Integrantes:

Nicolas Saavedra Gonzalez

Luis Alfredo Castelblanco Quintero

Valeria Caro Ramirez

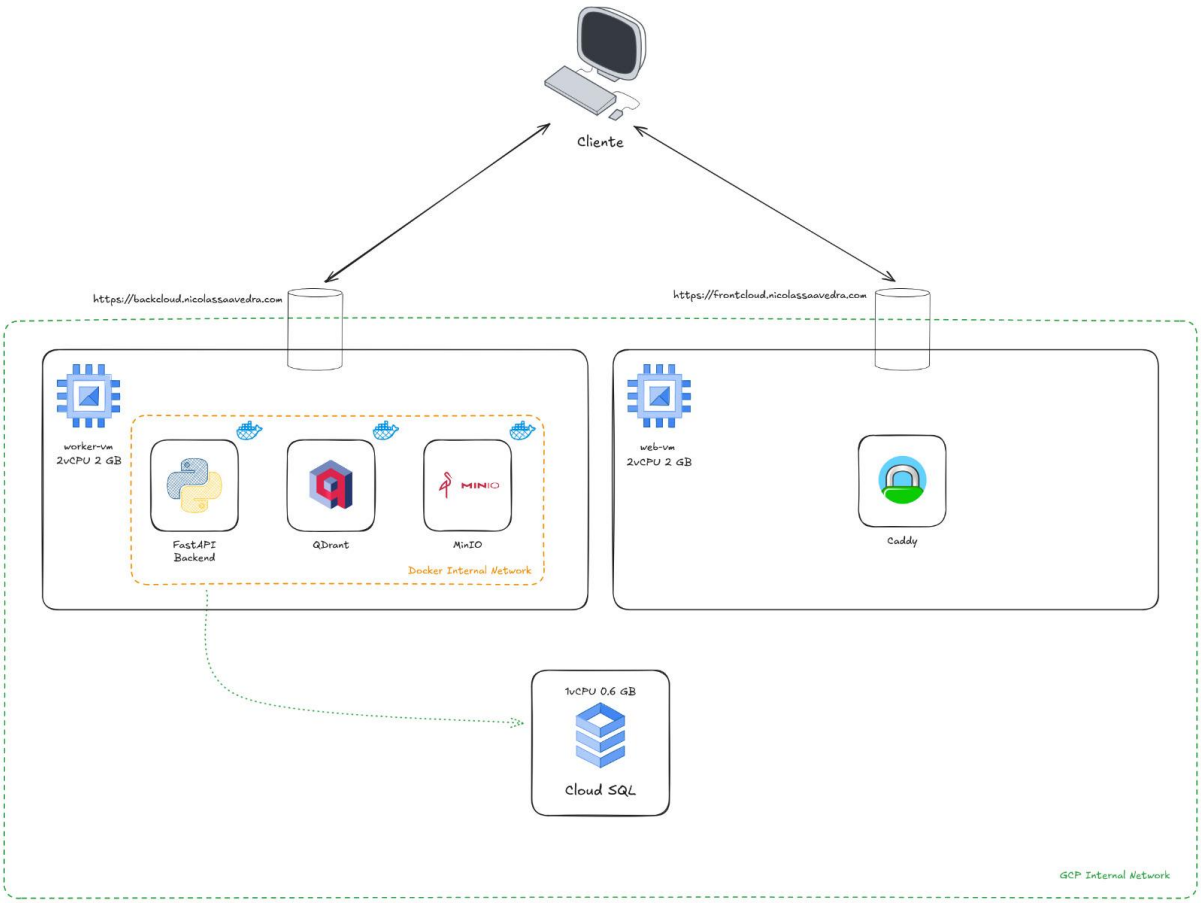
P2 – Análisis de capacidad

1. Configuración del entorno de pruebas

La aplicación está desplegada con la siguiente configuración:

- Web Server: Instancia de Compute Engine con 2 vCPU, 2 GiB RAM
- Worker: Instancia de Compute Engine con 2 vCPU, 2 GiB RAM
- Almacenamiento: En cuanto al almacenamiento, no se utilizó la instancia NFS debido a que la información no se almacenaba directamente en disco; además, ya se encontraba en uso una instancia S3 basada en Minio. Para la próxima entrega, se espera optimizar este servicio mediante su integración con Google Cloud Storage.
- Base de datos: Cloud SQL Postgres 15 SSD 10 GB

En el siguiente diagrama se presenta la arquitectura utilizada,



Este diagrama presenta una arquitectura basada en dos máquinas virtuales (VMs) alojadas en Google Cloud Platform (GCP):

- **worker-vm (2 vCPU, 2 GB)**:
  - Aloja tres contenedores Docker interconectados por una red Docker interna:
    - **FastAPI Backend**: Proporciona la lógica de negocio del sistema.
    - **QDrant**: Base de datos vectorial utilizada para almacenamiento y búsqueda eficiente.
    - **MinIO**: Solución de almacenamiento de objetos compatible con S3.
  - Conectado a una base de datos relacional administrada (**Cloud SQL**) que proporciona almacenamiento persistente adicional.
- **web-vm (2 vCPU, 2 GB)**:
  - Aloja un servidor web (**Caddy**) que gestiona y redirige tráfico HTTP/HTTPS hacia el backend alojado en la worker-vm.

