



# Plan de pruebas

## Objetivo

Evaluar el rendimiento y la capacidad de respuesta de la aplicación Cloud Conversion Tool bajo diferentes escenarios y cargas de trabajo.

## Objetivos específicos

- Evaluar el tiempo de respuesta de la aplicación bajo diferentes cargas de trabajo.
- Determinar el límite de capacidad de la aplicación en términos de usuarios concurrentes y transacciones por minuto.
- Identificar y abordar cuellos de botella en el rendimiento del sistema.
- Comprobar que la aplicación cumple con los criterios de aceptación definidos en cuanto a tiempo de respuesta y tasa de errores.

## Descripción general

La prueba evaluará el rendimiento de la aplicación de conversión de formatos de video bajo diferentes escenarios de carga simulada. Conforme a nuestros criterios de aceptación, la aplicación deberá mantener tiempos de respuesta aceptables y tasas de errores bajas bajo estas condiciones.

## Tipos de prueba a realizar

- Pruebas de capacidad
- Pruebas de carga

## Entorno de prueba

El entorno de pruebas requerido para realizar este plan debe coincidir con las siguientes características a nivel de hardware y software de un dispositivo local.

Hardware:

- Procesador: 2.6 GHz Intel Core i7 de seis núcleos
- Memoria: 16 GB 2667 MHz DDR4

Software:

- SO: macOS Ventura Versión 13.6

Red:

- Se configura interfaz de red tipo Bridge Network, con la finalidad de obtener la misma información de conexión del Host.

## Herramientas utilizadas

Entre las tecnologías que utilizamos se encuentran Docker, que nos permite empacar nuestras aplicaciones en contenedores para lograr un despliegue sencillo y consistente en diversos entornos. Empleamos Redis, un ágil motor de base de datos en memoria que ofrece un almacenamiento eficiente y opcionalmente persistente. Para la ejecución de tareas paralelas y asíncronas, hemos incorporado Celery, una biblioteca de Python de código abierto. Adicionalmente, utilizamos Postman para realizar pruebas y verificar el funcionamiento de nuestros endpoints. Por último, nuestra base de datos relacional está respaldada por Postgres, un sistema de gestión de bases de datos de código abierto y orientado a objetos.

Además, las herramientas de prueba a utilizar son:

- Apache Bench (ab) y JMeter para la creación y ejecución de escenarios de prueba.
- Herramientas de monitoreo de recursos para supervisar el rendimiento del sistema.

## Métricas consideradas

- Respuestas HTTP por segundo.
- Tiempo mínimo, máximo y promedio de respuesta.
- Respuestas por códigos HTTP.
- Uso de CPU, memoria y almacenamiento.

## Riesgos y limitaciones

La disponibilidad de recursos como hardware, ancho de banda de red y servidores puede impactar la ejecución de las pruebas.

Las pruebas se realizan en un entorno controlado, lo que significa que los resultados pueden no reflejar completamente el rendimiento en un entorno de producción real.

## Datos de prueba

Cuentas de Usuario:

Se utilizarán cuentas de usuario válidas para simular las operaciones de inicio de sesión y autenticación. Cada cuenta de usuario deberá incluir la siguiente información:

- Email válido.
- Nombre de usuario (username).
- Contraseña válida para la autenticación.

Datos de Entrada para Tareas:

Para las pruebas de creación de tareas de conversión, se necesitarán datos de entrada válidos. Cada tarea de conversión debe incluir:

- Nombre del archivo multimedia que se desea convertir.
- Formato de conversión al que se desea transformar el archivo.

Bases de Datos de Prueba:

Se utilizará una base de datos de prueba para reflejar el estado inicial del sistema antes de ejecutar las pruebas. La base de datos de prueba debe contener datos consistentes con las cuentas de usuario, tareas de conversión y otros datos necesarios para las pruebas.

## Criterios de aceptación

- El 100% de las peticiones configuradas y realizadas deben ser atendidas por el aplicativo.
- El tiempo de respuesta de las peticiones debe ser menos a 2 segundos y debe ser un código status exitoso (Familia de los códigos 2XX)
- Criterios de exitoso (para configuraciones diferentes)
- Se considera exitoso a la combinación de ajustes en el consumo de memoria menor al 80% asignada al recurso computacional.
- La aplicación Docker no debe de utilizar más del 70% del CPU disponible cuando los contenedores estén ejecución.
- El 100% de las peticiones configuradas y realizadas deben ser atendidas por el aplicativo con la combinación de ajustes definidas.

En relación con el desarrollo de la aplicación en entornos de prueba se definieron los siguientes criterios de aceptación:

- Escenario 1:
  - El sistema debe convertir el archivo al formato especificado por el usuario sin errores.
  - El archivo convertido debe tener un formato correcto y ser funcional según el nuevo formato.
  - El tiempo promedio de respuesta de la aplicación debe ser menos de 1500 ms.
  - La aplicación debe soportar como mínimo el 99% de los requests enviados en la prueba.
  - Evaluar la configuración de ajustes donde se obtengan las características de rendimiento más deseables.
- Escenario 2:
  - Se debe probar la cantidad de archivos que la aplicación puede procesar en 1 minuto.
  - Evaluar máximo hasta que el tiempo en cargar un archivo a la aplicación tarde 600 segundos.
  - Los archivos enviados en la prueba deben de ser máximo de 5 MB.
  - Evaluar la configuración de ajustes donde se obtengan las características de rendimiento más deseables.

## Escenarios de prueba

### Escenario 1

Subir y cambiar el formato de un archivo. El usuario debe proveer el archivo que desea convertir, el formato al cual desea cambiarlo y el token de autenticación para realizar dicha operación. El archivo debe ser almacenado en la plataforma, se debe guardar en base datos la marca de tiempo en el que fue subido el archivo y el estado del proceso de conversión (uploaded).

