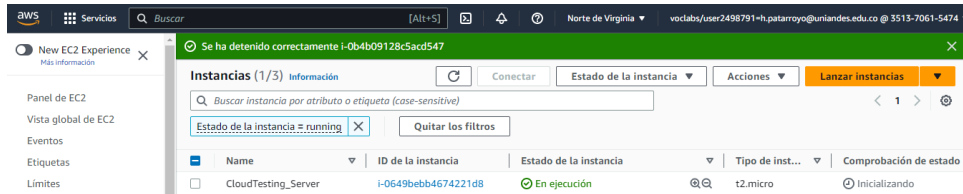


# ANÁLISIS DE CAPACIDAD MIGRACIÓN DE UNA APLICACIÓN WEB A LA NUBE PÚBLICA

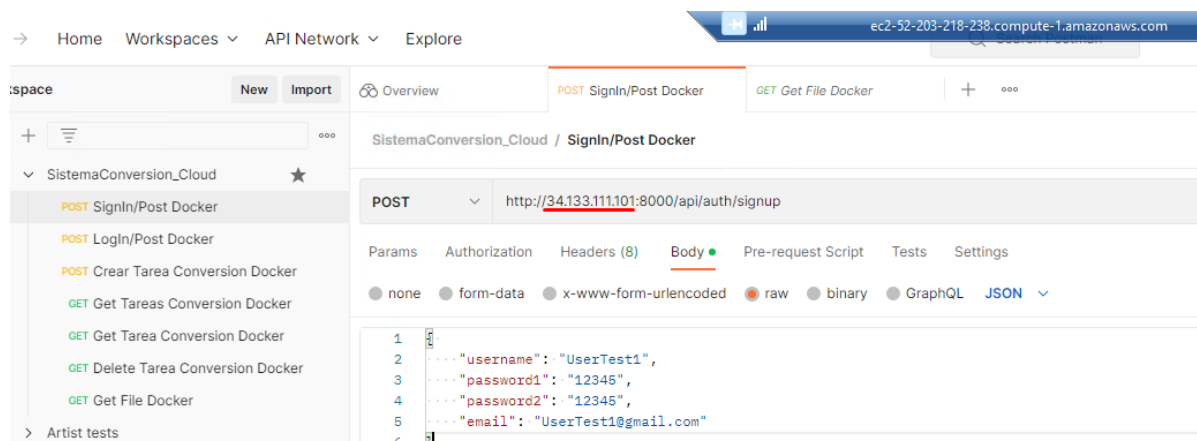
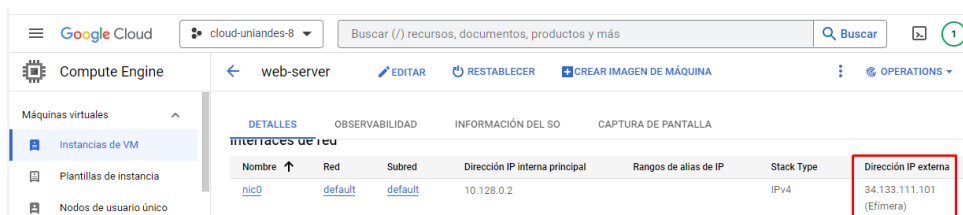
Las pruebas presentadas en este documento fueron realizadas en una instancia t2.micro de AWS.

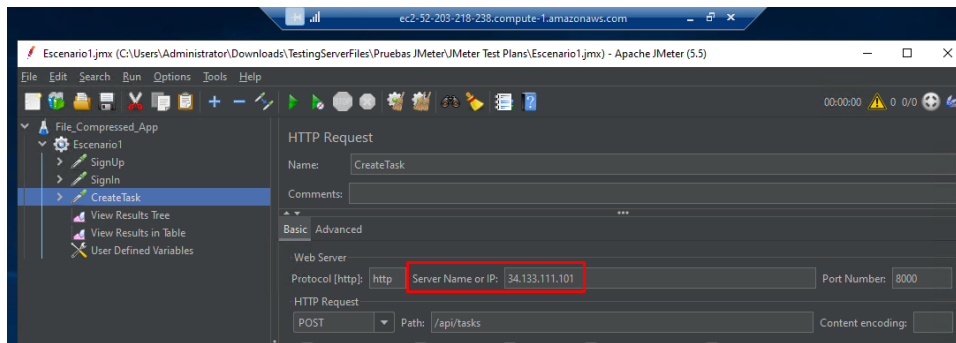


Las propiedades de este tipo de instancia se muestran en la siguiente imagen:

Instancia	CPU virtual*	Créditos por hora de CPU	Memoria (GiB)	Almacenamiento	Rendimiento de red
t2.nano	1	3	0,5	Solo EBS	Bajo
t2.micro	1	6	1	Solo EBS	De bajo a moderado

Para la realización de dichas pruebas las herramientas de Postman y Jmeter apuntaron a la IP externa de la instancia en GCP que corre el servicio Web de la aplicación.

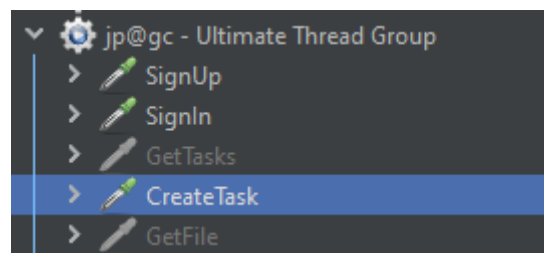




Se realizaron pruebas de rendimiento de acuerdo a los escenarios posteriormente descritos:

**Escenario 1.** Se deberá definir un escenario donde se pueda probar cuál es la máxima cantidad de requests HTTP por minuto que soporta la aplicación web con usuarios. Para hacer pruebas de estrés se debe utilizar la herramienta Apache Bench (ab) o JMeter. 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 clientes accediendo a la aplicación se incrementa hasta llegar al punto de degradar completamente el rendimiento de esta.

Para realizar esta prueba, se configuraron en JMeter las peticiones correspondientes al proceso de SignUp, SignIn, y Create Task.



Las peticiones de SignUp y SignIn eran necesarias en el grupo de peticiones ya que apartir de la información de respuesta de dichas peticiones se obtiene el requerido actualmente por la aplicación para poder realizar la petición de Create Task.

A continuación, se presentan los datos usados en las peticiones en mención:

## SignUp:

Petición HTTP

Nombre:

Comentarios

Basic Advanced

Servidor Web

Protocolo:  Nombre de Servidor o IP:

Petición HTTP

Ruta:

☐ Redirigir Automáticamente ☒ Seguir Redirecciones ☒ Utilizar KeepAlive ☐ Usar 'multipart/form-data'

Parameters Body Data Files Upload

```
1 {  
2   "username": "Sofia",  
3   "password1": "navidad",  
4   "password2": "navidad",  
5   "email": "sofia@navidad.com"  
6 }  
7
```

## Sign In:

Petición HTTP

Nombre:

Comentarios

Basic Advanced

Servidor Web

Protocolo:  Nombre de Servidor o IP:

Petición HTTP

Ruta:

☐ Redirigir Automáticamente ☒ Seguir Redirecciones ☒ Utilizar KeepAlive ☐ Usar 'multipart/form-data'

Parameters Body Data Files Upload

```
1 {  
2   "username": "Sofia",  
3   "password": "navidad"  
4 }  
5
```

## Create Task:

Petición HTTP

Nombre:

Comentarios:

Basic Advanced

Servidor Web

Protocolo:  Nombre de Servidor o IP:

Petición HTTP

POST

☐ Redirigir Automáticamente ☒ Seguir Redirecciones ☒ Utilizar KeepAlive ☐ Usar 'multipart/form-data' para HTTP POST ☐ Cabeceras compatibles con navegadores

Parameters Body Data Files Upload

Enviar Parámetros Con la Petición:

Nombre:	Valor
newFormat	ZIP

Petición HTTP

Nombre:

Comentarios:

Basic Advanced

Servidor Web

Protocolo:  Nombre de Servidor o IP:

Petición HTTP

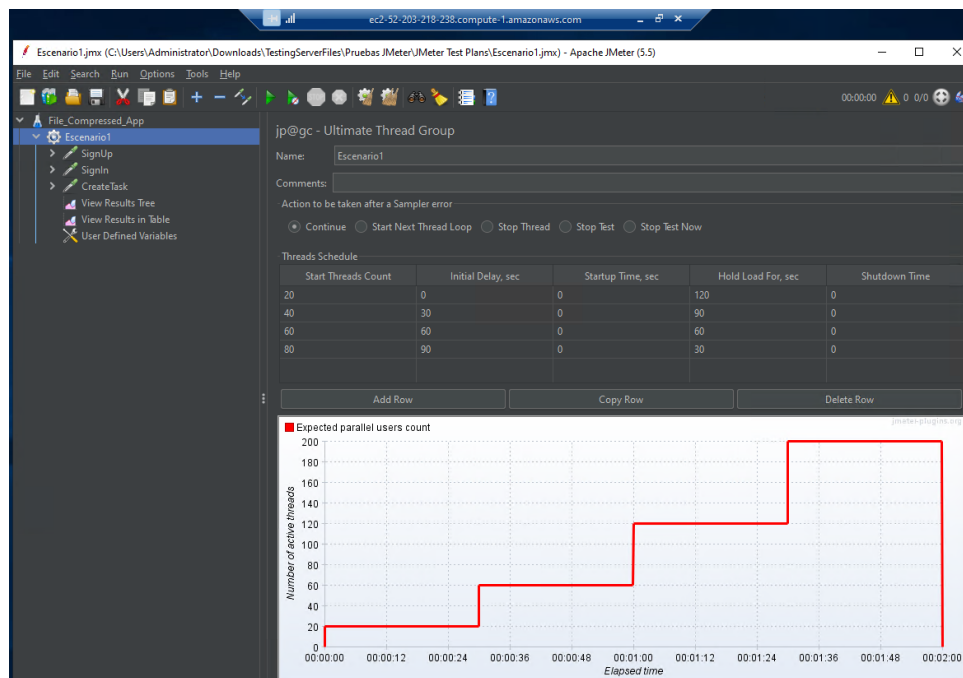
POST

☐ Redirigir Automáticamente ☒ Seguir Redirecciones ☒ Utilizar KeepAlive ☐ Usar 'multipart/form-data' para HTTP POST ☐ Cabeceras compatibles con navegadores

Parameters Body Data Files Upload

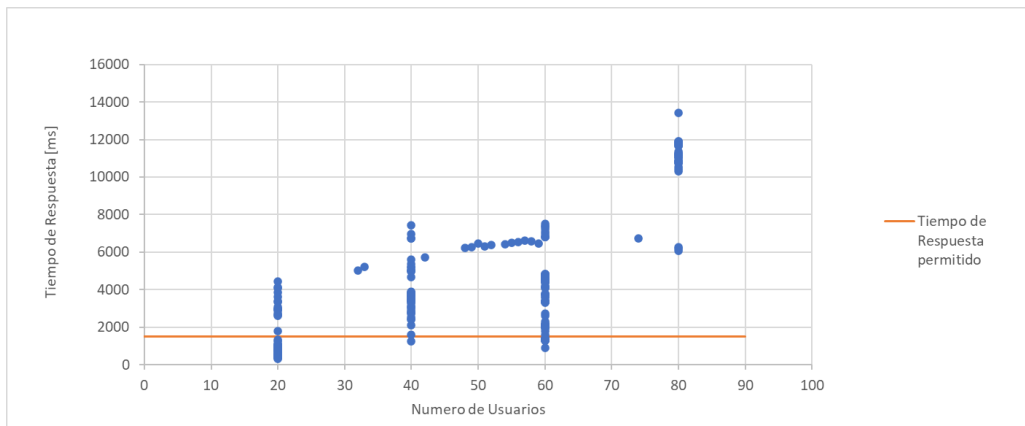
Nombre de Archivo:  Nombre de Parámetro:

La rampa de carga utilizada en este escenario se presenta en la siguiente imagen:

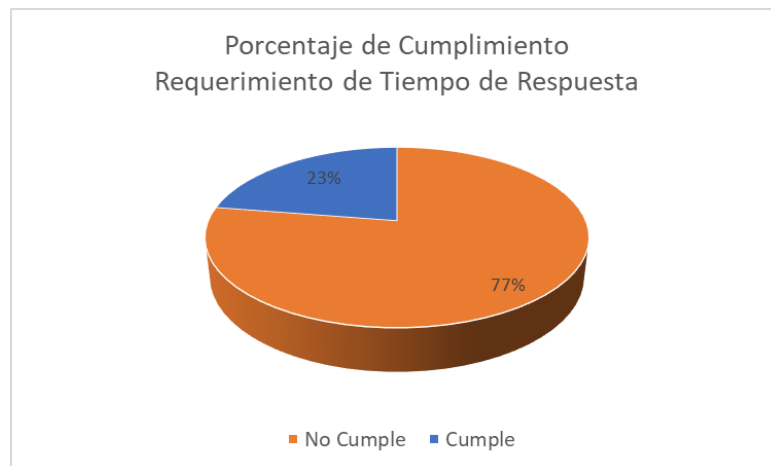


A continuación se presentan los resultados obtenidos para las pruebas realizadas en el despliegue local (Entrega Previa) y las pruebas realizadas sobre el despliegue en cloud (Entrega Actual).

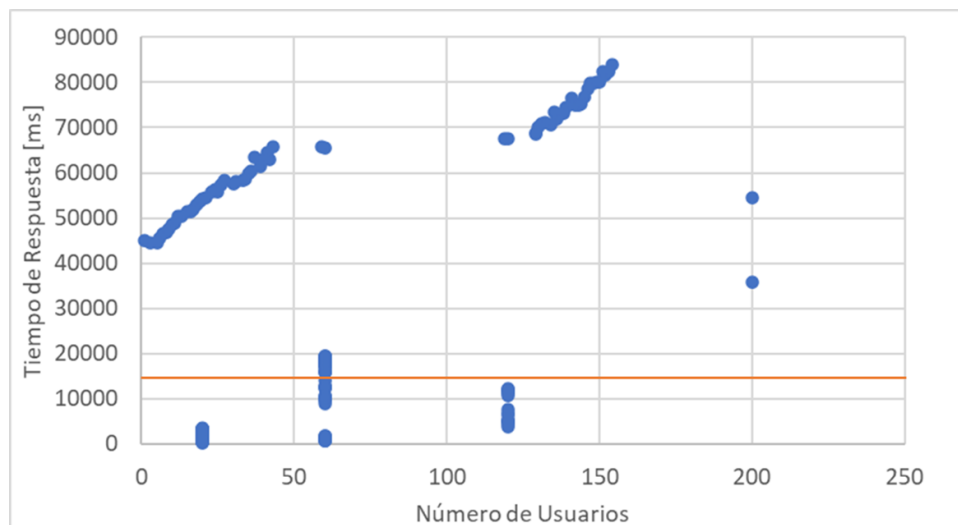
## Grafica Tiempo de Respuesta Vs Usuarios – Despliegue Local



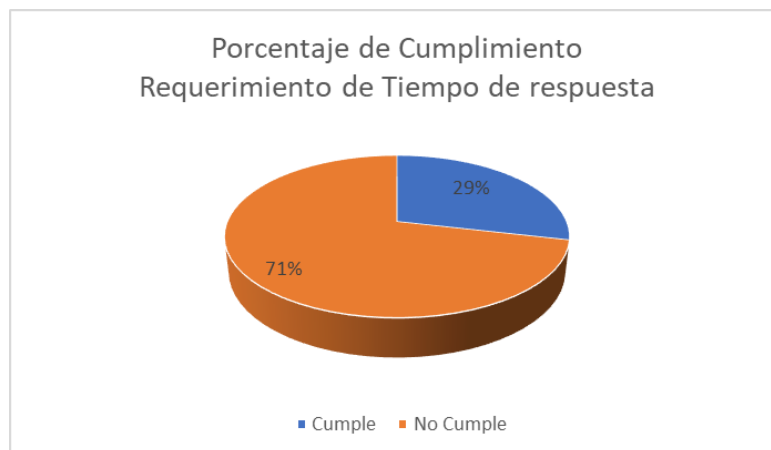
## Porcentaje Cumplimiento tiempo de respuesta – Despliegue Local