Ambas máquinas virtuales se encuentran dentro de una red interna segura de GCP, permitiendo conexiones seguras y aisladas entre servicios, con acceso controlado desde el cliente externo mediante URLs específicas.

2. Herramientas de prueba

- K6: Herramienta de pruebas de carga
- Grafana/InfluxDB: Visualizacion de dashboards y almacenamiento de metricas

3. Escenarios

Escenario 1 USO REGISTER/LOGIN/CHAT/SESIONES

- **Descripción:** Este escenario simula el uso cotidiano de la aplicación RAG por parte de múltiples usuarios concurrentes, se evalúa el rendimiento de operaciones frecuentes como autenticación, gestión de chats y envío de mensajes con consultas simples que no requieren procesamiento intensivo. El objetivo es determinar la capacidad del sistema para manejar una carga moderada de usuarios realizando acciones básicas simultáneamente.

  p95 < 1000 ms
  Tasa de error < 4%
  Throughput > 2 req/s
  Estabilidad durante 3 min de carga constante
  Comportamiento bajo incremento de carga: 1-20 usuarios

- **Parámetros:**
  - Usuarios simulados: Incremento hasta 20 usuarios concurrentes
  - Duración: 5 minutos en la siguiente forma, 1 min rampa ascendente, 3 min carga sostenida, y 1 min rampa descendente
  - Tasa de solicitudes: Dependerá de la concurrencia de usuarios

- Patrones de tráfico: La simulamos con una distribucion realista con pausas aleatorias entre 1.3 segundos
- Operaciones evaluadas: Registro/login, obtención de perfil, listado de chats, creacion de chats, envió y recepción de mensajes simples

- **Resultados:**
  - **Globales**

Metrica	**Valor**
Solicitudes totales	884
Solicitudes por segundo promedio	2.89
Tiempo de respuesta promedio	108.85 ms
Tiempo de respuesta p95	166.97 ms
Tasa de error	3.39%
Ancho de banda	Recibido: 1.6 KB/s, Enviado: 981 B/s

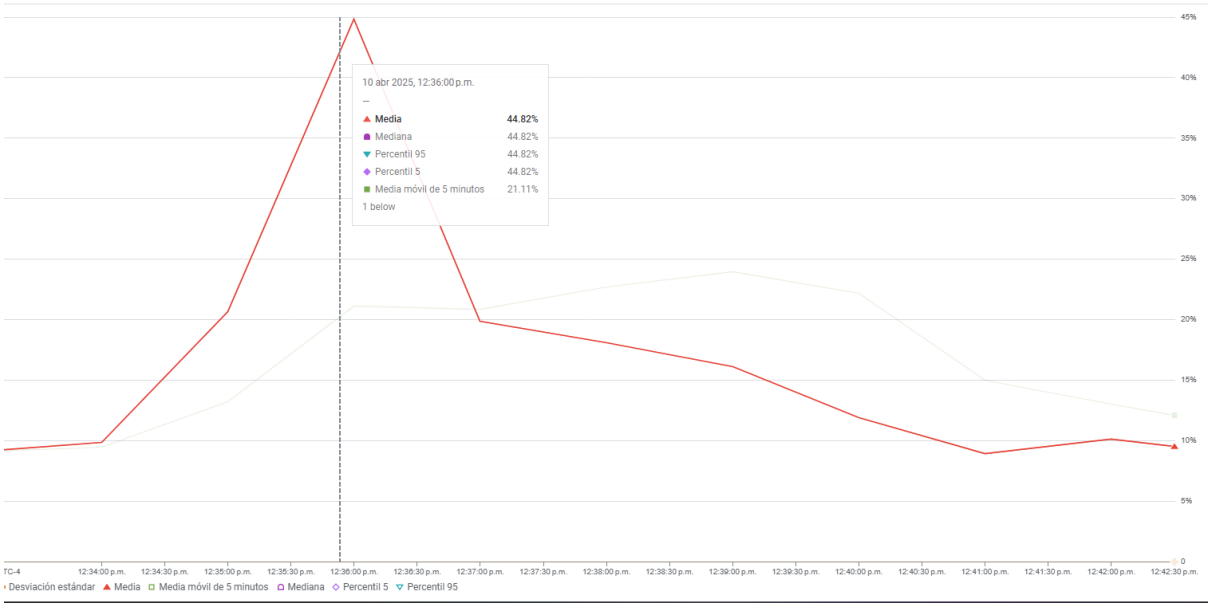
  - **Por EndPoint**

Endpoint	**Solicitudes**	**T. Promedio**	**p95**	**Tasa de error**
/auth/login	~20	427.17 ms	432.14 ms	~0%
/auth/register	~21	407.38 ms	431.01 ms	~5% (1 error de 21)
/users/me	~180	91.97 ms	96.1 ms	~0%
/users/{id}/chats (GET)	~150	92.41 ms	96.34 ms	~0%
/users/{id}/chats (POST)	~74	95.44 ms	99.13 ms	~0%
/chats/{id}/messages (GET)	~300	94.89 ms	98.13 ms	~0%
/chats/{id}/messages (POST)	~139	94.21 ms	100.1 ms	~0%