Datos de prueba: Nombre del archivo, formato de conversión, token del usuario

Métricas a recopilar: Tiempo de respuesta.

### Escenario 2

Para el segundo escenario se probará la máxima cantidad de conversión de formatos de archivos procesados de capacidad no mayor a 10MB que se pueden procesar en un minuto por un total de cincuenta usuarios de manera concurrente. Las métricas que se evaluarán durante la prueba serán capacidad de procesamiento mediante el procesamiento de archivos por minuto, tiempo de respuesta promedio y utilización de recursos. Las tres métricas deben cumplir con los criterios de aceptación definidos anteriormente para que la prueba se considere exitosa. Esta información se representará en una gráfica que ilustre el comportamiento de la aplicación en base a la recopilación de la prueba.

Datos de prueba: Nombre del archivo, formato de conversión, token del usuario

Métricas a recopilar: Tiempo de respuesta promedio y utilización de recursos.

## Ejecución de las pruebas

Después de tener el ambiente desarrollado en AWS, utilizamos la herramienta Apache Benchmark y el endpoint en el que se manda la petición de conversión de archivos `api/tasks`. Adicionalmente utilizamos la librería Gnuplot para graficar los resultados de las pruebas.

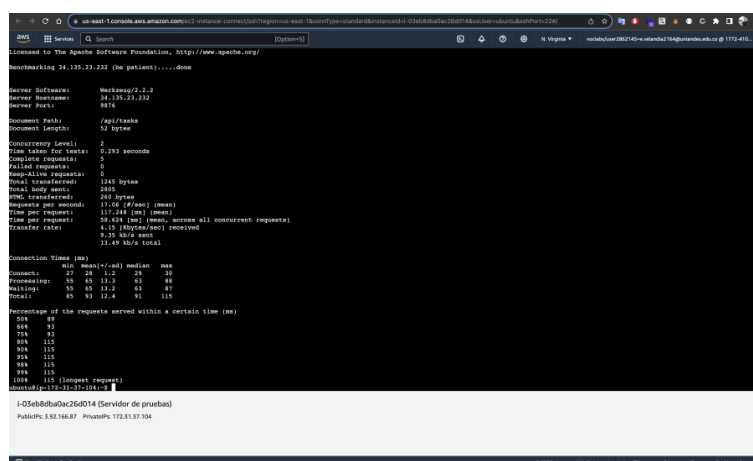
El comando `ab` (Apache Benchmark) se utiliza para realizar pruebas de carga en un servidor web, teniendo en cuenta los siguientes parámetros:

- n 50: Realiza un total de 50 solicitudes.
- c 50: Mantiene 50 conexiones concurrentes.
- p body-req.json: Envía el cuerpo de la solicitud desde el archivo "body-req.json".
- H 'Authorization: Bearer eyJhbGciOiJIUzI1Ni...': Incluye una cabecera de autorización con un token.
- T 'application/json': Especifica el tipo de contenido de la solicitud como JSON.
- H 'Content-Type: application/json': Indica que la solicitud es de tipo JSON.
- rk: Muestra resultados en formato CSV y mantiene las conexiones HTTP/1.1 persistentes.
- g output.csv: Guarda los resultados en un archivo CSV llamado "output.csv".
- http://34.70.19.70:9876/api/tasks: Realiza las solicitudes al servidor web en la dirección URL especificada.

Archivo JSON del endpoint utilizado:

```
{
  "fileName": "/app/files/base/video1.mp4",
  "newFormat": "avi"
}
```

La ejecución de las pruebas se hizo en una instancia EC2 alojada en AWS con las siguientes características:



```

Benchmarking 34.135.23.232 (be patient).....done

Server Software:      Werkzeug/2.2.2
Server Hostname:     34.135.23.232
Server ports:        8080
Document Path:       /api/tasks
Document Length:     52 bytes

Concurrency Level:   2
Time taken for tests: 0.293 seconds
Complete requests:   5
Failed requests:     0
Keep-Alive requests: 0
Total transfer size: 260 bytes
Total time taken:    293 ms
Internal transfer rate: 260 bytes/sec
Requests per second: 17.06 [#/sec] (mean)
Time per request:    17.248 [ms] (mean)
Time per request:    20.048 [ms] (mean across all concurrent requests)
Transfer rate:       4.15 [Kbytes/sec] received
Total time:          0.293000 sec
Total data sent:     11.49 Kbytes total

Connection Times (ms)
  distribution
  min  mean(=)  max
Connect:  27  28  1.2  29  30
Processing:  50  45  13.0  63  88
Waiting:  35  40  13.0  63  87
Total:  60  53  26.4  95  115

Percentage of the requests served within a certain time (ms)
 50%  87
 60%  93
 70%  99
 80%  110
 90%  112
 95%  113
 98%  115
 99%  115
100%  115 (longest request)
#####112-112-114112#####

i-05eb8ba0ac26d014 (Servidor de pruebas)
Public IP: 52.166.87 Private IP: 172.31.37.104
```

### - Escenario 1

Los parámetros de configuración para el primer escenario usando herramienta **Apache Bench** (`ab`) son los siguientes: `ab`