## Grafica Tiempo de Respuesta Vs Usuarios – Despliegue Cloud



## Porcentaje Cumplimiento tiempo de respuesta – Despliegue Cloud



En las pruebas realizadas en el despliegue local, el 77% de las pruebas realizadas no cumple con el requerimiento del tiempo de respuesta establecido (1500ms).

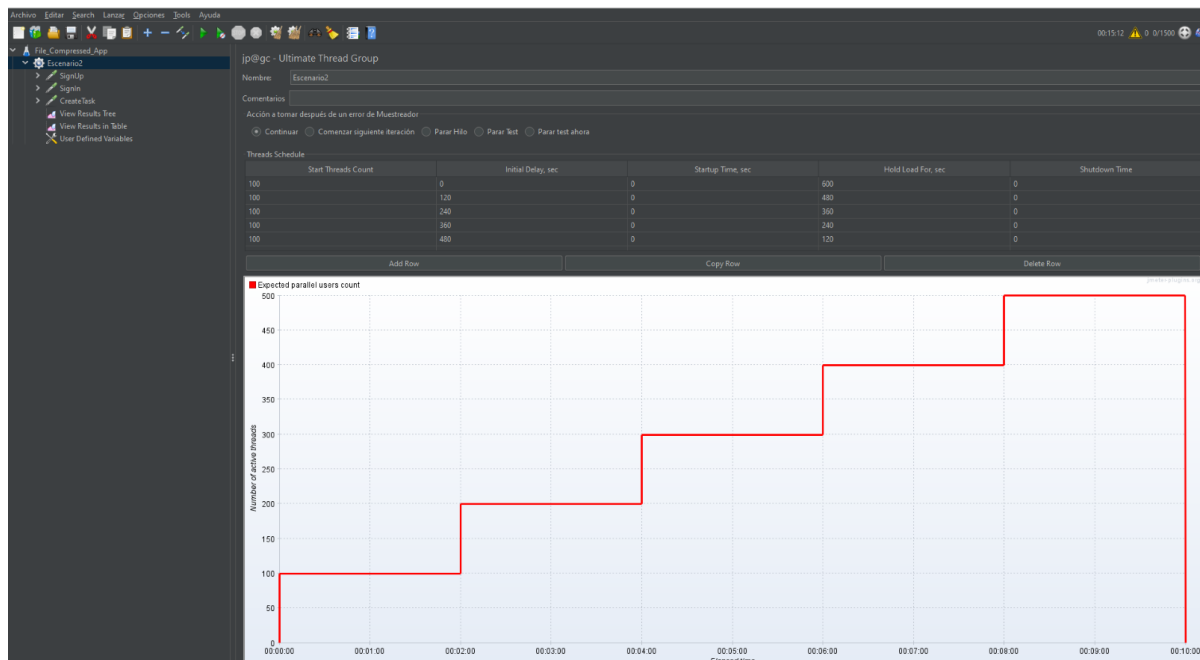
Aunque se esperaba que el despliegue en la nube mejorara el desempeño de la aplicación, el porcentaje de peticiones que cumplen el requerimiento de tiempo de respuesta es de tan solo 29%, 6% más de cumplimiento comparado con el despliegue local.

Algunas de las razones de este resultado pueden estar asociadas a la selección de un Cloud SQL Service de bajo desempeño (Disponibilidad Zonal: Zona Única) lo cual para efectos de esta asignación fue una decisión sustentada en ahorro de créditos disponibles.

Adicionalmente, es posible que la velocidad de red afecte los resultados obtenidos por lo que sería recomendable replicar el escenario considerando esta variable para tener más claridad sobre su efecto en los resultados obtenidos.