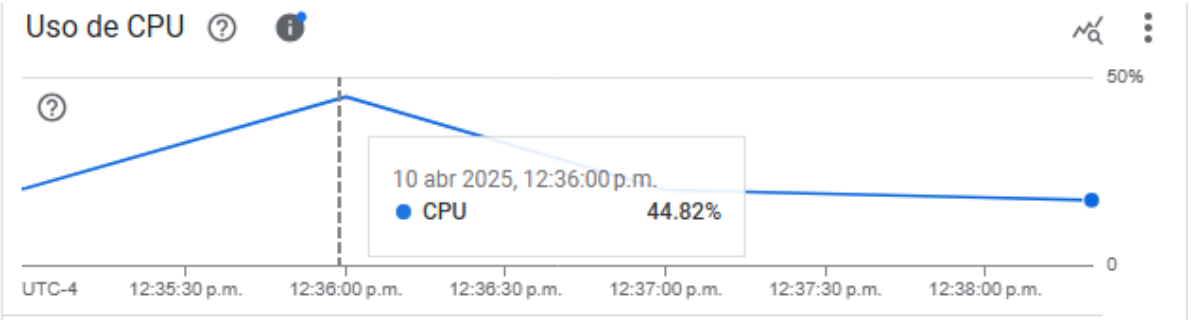
  - **Uso de recursos**

Componente	**CPU Promedio**	**CPU Pico**
Worker VM	11,22 %	44,82 %

  - **Graficas representativas**



CPU Promedio WorkerVM



CPU Pico Maximo WorkerVM

```
THRESHOLDS

  http_req_duration{name:/auth/login}
  ✓ 'p(95)<2000' p(95)=432.14ms

  http_req_duration{name:/auth/register}
  ✓ 'p(95)<2000' p(95)=431.01ms

  http_req_duration{name:/chats/{id}/messages (GET)}
  ✓ 'p(95)<2000' p(95)=98.13ms

  http_req_duration{name:/chats/{id}/messages (POST)}
  ✓ 'p(95)<2000' p(95)=100.1ms

  http_req_duration{name:/users/{id}/chats (GET)}
  ✓ 'p(95)<2000' p(95)=96.34ms

  http_req_duration{name:/users/{id}/chats (POST)}
  ✓ 'p(95)<2000' p(95)=99.13ms

  http_req_duration{name:/users/me}
  ✓ 'p(95)<2000' p(95)=96.1ms

  http_req_failed
  ✓ 'rate<0.055' rate=3.39%
```

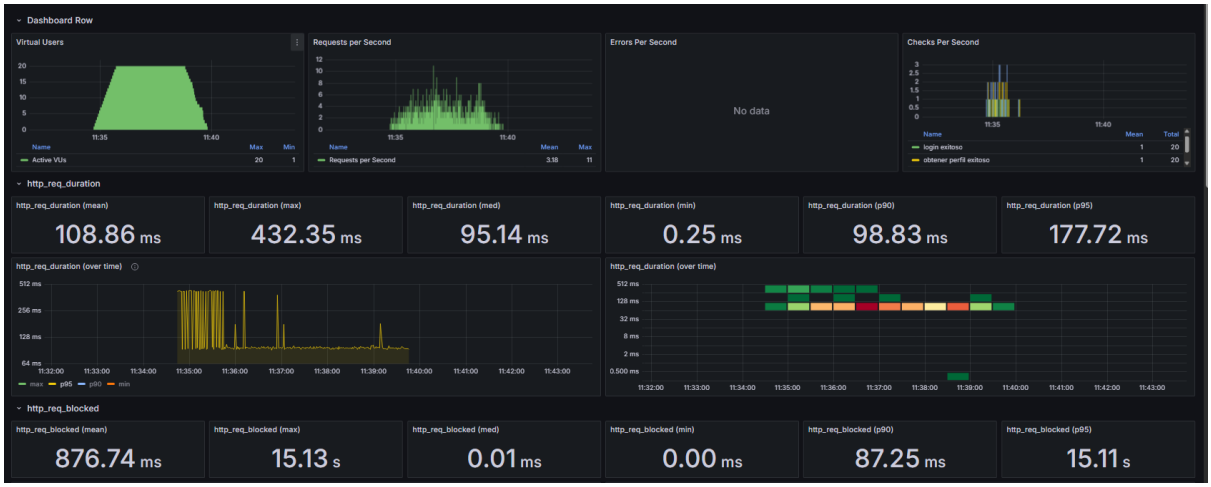
Resultados en k6 por EndPoint

```
TOTAL RESULTS

  checks_total.....................: 61     0.199267/s
  checks_succeeded.................: 98.36% 60 out of 61
  checks_failed....................: 1.63%  1 out of 61

  ✗ registro exitoso
    ↳  95% — ✓ 20 / ✗ 1
  ✓ login exitoso
  ✓ obtener perfil exitoso
```