- 5 peticiones con 2 de concurrencia

Ejecución de la prueba:

```

Licensed to The Apache Software Foundation, http://www.apache.org/
Benchmarking 34.135.23.232 (be patient).....done

Server Software:      nginx/2.2.2
Server Hostname:     34.135.23.232
Server Port:         9876

Document Path:       /api/tasks
Document Length:     50 bytes

Concurrency Level:   2
Time taken for tests: 0.293 seconds
Complete requests:   5
Failed requests:     0
Keep-Alive requests: 0
Total transferred:  1245 bytes
Total body sent:    260
HTML transferred:   260 bytes
Requests per second: 17.04 [#/sec] (mean)
Time per request:   117.248 [ms] (mean)
Time per request:   58.624 [ms] (mean, across all concurrent requests)
Transfer rate:      4.15 [Kbytes/sec] received
                   9.35 kb/s sent
                   13.49 kb/s total

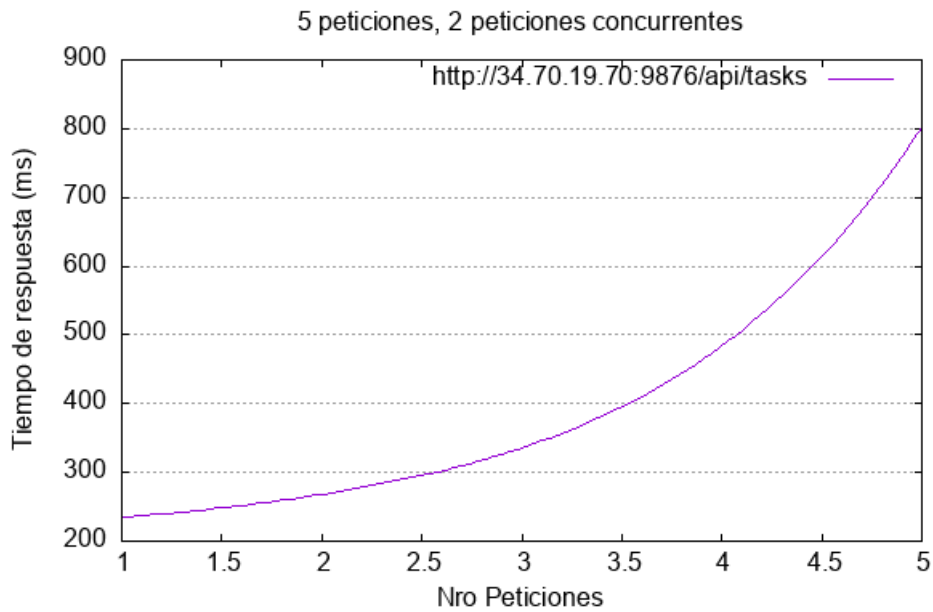
Connection Times (ms)
  min  mean[+/-sd] median max
Connect: 27  28  [+/-2]   29  30
Processing: 55  45  [13.3]   43  88
Waiting: 55  45  [13.2]   43  88
Total: 89  93  [13.4]   91  115

Percentage of the requests served within a certain time (ms)
 50%  89
 66%  93
 75%  93
 80%  115
 90%  115
 95%  115
 99%  115
100%  115 (longest request)

i-03eb8dba0ac26d014 (Servidor de pruebas)
PublicIP: 192.168.87 PrivateIP: 172.31.17.104

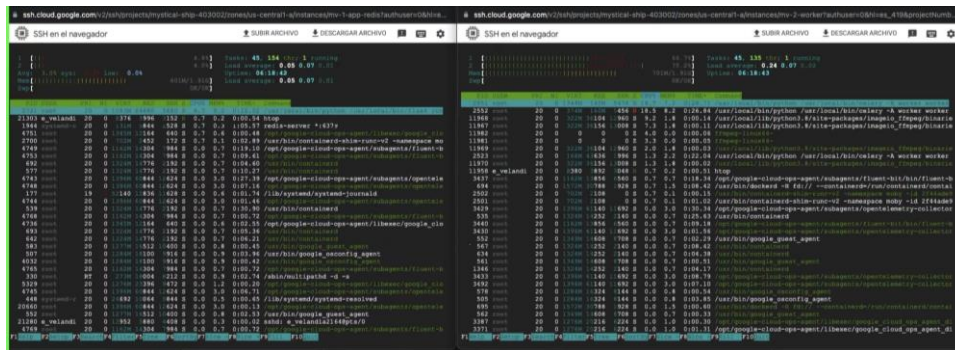
```

Gráfico:



starttime	seconds	ctime	dtime	ttime	wait
Fri Oct 27 07:30:42 2023	1698391842	30	55	85	55
Fri Oct 27 07:30:42 2023	1698391842	29	56	85	56
Fri Oct 27 07:30:42 2023	1698391842	27	62	89	61
Fri Oct 27 07:30:42 2023	1698391842	29	64	93	64

Uso de recursos:



- 10 peticiones con 3 de concurrencia

Ejecución de la prueba:

```

Benchmarking 34.70.19.70 (be patient)....done

Server Software: Werkzeug/2.2.2
Server Hostname: 34.70.19.70
Server Port: 9876

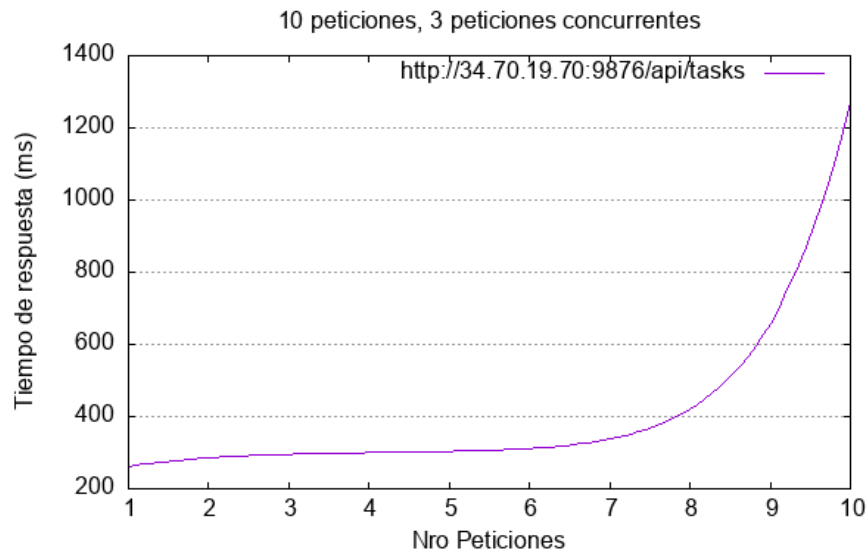
Document Path: /api/tasks
Document Length: 53 bytes

Concurrency Level: 3
Time taken for tests: 2.184 seconds
Complete requests: 10
Failed requests: 0
Keep-Alive requests: 0
Total transferred: 2500 bytes
Total body sent: 5570
HTML transferred: 530 bytes
Requests per second: 4.58 [#]/sec (mean)
Time per request: 655.106 [ms] (mean)
Time per request: 218.369 [ms] (mean, across all concurrent requests)
Transfer rate: 1.12 [Kbytes/sec] received
                2.49 kb/s sent
                3.61 kb/s total

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:  108  227 312.2   126  1114
Processing:  138  174  16.4    179   192
Waiting:  138  174  16.4    179   192
Total:  261  401 306.2   303  1270

Percentage of the requests served within a certain time (ms)
 50%    303
 66%    311
 75%    311
 80%    352
 90%   1270
 95%   1270
 98%   1270
 99%   1270
100%   1270 (longest request)
    
```

Gráfico:



Ejecución de la prueba:

- 20 peticiones con 5 de concurrencia

Ejecución de la prueba:

```

Benchmarking 34.70.19.70 (be patient).....done
Server Software: Werkzeug/2.2.2
Server Hostname: 34.70.19.70
Server Port: 9876

Document Path: /api/tasks
Document Length: 53 bytes

Concurrency Level: 5
Time taken for tests: 1.507 seconds
Complete requests: 20
Failed requests: 0
Keep-Alive requests: 0
Total transferred: 5000 bytes
Total body sent: 11140
HTML transferred: 1060 bytes
Requests per second: 13.27 [#/sec] (mean)
Time per request: 376.839 [ms] (mean)
Time per request: 75.368 [ms] (mean, across all concurrent requests)
Transfer rate: 3.24 [Kbytes/sec] received
              7.22 kb/s sent
              10.46 kb/s total

Connection Times (ms)
   min  mean[+/-sd] median  max
Connect:  108  122  12.4   122  152
Processing:  146  163  14.2   158  195
Waiting:  146  163  14.5   158  195
Total:  254  285  19.6   283  334

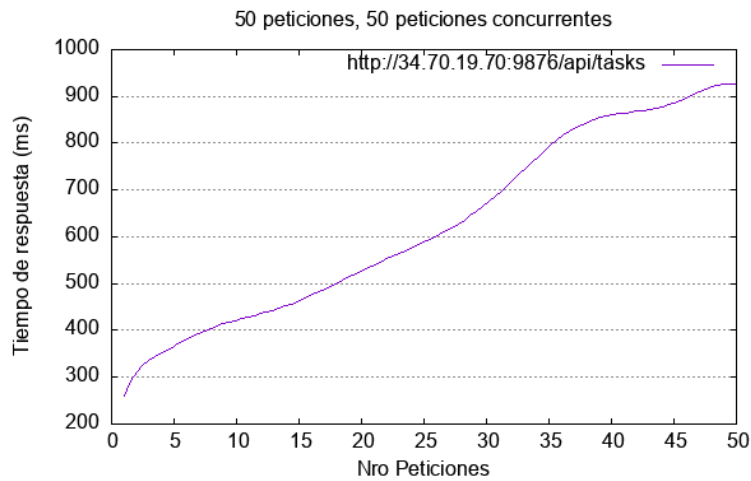
Percentage of the requests served within a certain time (ms)
 50%  283
 66%  291
 75%  300
 80%  303
 90%  313
 95%  334
 98%  334
 99%  334
100%  334 (longest request)

```

Gráfico:







starttime	seconds	ctime	dtime	ttime	wait
Fri Oct 27 06:04:46 2023	1698386686	26	77	103	77
Fri Oct 27 06:04:46 2023	1698386686	30	74	104	73
Fri Oct 27 06:04:46 2023	1698386686	26	236	262	236
Fri Oct 27 06:04:46 2023	1698386686	26	266	292	266
Fri Oct 27 06:04:46 2023	1698386686	26	341	367	341
Fri Oct 27 06:04:46 2023	1698386686	27	366	393	366
Fri Oct 27 06:04:46 2023	1698386686	26	367	393	367
Fri Oct 27 06:04:46 2023	1698386686	28	383	411	383
Fri Oct 27 06:04:46 2023	1698386686	27	420	447	417
Fri Oct 27 06:04:46 2023	1698386686	27	485	512	485
Fri Oct 27 06:04:46 2023	1698386686	26	493	519	493
Fri Oct 27 06:04:46 2023	1698386686	27	494	521	472
Fri Oct 27 06:04:46 2023	1698386686	29	495	523	484
Fri Oct 27 06:04:46 2023	1698386686	26	504	531	504
Fri Oct 27 06:04:46 2023	1698386686	29	503	532	499
Fri Oct 27 06:04:46 2023	1698386686	26	509	535	509
Fri Oct 27 06:04:46 2023	1698386686	27	512	539	512
Fri Oct 27 06:04:46 2023	1698386686	26	526	552	525
Fri Oct 27 06:04:46 2023	1698386686	27	529	556	529
Fri Oct 27 06:04:46 2023	1698386686	29	535	563	534
Fri Oct 27 06:04:46 2023	1698386686	26	552	578	552
Fri Oct 27 06:04:46 2023	1698386686	37	543	580	543
Fri Oct 27 06:04:46 2023	1698386686	28	552	581	552
Fri Oct 27 06:04:46 2023	1698386686	36	567	603	567
Fri Oct 27 06:04:46 2023	1698386686	36	569	605	569
Fri Oct 27 06:04:46 2023	1698386686	36	574	610	570
Fri Oct 27 06:04:46 2023	1698386686	36	578	614	578
Fri Oct 27 06:04:46 2023	1698386686	37	588	625	588
Fri Oct 27 06:04:46 2023	1698386686	37	615	652	615