**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 local. Para hacer pruebas de estrés se recomienda utilizar la herramienta Apache Bench (ab) o JMeter. Las pruebas de estrés deberán realizarse desde otro equipo diferente 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 la aplicación a medida que el número de usuarios procesando 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). Restricciones del escenario. El archivo enviado a convertir durante las pruebas debe ser de un tamaño mínimo de 15 MiB y un máximo de 20 MiB.

Para la realización de esta prueba se utilizó un archivo de 20 MiB. En JMeter se creó una rampa que se muestra en la siguiente imagen:



Esta rampa tiene una duración total de 10 minutos.

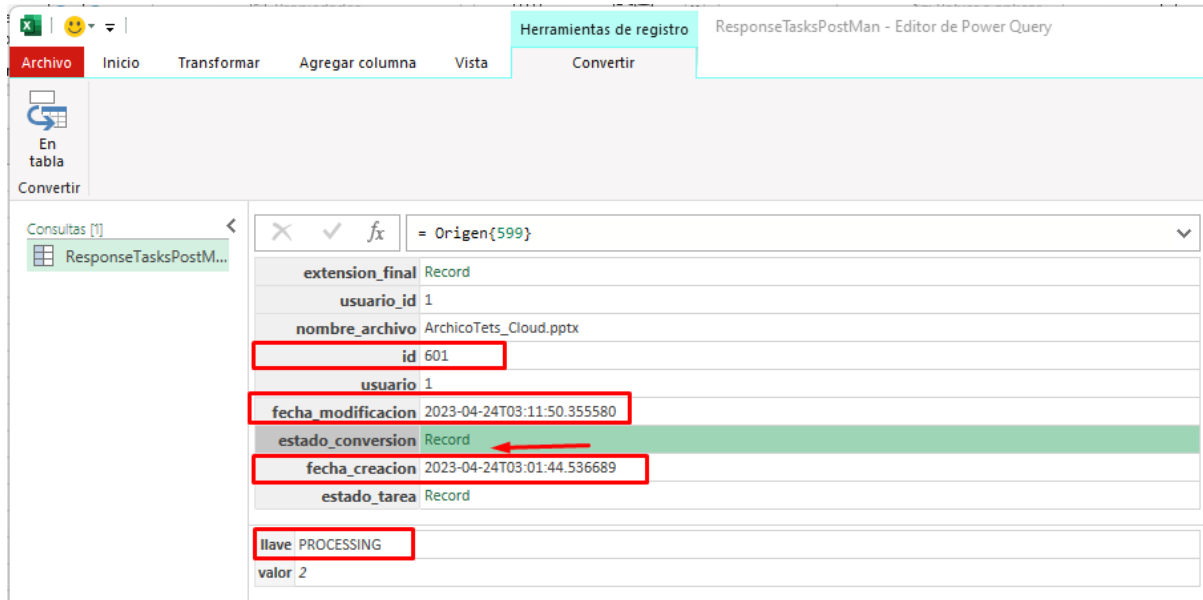
A continuación, se presentan los resultados obtenidos para las pruebas realizadas en el despliegue local (Entrega Previa) y las pruebas realizadas sobre el despliegue en cloud (Entrega Actual).

De acuerdo a la información obtenida en la entrega anterior se sabe que en el despliegue local la aplicación logró procesar 365 archivos en 18 minutos (10 minutos de tiempo de prueba más 8 minutos de espera) por lo que se estimó una velocidad de procesamiento de 20 archivos por minuto.

Para las pruebas de este escenario en el despliegue de Cloud se realizó la misma metodología usada en la entregada anterior. Una vez se corrieron las pruebas en JMeter se esperó 8 minutos y se hizo una consulta a la base de datos para obtener todas las tareas creadas hasta el momento con su respectivo estado.