Check en k6

```
   HTTP
   http_req_duration.....................................................: avg=108.85ms min=0s      med=95.13ms  max=432.35ms p(90)=98.8
1ms  p(95)=166.97ms
     { expected_response:true }.......................................: avg=111.51ms min=90.35ms  med=95.23ms  max=432.35ms p(90)=98.9
ms   p(95)=178.22ms
     { name:/auth/login }.............................................: avg=427.17ms min=422.59ms med=426.45ms max=432.35ms p(90)=430.
95ms p(95)=432.14ms
     { name:/auth/register }..........................................: avg=407.38ms min=0s      med=427.81ms max=431.65ms p(90)=430.
76ms p(95)=431.01ms
     { name:/chats/{id}/messages (GET) }..............................: avg=94.89ms  min=0s      med=95ms     max=387.02ms p(90)=97.2
4ms  p(95)=98.13ms
     { name:/chats/{id}/messages (POST) }.............................: avg=94.21ms  min=0s      med=97.12ms  max=102.17ms p(90)=99.1
7ms  p(95)=100.1ms
     { name:/users/{id}/chats (GET) }.................................: avg=92.41ms  min=0s      med=93.3ms   max=181.42ms p(90)=95.1
8ms  p(95)=96.34ms
     { name:/users/{id}/chats (POST) }................................: avg=95.44ms  min=0s      med=96.75ms  max=106.08ms p(90)=98.7
ms   p(95)=99.13ms
     { name:/users/me }...............................................: avg=91.97ms  min=0s      med=93.36ms  max=101.22ms p(90)=95.2
8ms  p(95)=96.1ms
   http_req_failed.....................................................: 3.39%  30 out of 884
   http_reqs...........................................................: 884    2.887733/s

   EXECUTION
   iteration_duration..................................................: avg=5.42s    min=2s      med=4.09s    max=35s      p(90)=6.18
s    p(95)=19.22s
   iterations..........................................................: 909    2.9694/s
   vus.................................................................: 1      min=1        max=20
   vus_max.............................................................: 20     min=20       max=20
```

Metricas por endpoint y globales



Dashboard de resultados en Grafana

Escenario 1.1 Punto Inflexion

Descripción: Este escenario busca identificar el punto de inflexión exacto donde el sistema comienza a mostrar signos de degradación mientras aún cumple con los umbrales de rendimiento aceptables. Se incrementa gradualmente la carga para determinar la capacidad óptima del sistema, ese balance entre máximo throughput y tiempos de respuesta dentro de parámetros admisibles.

p95 < 1000 ms
Tasa de error < 4%
Throughput óptimo sin degradar tiempos de respuesta
Identificación del punto de inflexión: 30-50 usuarios concurrentes

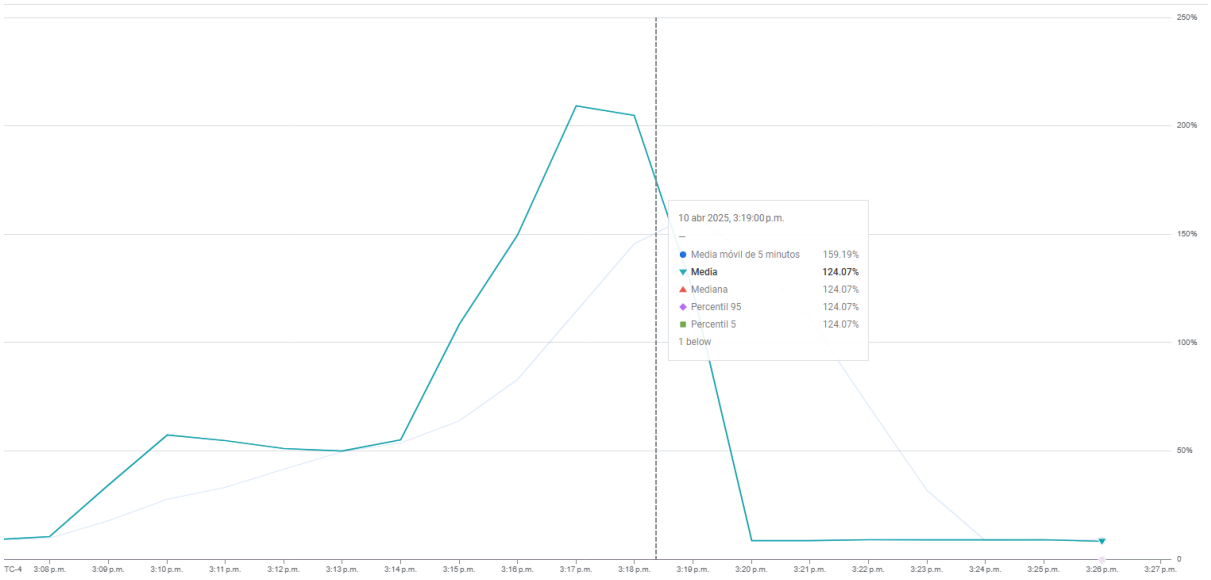
Parámetros:

- Usuarios simulados: Incremento escalonado (30 → 35 → 40 → 45 → 50 usuarios)
- Duración: 10 minutos (2 min por cada escalón de carga)
- Tasa de solicitudes: Variable según la concurrencia de usuarios
- Patrones de tráfico: Pausas moderadas entre operaciones (0.5-1.5 segundos)
- Operaciones evaluadas: Mezcla balanceada de operaciones de lectura y escritura, incluyendo consultas de complejidad media al LLM

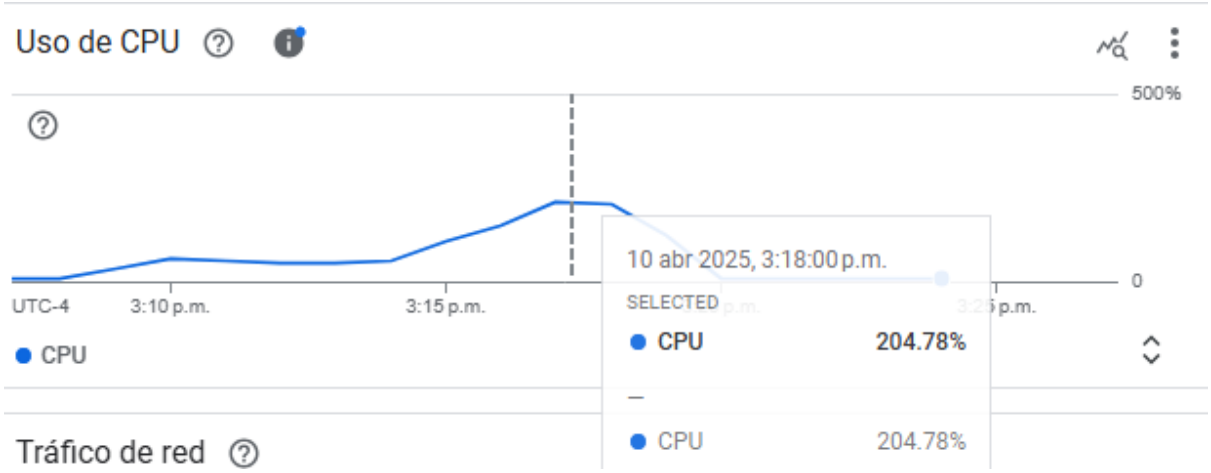
Metrica	**Valor**
Solicitudes totales	7857
Solicitudes por segundo promedio	72.64
Tiempo de respuesta promedio	105.63 ms
Tiempo de respuesta p95	116.72 ms
Tasa de error	9.46%
Ancho de banda	Recibido: 33 KB/s, Enviado: 4.8 B/s

Endpoint	**Solicitudes**	**T. Promedio**	**p95**	**Tasa de error**
/auth/login	~50	465.75 ms	701.61 ms	~0%
/auth/register	~53	414.89 ms	485.49 ms	~8% (4 de ~53)
/users/me	~1200	94.72 ms	103.73 ms	~10%
/users/{id}/chats (GET)	~1050	108.91 ms	103.34 ms	~10%
/users/{id}/chats (POST)	~1570	96.38 ms	105.48 ms	~10%
/chats/{id}/messages (GET)	~2750	104.71 ms	181.06 ms	~10%
/chats/{id}/messages (POST)	~2350	98.24 ms	109.24 ms	~10%

Componente	**CPU Promedio**	**CPU Pico**
Worker VM	124,07 %	204,78 %



CPU Promedio WorkerVM



CPU Promedio WorkerVM

```
THRESHOLDS

  http_req_duration{name:/auth/login}
  ✓ 'p(95)<1000' p(95)=701.61ms

  http_req_duration{name:/auth/register}
  ✓ 'p(95)<1000' p(95)=485.49ms

  http_req_duration{name:/chats/{id}/messages (GET)}
  ✓ 'p(95)<1000' p(95)=181.06ms

  http_req_duration{name:/chats/{id}/messages (POST)}
  ✓ 'p(95)<1000' p(95)=109.24ms

  http_req_duration{name:/users/{id}/chats (GET)}
  ✓ 'p(95)<1000' p(95)=103.34ms

  http_req_duration{name:/users/{id}/chats (POST)}
  ✓ 'p(95)<1000' p(95)=105.48ms

  http_req_duration{name:/users/me}
  ✓ 'p(95)<1000' p(95)=103.73ms

  http_req_failed
  ✗ 'rate<0.04' rate=9.46%
```

Resultados en k6 por endpoint

```
TOTAL RESULTS

checks_total.....................: 151    0.242836/s
checks_succeeded...................: 97.35% 147 out of 151
checks_failed.....................: 2.64%  4 out of 151

✗ registro exitoso
  ↳  92% — ✓ 49 / ✗ 4
✓ login exitoso
✓ obtener perfil exitoso
```

Check en k6

Dashboard de los resultados en Grafana

Escenario 2 USO DEL RAG

- **Descripción:**

  Este escenario busca identificar el punto de inflexión específico donde el sistema comienza a degradarse al procesar operaciones intensivas de RAG . Se evalúa la capacidad del Worker para manejar la indexación de documentos, generación de embeddings y consultas contextuales bajo carga creciente.

