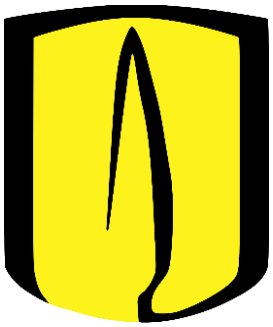
DESARROLLO DE SOLUCIONES CLOUD



Proyecto 1 – Plan de pruebas de carga

Integrantes:

Esteban Caicedo Graciano

Steban Nicolas Tibata Castañeda

Nicolas Miguel Murillo Cristancho

1. ¿Cuál es su entorno de prueba?

El entorno de prueba usará GCP Como proveedor, el cual fue elegido por su disponibilidad, a diferencia de AWS que cuenta con limitaciones de disponibilidad dadas por el carácter académico de la cuenta. Las limitaciones de GCP vienen dadas por las restricciones de presupuesto. El entorno físico constará de Vms interconectadas por una vpc, se planea administrar estas vms asignando diferentes responsabilidades a cada una, es decir, cada máquina tendrá una parte diferente del software, como la capa web (front), la capa del worker (RAGs) y las capas de almacenamiento, se planea que cada máquina sea de gama media-baja (mayores descripciones técnicas se describen en el punto 6).

2. ¿Cuáles son los criterios de aceptación?

Objetivos y restricciones

Tiempo de respuesta:

Los tiempos de respuesta aceptables serán considerablemente menores que los que obtendría un usuario promedio al usar directamente los servicios del proveedor, debido a nuestra administración adicional de RAGs y el presupuesto para aprovisionar los recursos de las máquinas virtuales.

Subida y análisis de documentos:

- El tiempo para subir un documento debe ser <10s por cada solicitud para un usuario concurrente, en al menos el 90% de los casos
- El tiempo para subir un documento debe ser <15s por cada solicitud para 100 usuarios concurrentes, en al menos el 90% de los casos

Procesamiento de documentos en cola (tiempo de respuesta a un prompt):

- En promedio, el tiempo de respuesta debe ser <25s por cada usuario para 100 usuarios concurrentes
- En promedio, el tiempo de respuesta debe ser <15s para un usuario concurrente

Rendimiento:

- El sistema debe procesar al menos 100 prompts por minuto manteniendo el tiempo de respuesta promedio por debajo de 25 segundos y cumpliendo satisfactoriamente el 95% de las solicitudes.
- El sistema debe procesar al menos la carga de 100 documentos por minuto manteniendo el tiempo de respuesta promedio por debajo de 15 segundos y cumpliendo satisfactoriamente el 98% de las solicitudes.

Utilización de recursos:

- <80% de uso de memoria total de todas las máquinas usadas para 100 usuarios concurrentes, realizando tareas de prompt completion
- <50% de uso de cpu de una sola máquina para un usuario concurrente, realizando tareas de prompt completion.
- <90% de uso de memoria de cada una de las máquinas usadas para 100 usuarios concurrentes, realizando tareas de subida y procesamiento de archivos
- <80% de uso de cpu de una sola máquina para un usuario concurrente, realizando tareas de prompt completion.

Restricciones adicionales:

Impuestas con el fin de prevenir casos excepcionales que puedan afectar de manera extraordinaria el tiempo de respuesta, rendimiento o uso de recursos:

- Aunque es probable soportar más de 100 usuarios concurrentes, no se ve viable garantizar estos niveles de concurrencia de manera constante en el tiempo, sobre todo en las primeras fases del proyecto
- Con el fin de prevenir uso desigual de acceso a los recursos por parte de los usuarios, se impuso una restricción de tamaño a los archivos de 10mb
- El límite de palabras que se puede generar en una respuesta de parte del llm viene condicionado por el proveedor y su API, usualmente 96.000 palabras.

Criterios de éxito adicionales (No medibles en el alcance del curso o por otras limitaciones)

- Asegurar una escalabilidad automática de 5 a 500 usuarios concurrentes por minuto en semanas de parciales, cumpliendo satisfactoriamente el 95% de las solicitudes de prompt completion.
- Asegurar una disponibilidad del 99% con tiempos de fallas distribuidos con un máximo del 10% del tiempo de falla total por año para cada fallo.
- Asegurar un índice de satisfacción del 90% para las respuestas de los llm, soportados por un 90% de precisión de los RAGs y chunks al momento de seleccionar el contexto necesario.

