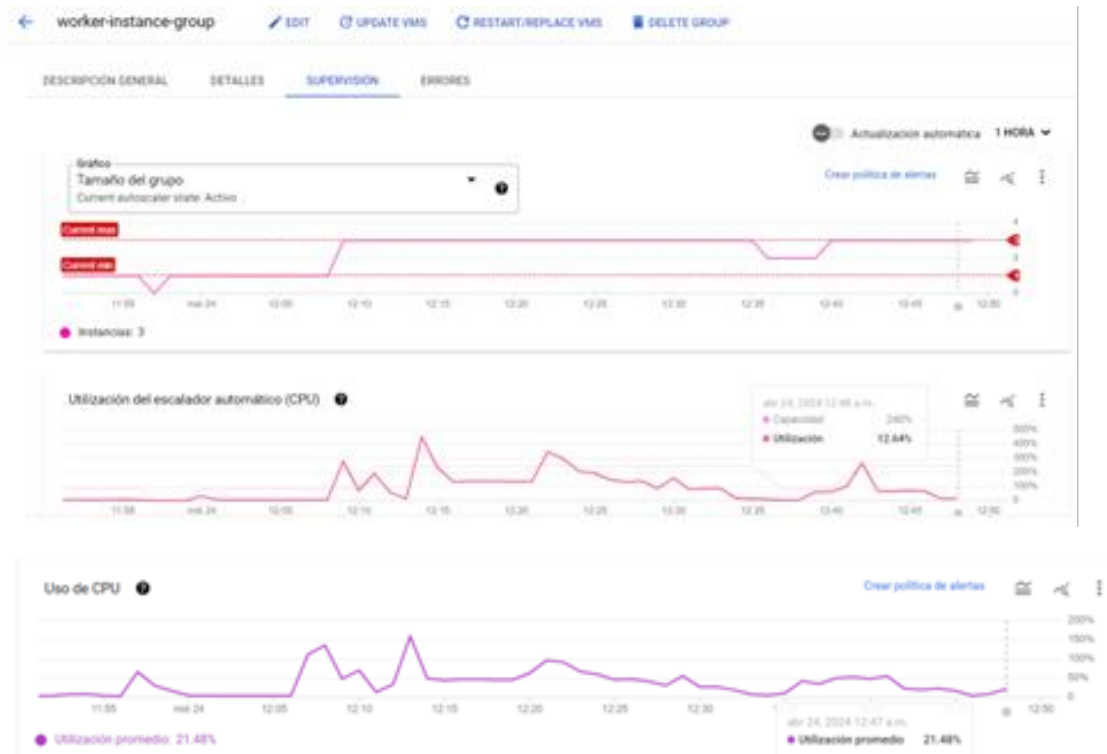
- Y al cabo de 1 minuto arroja un XML que muestra que, de las 15 tareas, 0 se transformaron en ese periodo de tiempo (hay 30 porque están las 15 anteriores):

```

Welcome  x  import shutil  Untitled-1  ●  Entrega3Prueba1Tree.csv  x
C: > Users > genia > OneDrive > Desktop > Trabajos 2024-1 > Cloud > Entrega3Prueba1Tree.csv
1  <?xml version="1.0" encoding="UTF-8"?>
2  <testResults version="1.2">
3  <httpSample>
4  | <responseData class="java.lang.String">[{"input_file_path":"uploads/prueba1docx_5.docx&
5  </httpSample>
6
7  </testResults>
8
  
```

Resultado Iteración 2: Tasa de 0 archivos transformados por minuto.

El Worker registró:



Donde se obtuvo un % de error del 100% ya que los 15 archivos no fueron procesados (al cabo de 10 min)

RESULTADO FINAL: No se realiza otra prueba debido a que el error ya es del 100% en todos los casos

Datos finales con 5 threads por cada tipo de archivo:

Promedio de archivos por minuto de las 2 pruebas= 0

Promedio de % de error al minuto= 100%

Promedio de % de error a los 10 min= 100%

Como caso extra se realizó una modificación para ver si se solucionaba el problema del worker, se empezó con 2 workers iniciales escalables a 3 para ver si el procesamiento se corregía, con un máximo de CPU del 55%:

- Pruebas con 6 threads, para 1 tipo de archivo “*.docx”, el cual tiene un tamaño de 433kb. Sin embargo, de las 6 tareas a procesar en 1 minuto, solo procesó 1.

Summary Report

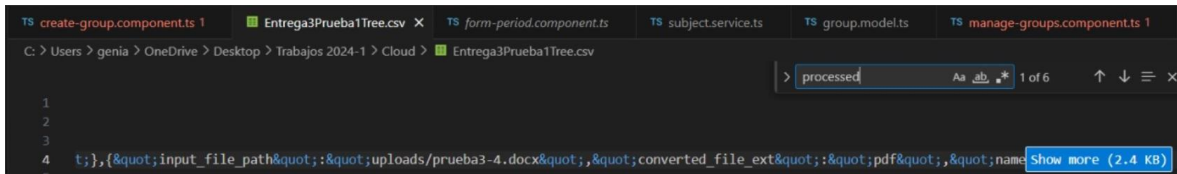
Name:

Comments:

Write results to file / Read from file

Filename: Log/Display Only: ☐ Errors ☐ Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
CrearTarea.docx	6	111	89	137	20.64	0.00%	43.8/sec	19.97	27.29	467.0
TOTAL	6	111	89	137	20.64	0.00%	43.8/sec	19.97	27.29	467.0



```

C:\Users\genia>OneDrive>Desktop>Trabajos 2024-1>Cloud>Entrega3Prueba1Tree.csv
> processed 1 of 6
t;},{"input_file_path":"uploads/prueba3-4.docx","converted_file_ext":"pdf","name
  
```

Como se puede apreciar, no es un problema directo de la escalabilidad del worker sino relacionado al procesamiento interno de este.

CONCLUSIONES FINALES GENERALES:

En términos generales, al realizar el conjunto de pruebas, la solución desarrollada no logró el cumplimiento de los indicadores de calidad o de desempeño predefinidos. Esto, a pesar de los múltiples escenarios planteados, por ejemplo: con diferentes números de tareas a procesar, e incluso con varios tamaños de archivos, puede concluirse que las métricas de desempeño obtenidas no fueron las esperadas, debido a algún problema de transición de Celery a lo que se está usando actualmente para hacer la transformación de los archivos.

Aunque las tareas se procesaron, lo hicieron en menor escala o tiempo de lo esperado y, además, la tasa de error fue superior al valor predefinido del 0,1 %. Esto puede ser atribuible, en parte, a una limitación que se encontró en GCP para procesar más de 150 hilos-tareas.

Por otro lado, vale la pena rescatar y resaltar que, durante las pruebas, el escalamiento SÍ operó, donde inicialmente se tenía un escalamiento de 1 a 3 MV, también se utilizó la opción de empezar con 2 Workers hasta llegar a 3, pero al parecer el problema no es debido al escalamiento.

Bibliografía

[1] Software Engineering at Google (abseil.io)