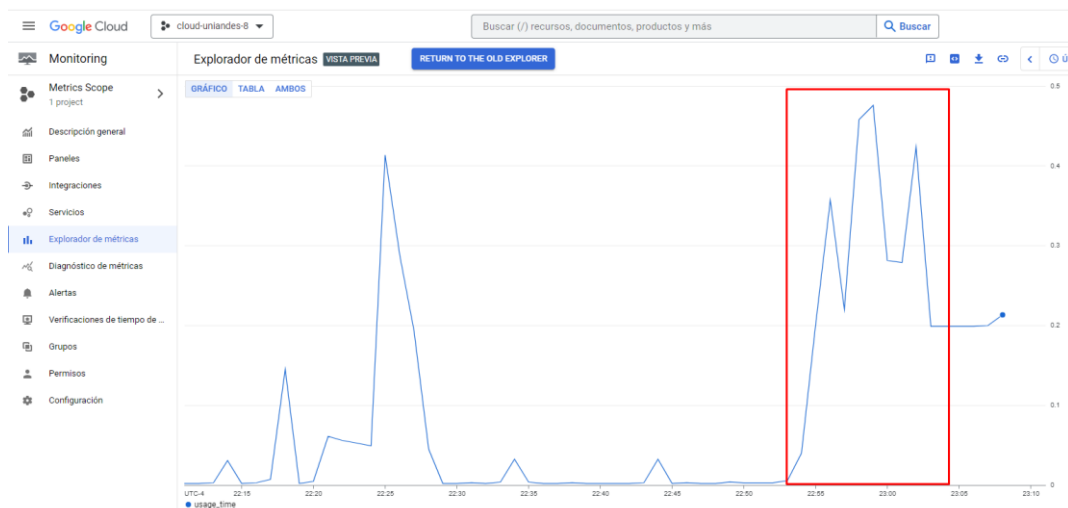
Dicha consulta se realizó a través de Postman usando en el endpoint de Tasks, la respuesta de la petición fue exportada en formato Json y posteriormente analizada en Excel donde se identificó en el durante los 18 minutos transcurridos desde el inicio de la prueba se habían procesado 322 archivos (teniendo en cuenta que en la base de datos ya existían 279 tareas del escenario 1 que ya estaban procesados antes de empezar esta prueba). Dadas estas condiciones se estimó una velocidad de procesamiento de 18 archivos/minuto.

Como se muestra en la siguiente imagen, para la tarea con ID 601 pasaron 10 minutos desde su creación hasta el tiempo de modificación en el cual su estado corresponde a *Processing*.



extension_final	usuario_id	nombre_archivo	id	usuario	fecha_modificacion	estado_conversion	fecha_creacion	estado_tarea	llave	valor
Record	1	ArchicoTets_Cloud.pptx	601	1	2023-04-24T03:11:50.355580	Record	2023-04-24T03:01:44.536689	Record	PROCESSING	2

En adición a los datos capturados haciendo uso de la herramienta JMeter, usando la funcionalidad de Monitoring disponible en GCP se capturó la información de utilización de CPU durante la ejecución de la prueba.



Los resultados obtenidos en las pruebas realizadas al despliegue cloud son similares a los obtenidos en el despliegue local. Una de las razones de la baja velocidad de procesamiento de archivos puede estar asociada a la baja disponibilidad del servicio



Cloud SQL seleccionado (decisión tomada por temas de ahorro de costos). La gráfica del uso de CPU muestra que durante la ejecución de la prueba los valores de utilización oscilan entre 20% y 50%, alcanzando el valor máximo a la mitad de la prueba punto donde se identificó que el tiempo de procesamiento supera los 10 minutos.

## **Conclusiones**

Dado que el requerimiento en la velocidad de procesamiento de las peticiones no fue cumplido se debe ajustar el escenario de despliegue y la asignación de recursos para poder alcanzar los objetivos operacionales propuestos.

La implementación de rampas de carga con mayor nivel de detalle podría aumentar la precisión en las estimaciones correspondientes a los puntos de degradación del servicio. Para estimaciones futuras debe analizar si es conveniente cambiar la formulación de dicha rampa y discutir los cambios que puedan mejorar la calidad de los resultados.

En el desarrollo de esta actividad tenemos una limitante asociada a los créditos disponibles en las plataformas cloud lo que hace que los recursos que usemos sean los que tienen menores recursos o están incluidos en el modo gratuito. En condiciones reales la selección de los recursos podrá ajustarse a los requerimientos del negocio de acuerdo al presupuesto disponible.