p90 < 8000 ms para operaciones generales
p90 < 5000 ms para generación de respuestas LLM
Tasa de error < 2%
Identificación del número óptimo de usuarios RAG simultáneos

- **Objetivo**
  Identificación del número óptimo de usuarios RAG simultáneos

- **Parámetros:**
  - Usuarios simulados: Incremento escalonado (5 → 10 → 15 usuarios)
  - Duración: 10 minutos distribuidos en escalones de carga
  - Tasa de solicitudes: Variable según la concurrencia de usuarios
  - Patrones de tráfico: Pausas realistas entre operaciones (2-6 segundos)
  - Operaciones evaluadas: Carga de documentos, indexación, consultas RAG y generación de respuestas contextuales

- **Resultados:**
  - **Globales**

Metrica	Valor
Solicitudes totales	648
Solicitudes por segundo promedio	2.09
Tiempo de respuesta promedio	2.8s
Tiempo de respuesta p95	6.65s
Tiempo de respuesta p90	5.71s
Tasa de error	0.15%
Ancho de banda	Recibido: 1.2 KB/s, Enviado: 1.0 B/s

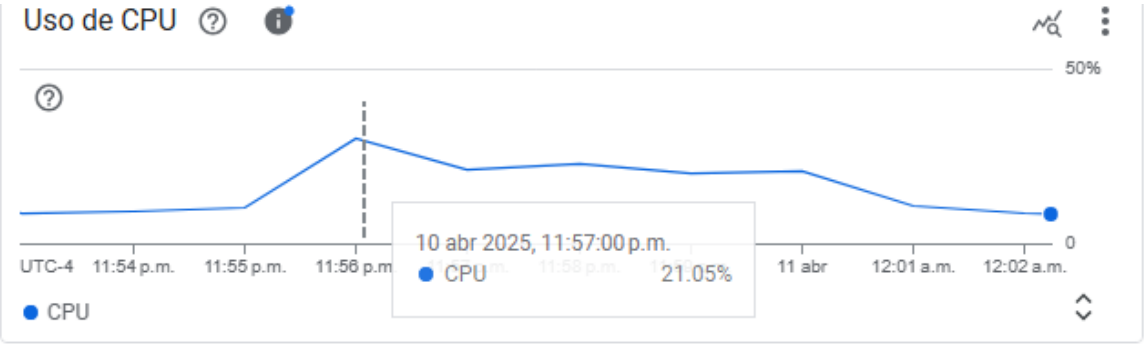
  - **Por EndPoint**

Endpoint	Solicitudes	T. Promedio	p90	p95	Tasa de error
/auth/login	~12	3.77s	9.57s	9.66s	~0%
/auth/register	~12	1.97s	4.48s	5.33s	~0%
/users/{id}/documents (POST)	~36	4.26s	10.1s	6.67s	0.34(1 de 288)
/llm/generate	~288	3.4s	6.32s	10.77s	~0%