3. ¿Cuáles son los escenarios de prueba?

Los escenarios de prueba para RAG_SaaS se centran en los procesos clave de la aplicación: la carga de documentos y la recuperación de información mediante

consultas a la base de conocimientos. Estas pruebas permitirán evaluar el rendimiento del sistema, la escalabilidad y la estabilidad bajo carga.

Escenarios clave

1. Carga y procesamiento de documentos
    a. Un usuario carga un documento (PDF, DOCX, TXT o MD) en la plataforma. El sistema extrae el contenido y lo almacena en la base de datos, generando embeddings con OpenAI para futuras consultas.
    b. Objetivos de prueba
        i. Medir el tiempo de carga y procesamiento por documento.
        ii. Evaluar el impacto de múltiples usuarios concurrentes subiendo documentos.
        iii. Monitorear el uso de CPU, memoria y almacenamiento durante la extracción de contenido.
    c. Métricas a recolectar
        i. Tiempo de carga y procesamiento (segundos/documento).
        ii. Tasa de éxito de subida de documentos (porcentaje de documentos procesados sin error).
        iii. Uso de CPU/RAM en el servidor backend bajo diferentes cargas de usuario.
2. Consultas a la base de conocimiento
    a. Los usuarios realizan preguntas sobre los documentos cargados, y el sistema usa embeddings y GPT-4 para generar respuestas basadas en el contenido.
    b. Objetivos de prueba
        i. Medir la latencia de respuesta para consultas individuales y concurrentes.
        ii. Analizar la estabilidad del sistema bajo alta concurrencia.
        iii. Verificar el correcto funcionamiento de la recuperación de información.
    c. Métricas a recolectar
        i. Tiempo de respuesta promedio (segundos por consulta).
        ii. Tasa de éxito de consultas (porcentaje de respuestas generadas sin error).
        iii. Uso de CPU/RAM del backend y de la API de OpenAI.
3. Escalabilidad y estabilidad del sistema
    a. Simulación de cargas progresivas para determinar el punto de saturación del sistema.
    b. Objetivos de prueba

i. Determinar la cantidad máxima de usuarios concurrentes que puede soportar la aplicación sin degradación del rendimiento.
ii. Identificar el comportamiento del sistema ante un aumento progresivo de carga.

c. Métricas a recolectar
i. Número de usuarios concurrentes antes de degradación.
ii. Uso de recursos en cada nivel de carga.
iii. Tiempo de respuesta bajo diferentes cargas concurrentes.

4. ¿Cuáles son los parámetros de configuración?

Para garantizar pruebas eficientes, se deben configurar correctamente el entorno de prueba, las herramientas y los recursos necesarios. Se establecen los siguientes parámetros:

Herramientas de prueba

- Apache JMeter. Simulación de múltiples usuarios concurrentes.
- Django Debug Toolbar. Monitoreo del rendimiento en el backend.
- Cloud logging. Recopila logs de los servicios en GCP, permitiendo analizar fallos y cuellos de botella.
- Cloud Monitoring. Permite supervisar en tiempo real el rendimiento de instancias, bases de datos, redes y otros recursos en GCP.

Configuración del entorno de prueba en GCP

El entorno de pruebas en GCP contará con varias máquinas virtuales interconectadas a través de una VPC privada. Se distribuirán los componentes de la siguiente manera:

1. Servidor Backend (Django)
   a. VM: e2-standard-4 (4 vCPUs, 16GB RAM)
   b. SO: Ubuntu 22.04 LTS
   c. Software: Django 4.2.10
   d. Puerto: 8000 (servidor de desarrollo de Django)
2. Base de Datos (MySQL 8.0)
   a. VM: e2-standard-4 (4 vCPUs, 16GB RAM)
   b. Almacenamiento: SSD 100GB
   c. Puerto: 3306 (MySQL)
3. Servidor de pruebas de carga (Apache JMeter)
   a. AWS EC2 Learner Lab
   b. Instancia: t3.xlarge (4 vCPUs, 16GB RAM)

c. Disco: 100GB SSD
d. Pruebas en paralelo con hasta 100 usuarios concurrentes

Parámetros de pruebas de carga

Para Apache JMeter y Locust, se configuran los siguientes parámetros:

1. Carga de documentos
    a. Usuarios simulados: 10, 50, 100, 200 concurrentes
    b. Tamaño de documentos: 100KB, 1MB, 5MB
    c. Tipo de archivos: PDF y DOCX
    d. Umbral de error aceptable: Máximo 5% de fallos
2. Consultas a la base de conocimientos
    a. Usuarios concurrentes: 10, 50, 100
    b. Tipo de consulta: preguntas simples vs preguntas complejas
    c. Tiempo de respuesta objetivo: < 3s en condiciones normales, < 5s en alta carga

Métricas clave a monitorear

- Latencia del backend (Django ORM y procesamiento de consultas)
- Uso de CPU/RAM en cada VM
- Tiempos de respuesta de API y base de datos
- Tasa de error y tiempo de recuperación ante fallos

5. Identifique dos escenarios clave para llevar a cabo pruebas de carga en las próximas entregas.

Dentro de los escenarios clave para llevar a cabo la aplicación existen 2 escenarios críticos del usuario, relacionados con el cargue de archivos dentro de la aplicación y el procesamiento por lotes, dichos escenarios se describen a continuación:

1. **Escenario relacionado con la capa web**:

- **Subida y análisis de documentos**: Involucra la acción del usuario de subir un documento para que sea analizado. La prueba de carga debe medir:
    - Tiempo de carga de los documentos desde diversos dispositivos y diferentes tipos de archivo.
    - Respuesta del sistema en caso de múltiples usuarios subiendo documentos simultáneamente.
    - Comportamiento de la interfaz web bajo alta concurrencia y con distintos tamaños de archivo (dentro del límite de carga).

El **objetivo** es garantizar que los tiempos de respuesta sean aceptables y que el sistema se mantenga estable incluso bajo carga máxima o bajo uso simultáneo.

2. **Escenario relacionado con la capa de procesamiento por lotes**:

- **Procesamiento de documentos en cola**: Esta capa gestiona la interpretación de los documentos subidos, especialmente cuando múltiples usuarios están interactuando con la aplicación. Aquí las pruebas deben evaluar:
    - Tiempo de procesamiento promedio por documento bajo diferentes volúmenes de carga.
    - Manejo adecuado de cuellos de botella cuando hay picos en la cantidad de documentos.
    - Monitoreo del consumo de recursos (CPU, memoria) y la tolerancia del sistema para mantener diferentes procesamientos por lotes de usuarios.

El **objetivo** consiste en asegurar que el procesamiento por lotes funcione de manera eficiente, evitando retrasos significativos en la carga de archivos.

A pesar de que ambos escenarios sean similares, el primero hace referencia a la interacción directa del usuario con la interfaz de la aplicación, donde aquí las pruebas miden factores como la latencia, errores o tasa de éxito. Por el otro lado, el segundo escenario se centra en el procesamiento interno realizado por la aplicación para analizar y extraer información de los documentos, donde se evalúa más que todo el rendimiento del backend.

En ambos escenarios se tendrán en cuenta diferentes SLIs y SLOs. Para el primer escenario contamos con SLIs como latencia de carga, tasa de éxito de subida de documento y latencia de respuesta, idealmente se quiere un SLO con una tasa de éxito en la subida de documentos sobre un 90% en condiciones de alta concurrencia y un 95% en condiciones normales. En el segundo escenario contamos con SLIs como tiempo de procesamiento promedio por documento, tasa de errores en el procesamiento y disponibilidad de servicio, con un SLO donde la tasa de errores debe mantenerse por debajo del 3% y una disponibilidad de servicios del 95%.

6. Indique la herramienta que va a emplear para las pruebas de carga y describa las capacidades de la infraestructura requerida para su ejecución.

Le herramienta a utilizar será Apache JMeter ya que es versatil por la capacidad de simular múltiples usuarios concurrentes y por la variedad de protocolos, además de que es escalable para pruebas distribuidas. DIcha herramienta requiere ciertas características específicas en su infraestructura, como una instancia con CPU de 4-8

vCPUs, 16GB de memoria RAM y disco SSD con 100GB. Instancias con dichas características en GCP pueden ser como e2-standard-4 o e2-standard-8 dependiendo de la cantidad de usuarios simulados de manera simultánea.