Ejecución de la prueba:

- 100 peticiones con 50 de concurrencia

```
Benchmarking 34.70.19.70 (be patient).....done

Server Software: Werkzeug/2.2.2
Server Hostname: 34.70.19.70
Server Port: 9876

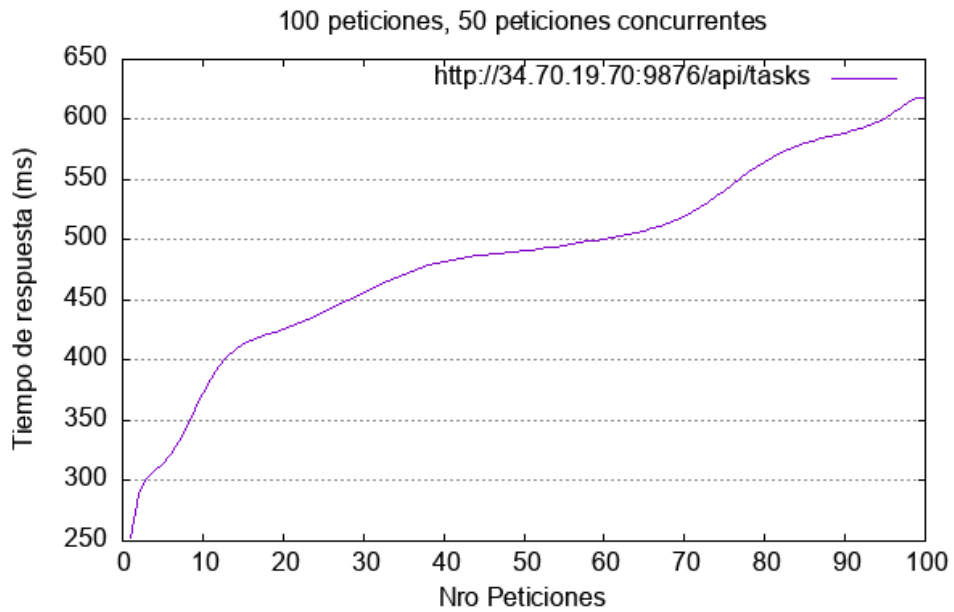
Document Path: /api/tasks
Document Length: 92 bytes

Concurrency Level: 50
Time taken for tests: 1.359 seconds
Complete requests: 100
Failed requests: 0
Non-2xx responses: 100
Keep-Alive requests: 0
Total transferred: 29800 bytes
Total body sent: 56000
HTML transferred: 9200 bytes
Requests per second: 73.56 [#/sec] (mean)
Time per request: 679.701 [ms] (mean)
Time per request: 13.594 [ms] (mean, across all concurrent requests)
Transfer rate: 21.41 [Kbytes/sec] received
                40.23 kb/s sent
                61.64 kb/s total

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:  116  206  64.2   217   366
Processing: 137  282  76.0   276   432
Waiting:   135  279  75.0   276   430
Total:     253  487  80.1   492   618

Percentage of the requests served within a certain time (ms)
 50%    492
 66%    509
 75%    551
 80%    578
 90%    589
 95%    597
 98%    618
 99%    618
100%    618 (longest request)
```

Gráfico:



- 150 peticiones con 50 de concurrencia

Ejecución de la prueba:

```

Benchmarking 34.70.19.70 (be patient)....done

Server Software: Werkzeug/2.2.2
Server Hostname: 34.70.19.70
Server Port: 9876

Document Path: /api/tasks
Document Length: 92 bytes

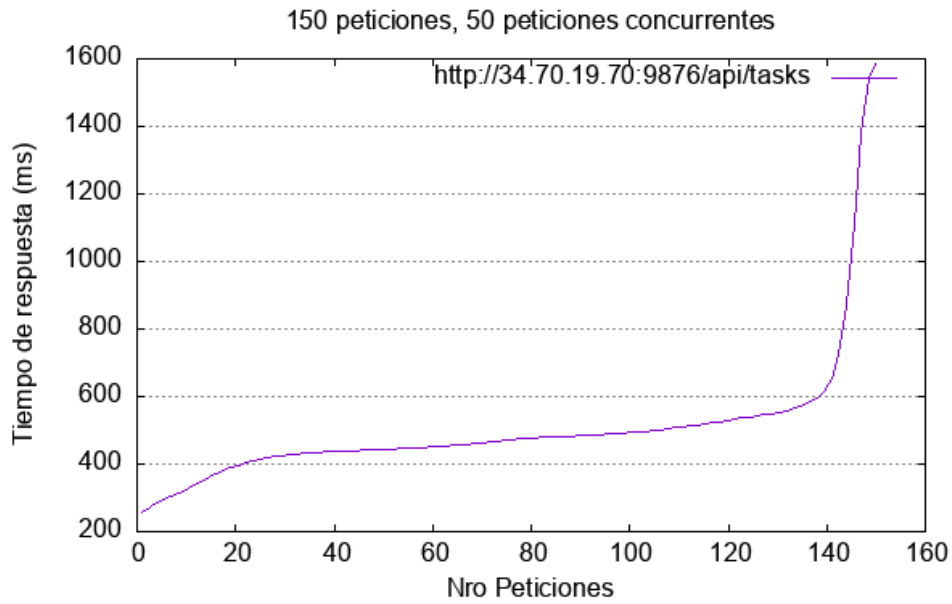
Concurrency Level: 50
Time taken for tests: 2.010 seconds
Complete requests: 150
Failed requests: 0
Non-2xx responses: 150
Keep-Alive requests: 0
Total transferred: 44700 bytes
Total body sent: 84000
HTML transferred: 13800 bytes
Requests per second: 74.62 [#/sec] (mean)
Time per request: 670.097 [ms] (mean)
Time per request: 13.402 [ms] (mean, across all concurrent requests)
Transfer rate: 21.71 [Kbytes/sec] received
40.81 kb/s sent
62.52 kb/s total

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:  108  201  52.0   213   312
Processing:  139  301  216.7  254  1427
Waiting:  134  298  217.0  253  1427
Total:  257  501  210.4  472  1586

Percentage of the requests served within a certain time (ms)
 50%  472
 66%  491
 75%  516
 80%  531
 90%  576
 95%  607
 98%  1554
 99%  1557
100% 1586 (longest request)