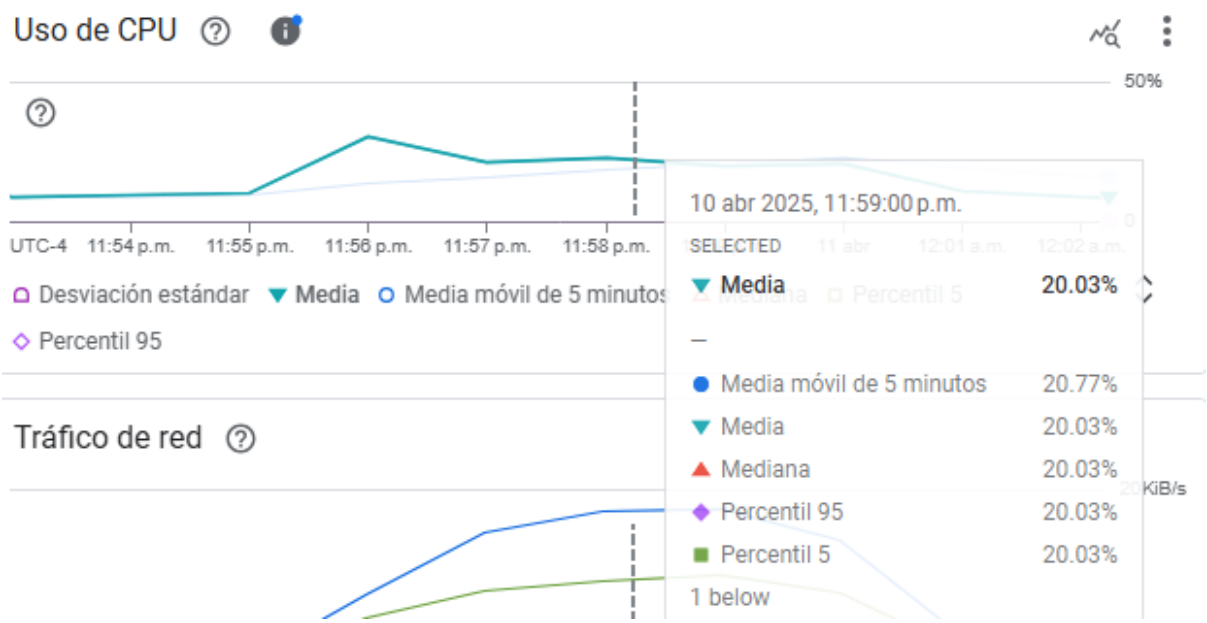
  - **Uso de recursos**

Componente	CPU Promedio	CPU Pico
Worker VM	124,07 %	21,08 %

  - **Graficas representativas**



CPU Uso maximo en WorkerVM



CPU Uso Promedio en WorkerVM

```
THRESHOLDS

http_req_duration{name:/auth/login}
X 'p(90)<8000' p(90)=9.57s

http_req_duration{name:/auth/register}
✓ 'p(90)<8000' p(90)=4.48s

http_req_duration{name:/llm/generate}
X 'p(90)<5000' p(90)=6.32s

http_req_duration{name:/users/{id}/documents (POST)}
X 'p(90)<8000' p(90)=10.1s

http_req_failed
✓ 'rate<0.05' rate=0.15%
```

Resultados por endpoint en k6

```
TOTAL RESULTS

  checks_total.....: 324    1.043885/s
  checks_succeeded.....: 99.69% 323 out of 324
  checks_failed.....: 0.30%  1 out of 324

  ✓ registro exitoso
  ✓ login exitoso
  ✓ obtener perfil exitoso
  X generación de respuesta RAG exitosa
15%  1 out of 648
  http_reqs.....: 648    2.087769/s

  EXECUTION
  iteration_duration.....: avg=8.89s min=1.27s    med=8.39s    max=52.15s p(90)=12.49s p(95)=14.45s
  iterations.....: 300    0.96656/s
  vus.....: 1      min=1       max=12
  vus_max.....: 12     min=12      max=12

  NETWORK
  data_received.....: 387 kB 1.2 kB/s
  data_sent.....: 315 kB 1.0 kB/s
```

Resultados globales y checks en k6

Dashboards de métricas en Grafana

4. Análisis:

Escenario 1:

Con estos datos de la prueba de estrés se muestra un comportamiento adecuado del sistema bajo una carga moderada de 20 usuarios concurrentes.

En el rendimiento general obtuvimos 108.85 ms que es un excelente resultado en comparacion al umbral establecido de 1000 ms, también podemos ver que en el percentil 95 obtuvimos 166.97 ms en los tiempos de respuesta que se sigue manteniendo muy bajo, indicándonos un rendimiento bastante estable y consistente para la mayoría de las operaciones

En los endpoints de autenticación como /auth/login y /auth/register muestran tiempos de respuesta más altos, esto se puede deber al proceso de obtención, generación y comprobación del token de JWT, por otro lado, operaciones de consulta y manipulación de datos /users/me, chats, messages tienen un rendimiento muy consistente, con tiempos de respuesta alrededor de 95ms y poca variación entre ellos y con esto podemos aclarecer y confirmar que todo los endpoints cumplieron ampliamente con el umbral de rendimiento p95>1000 ms.

En la tasa de error global que quedo en 3.39 está por debajo del umbral que es del 4%, está cerca entonces sería importante abordar los problemas de estabilidad en especial en el endpoint de /auth/register con aproximadamente un

5% de tasa de error, posiblemente debido restricciones en las validaciones de los datos de los nuevos usuarios generados o como pudimos visualizar en las pruebas un problema de EOF.

El sistema proceso un promedio de 2.89 solicitudes por segundo, lo que podemos traducir a aproximadamente 173 solicitudes por minutos, considerando esto y que los tiempos de respuesta se mantienen muy por debajo de los umbrales, hay bastante margen para incrementar la carga de usuarios antes de observar degradación en el rendimiento

Escenario 1.1

El escenario 1.1 identificamos el punto donde el sistema comienza a degradarse bajo carga incremental de 30 a 50 usuarios durante 10 minutos. Los resultados mostraron tiempos de respuesta excepcionalmente bajos promedio 105.63ms, p95 116.72ms incluso bajo carga máxima, significativamente por debajo del umbral de 1000 ms pero sabemos de la degradación del uso de la CPU y porcentaje de errores.

El sistema alcanzó un throughput de 12.64 solicitudes por segundo, pero con una tasa de error del 9.46%, superando el umbral aceptable del 4%. Los endpoints de autenticación presentaron los mayores tiempos de respuesta 465.75ms para login, mientras que las operaciones regulares post del llm y documento mantuvieron tiempos bajos.

Aun asi algo a ver fue la utilización de CPU del Worker, con un promedio de 124.07% y picos de 204.78%, operando consistentemente por encima de su capacidad óptima. Esta sobrecarga explica directamente la elevada tasa de error, ya que el sistema rechaza conexiones al saturarse.

El punto de inflexión se sitúa entre 35-40 usuarios concurrentes, donde el sistema mantendría tiempos aceptables con errores por debajo del 4%.

Escenario 2

El escenario 2 evaluó el comportamiento del sistema durante operaciones intensivas de RAG , con el objetivo específico de identificar el número óptimo de usuarios simultáneos que pueden realizar estas operaciones antes de que el

rendimiento se degrade significativamente. La prueba utilizó un incremento escalonado de usuarios (5→10→15) durante 10 o (5 en el video) minutos para medir el impacto en el rendimiento del sistema.

Los resultados muestran que el procesamiento RAG representa una carga bastante mayor para el sistema que las operaciones básicas. Con solo 15 usuarios concurrentes, el tiempo de respuesta promedio alcanzó 2.8s, con un p90 de 5.71s y p95 de 6.65s y aunque estos valores están dentro del umbral general de 8000ms para operaciones básicas, algunos endpoints específicos superaron sus umbrales designados.

El análisis por endpoint revela informacion muy importante como en el endpoint de carga de documentos (`/users/{id}/documents (POST)`) mostró el peor rendimiento con un tiempo promedio de 4.26s y un p90 de 10.1s, superando el umbral de 8000ms, esto nos indica que la generación de embeddings y la indexación de documentos forman el principal cuello de botella del sistema luego el endpoint de generación LLM (`/llm/generate`) también superó su umbral específico con un p90 de 6.32s, por encima del límite de 5000ms establecido, por otro lado, incluso los endpoints de autenticación mostraron tiempos elevados, con `/auth/login` alcanzando un p90 de 9.57s. Esto nos dice que la contención de recursos durante operaciones intensivas de RAG afecta el rendimiento de todo el sistema, no solo de los componentes directamente relacionados con el procesamiento RAG.

Respecto a la tasa de error, el sistema mantiene una tasa muy baja de 0.15%, muy por debajo del umbral del 2%. Esto indica que el sistema prioriza completar todas las solicitudes a costa de tiempos de respuesta más largos, en lugar de rechazar conexiones cuando está bajo presión lo cual bajo nuestra percepción nos parece importante en estos tipos de sistemas RAG.

Los datos de uso de recursos muestran un comportamiento interesante: mientras que el CPU promedio del Worker (124.07%) indica una utilización por encima de su capacidad nominal, el pico de CPU es bajo (21.08%). Esta discrepancia sugiere posibles problemas en la medición o que otros recursos como memoria o I/O podrían ser unos limitantes

Basándonos en estos resultados, podemos concluir que el número óptimo de usuarios RAG simultáneos es aproximadamente 8-9 antes de que el sistema comience a mostrar degradación significativa. A partir de 10 usuarios, los endpoints críticos superan sus umbrales de rendimiento, aunque el sistema sigue procesando solicitudes con una tasa de error mínima.

5. Limitaciones actuales

El sistema presenta restricciones operativas bien definidas a través de los distintos escenarios de prueba. En condiciones básicas (Escenario 1), aunque mantiene excelentes tiempos de respuesta p95 de 166.97 ms, muestra una tasa de error cercana al límite aceptable 3.39% vs 4% umbral. Al incrementar la carga (Escenario 1.1), se identifica un punto de inflexión entre 35-40 usuarios concurrentes, donde la tasa de error supera el umbral del 4%, llegando al 9.46% con 50 usuarios, mientras que el Worker opera consistentemente por encima de su capacidad óptima CPU promedio 124.07%, picos de 204.78%. Para operaciones intensivas de RAG (Escenario 2), el sistema solo puede manejar eficientemente 8-9 usuarios simultáneos antes de degradarse significativamente. Los principales cuellos de botella se encuentran en el procesamiento de documentos /users/{id}/documents POST con p90 de 10.1s y generación de respuestas LLM (p90 de 6.32, ambos superando sus respectivos umbrales. La contención de recursos durante operaciones RAG afecta incluso a componentes no relacionados, como la autenticación, donde /auth/login alcanza un p90 de 9.57s. Existe una notable diferencia entre la arquitectura para operaciones básicas prioriza velocidad rechazando conexiones y operaciones RAG prioriza completar solicitudes a costa de tiempos más largos.

6. Recomendaciones futuras

Para superar las limitaciones anteriormente identificadas y mejorar el rendimiento general del sistema, pensamos en implementar un conjunto de medidas técnicas y arquitectónicas de computacion en la nube. En primer lugar, es fundamental optimizar el endpoint de registro (/auth/register) para reducir la tasa de error del 5% observada, posiblemente mejorando las validaciones de datos y solucionando los problemas EOF detectados en la terminal de K6. Para el punto de inflexión identificado (35-40 usuarios), pensamos el escalamiento horizontal del Worker, aumentando sus instancias para manejar mejor la sobrecarga observada y reducir el rechazo de conexiones bajo carga elevada. Respecto a las operaciones RAG, se debe priorizar la optimización del proceso de generación de embeddings e indexación de documentos, que constituyen el principal cuello de botella del sistema. La implementación de un sistema de colas permitiría gestionar picos de demanda sin degradar la experiencia de usuario, especialmente para operaciones intensivas. A mediano plazo, considerar una arquitectura distribuida con múltiples workers y un sistema de balanceo de carga mejoraría la capacidad general. También sería beneficioso para nuestro proyecto implementar un mecanismo de caché para respuestas frecuentes, reduciendo la necesidad de procesamiento completo para consultas similares y mejorando la eficiencia global del sistema.