```

Gráfico:



- 200 peticiones con 50 de concurrencia

Ejecución de la prueba:

```

Benchmarking 34.70.19.70 (be patient)
Completed 100 requests
Completed 200 requests
Finished 200 requests

Server Software: Werkzeug/2.2.2
Server Hostname: 34.70.19.70
Server Port: 9876

Document Path: /api/tasks
Document Length: 92 bytes

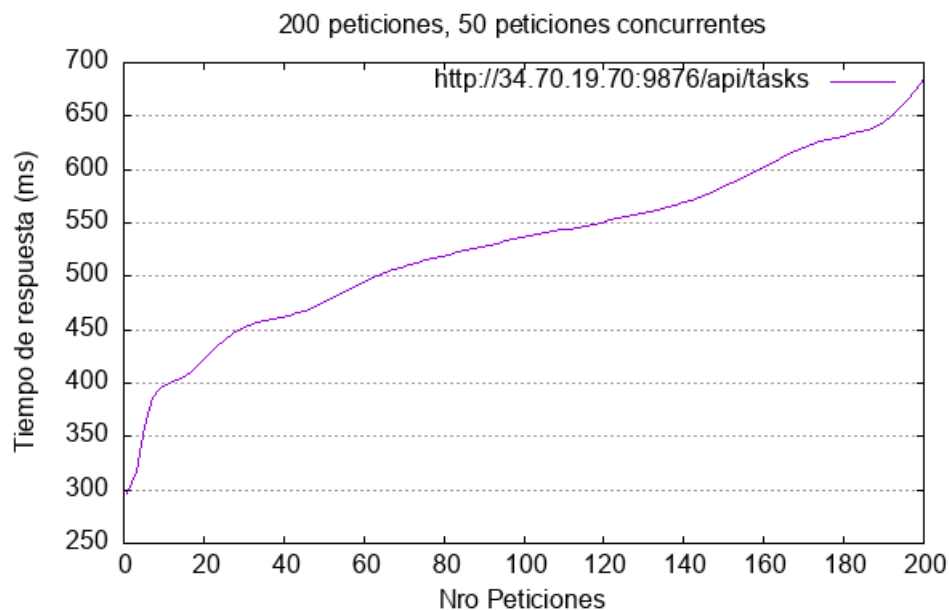
Concurrency Level: 50
Time taken for tests: 2.966 seconds
Complete requests: 200
Failed requests: 0
Non-2xx responses: 200
Keep-Alive requests: 0
Total transferred: 59600 bytes
Total body sent: 112000
HTML transferred: 18400 bytes
Requests per second: 67.42 [# /sec] (mean)
Time per request: 741.567 [ms] (mean)
Time per request: 14.831 [ms] (mean, across all concurrent requests)
Transfer rate: 19.62 [Kbytes/sec] received
36.87 kb/s sent
56.49 kb/s total

Connection Times (ms)
min mean[+/-sd] median max
Connect: 113 240 58.0 237 508
Processing: 118 291 78.4 278 460
Waiting: 116 288 78.7 271 460
Total: 296 531 77.7 538 685

Percentage of the requests served within a certain time (ms)
50% 538
66% 561
75% 584
80% 605
90% 633
95% 653
98% 673
99% 679
100% 685 (longest request)

```

Gráfico:



### Análisis de resultados

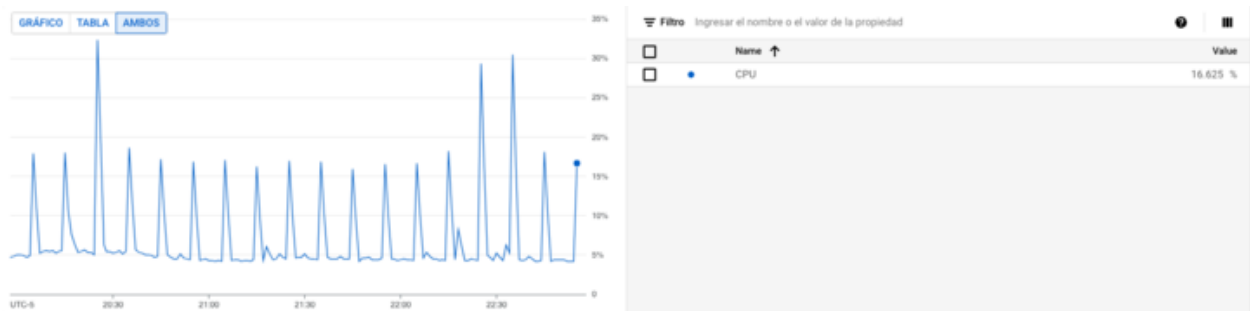
Cuando aumentamos la concurrencia significa que hay múltiples solicitudes en ejecución al mismo tiempo. Esto ejerce una mayor presión sobre el servidor, ya que debe administrar y procesar varias solicitudes simultáneamente. Si combinamos un mayor número de peticiones con una mayor concurrencia, el servidor experimenta una carga de trabajo considerablemente más alta en comparación con pruebas que solo aumentan una de estas variables. Además, puede provocar un aumento significativo en el tiempo de respuesta promedio. Las solicitudes pueden experimentar retardo debido a la competencia por recursos y la sobrecarga del servidor.

El análisis de los tiempos de respuesta en el primer escenario de conversión de archivos muestra una relación clara entre el aumento de la concurrencia y los tiempos de respuesta. A medida que se incrementa la concurrencia, se observa un aumento en los tiempos de respuesta. Esto se debe a que el servidor se ve sometido a una mayor presión, ya que debe administrar y procesar varias solicitudes simultáneamente. Esto puede implicar que la aplicación experimenta una leve degradación del rendimiento bajo cargas pesadas. A grandes rasgos, los tiempos de respuesta más largos pueden afectar negativamente la experiencia del usuario, lo que subraya la importancia de optimizar la aplicación para manejar eficazmente solicitudes concurrentes.

En el segundo escenario de conversión de múltiples archivos, las métricas clave son el tiempo de respuesta promedio y la utilización de recursos. El tiempo de respuesta promedio es generalmente bajo en todas las cargas de prueba, lo que indica una buena capacidad de respuesta del sistema incluso con una alta concurrencia. Esto es positivo y sugiere que el sistema puede manejar múltiples solicitudes concurrentes de manera eficiente.

## Uso CPU

Los picos observados en la gráfica se pueden dar por diversas razones. En primer lugar, la variabilidad en la carga de trabajo de la aplicación es un factor clave, ya que los momentos de aumento en las solicitudes de conversión de archivos de video pueden generar picos en el uso de CPU. También, la conversión simultánea de varios archivos de video influye en la demanda de recursos y puede causar fluctuaciones en el uso de la CPU.



## Instancia de la aplicación

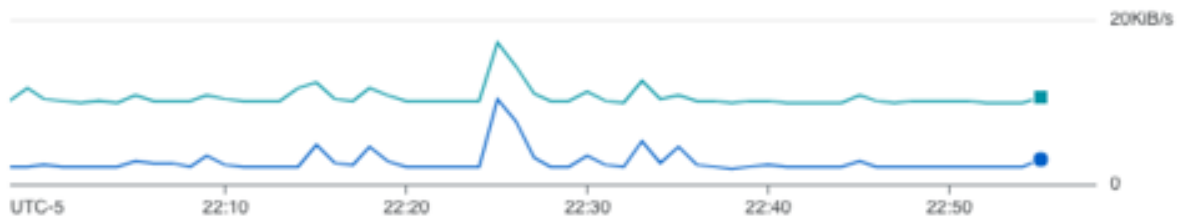
A medida que aumenta la concurrencia y el número de peticiones, es normal observar un aumento en el uso de CPU de la aplicación. Esto indica que la aplicación está procesando más solicitudes simultáneamente. Sin embargo, es importante monitorear el uso de CPU para asegurarse de que no esté alcanzando niveles críticos que puedan provocar una degradación del rendimiento.



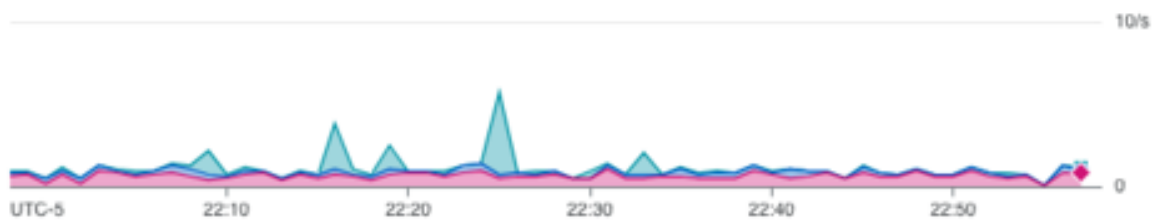
### Uso de CPU ?



### Tráfico de red ?



### Conexiones nuevas con VM/Recurso externo/Google ?



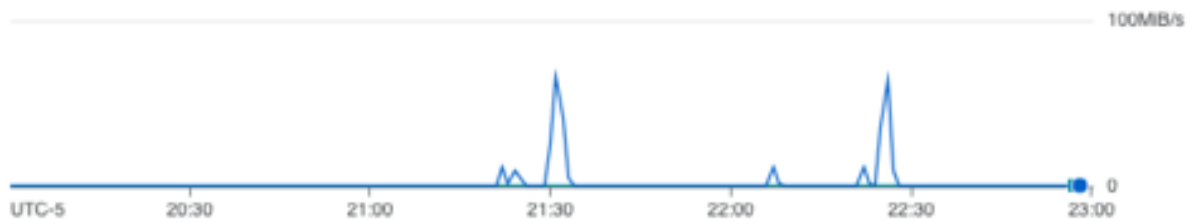
## Instancia del Worker

La instancia del worker es responsable de procesar las tareas en segundo plano, como la conversión de archivos de video. Un aumento en la utilización de recursos en esta instancia puede indicar que el servidor de tareas en segundo plano está experimentando una carga de trabajo más intensa.

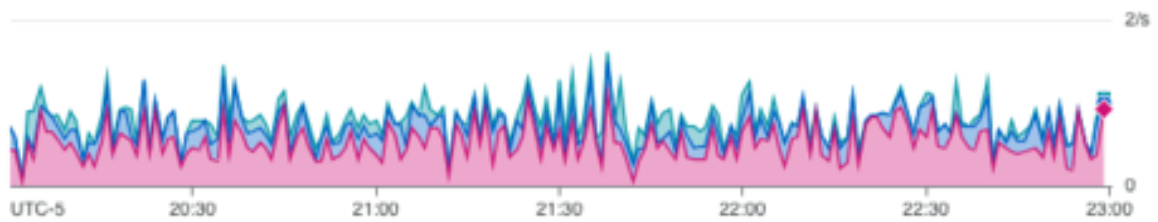
### Uso de CPU ?



### Tráfico de red ?



### Conexiones nuevas con VM/Recurso externo/Google ?



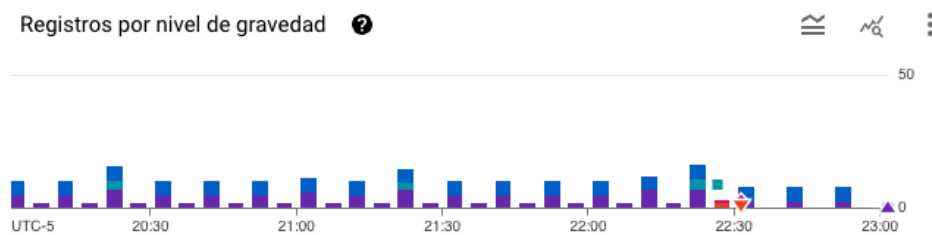
## Servidor de archivos

El uso de CPU en el servidor de archivos es importante para garantizar que el sistema de almacenamiento esté funcionando sin problemas. Cuando vemos que hay incrementos significativos,

indica que el servidor de archivos está experimentando una alta demanda de lectura o escritura de archivos.



Los registros por nivel de gravedad son esenciales para identificar problemas en la aplicación. Especialmente, los errores pueden señalar dificultades en el rendimiento o fallos en el sistema. Un error crítico a destacar es la falta de memoria, ya que este problema puede provocar ralentizaciones, esto se pudo producir por las cargas altas que experimentó el sistema.



Métricas de la base de datos

Todas las instancias > conversion-tool-pg-db  
conversion-tool-pg-db  
PostgreSQL 15



Todas las instancias > conversion-tool-pg-db  
conversion-tool-pg-db  
PostgreSQL 15



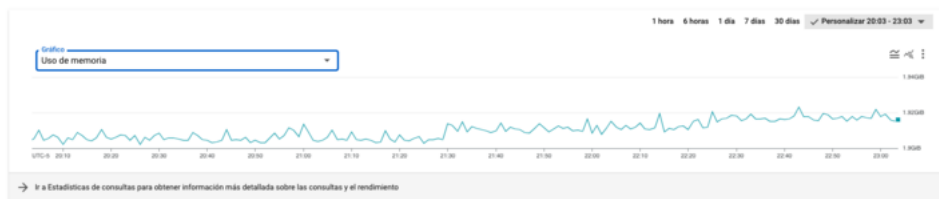
Todas las instancias > conversion-tool-pg-db  
conversion-tool-pg-db  
PostgreSQL 15



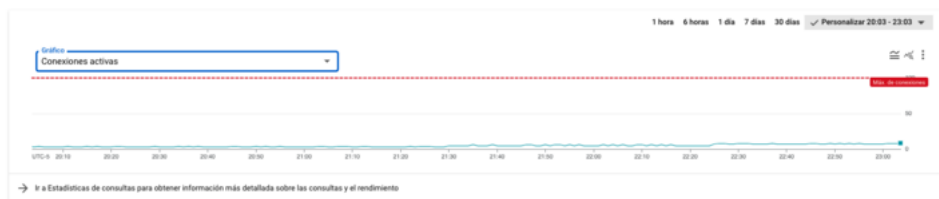
Todas las instancias > conversion-tool-pg-db  
conversion-tool-pg-db  
PostgreSQL 15



Todas las instancias > conversion-tool-pg-db  
conversion-tool-pg-db  
PostgreSQL 15



Todas las instancias > conversion-tool-pg-db  
conversion-tool-pg-db  
PostgreSQL 15



Conclusiones

- La aplicación experimentó un aumento en el uso de recursos de CPU y memoria a medida que se incrementaron las peticiones y la concurrencia. Esto indica la necesidad de una escalabilidad eficiente para manejar cargas elevadas sin comprometer el rendimiento.
- Las pruebas con Apache Benchmark permitieron evaluar la capacidad de procesamiento de la aplicación en diferentes situaciones
- El monitoreo detallado de métricas, como el uso de CPU, tráfico de red y conexiones, proporcionó información valiosa sobre el comportamiento de la aplicación bajo carga.
- Los registros por nivel de gravedad destacaron la importancia de abordar errores críticos, como la falta de memoria, que pueden afectar significativamente la estabilidad y el rendimiento de la aplicación.
- La integración de herramientas como Apache Benchmark y Gnuplot resultó útil para realizar pruebas de carga y analizar los resultados de manera efectiva, permitiendo la identificación de áreas de mejora y la optimización de la aplicación.