Universidad de los Andes

Facultad de Ingeniería

Desarrollo de Soluciones Cloud

Arquitectura de la aplicación

Grupo 1

Juan Sebastian Colmenares - 202312351

William Fernando Alarcón - 202423127

Jhon Mario Forero – 20123466

Contenido

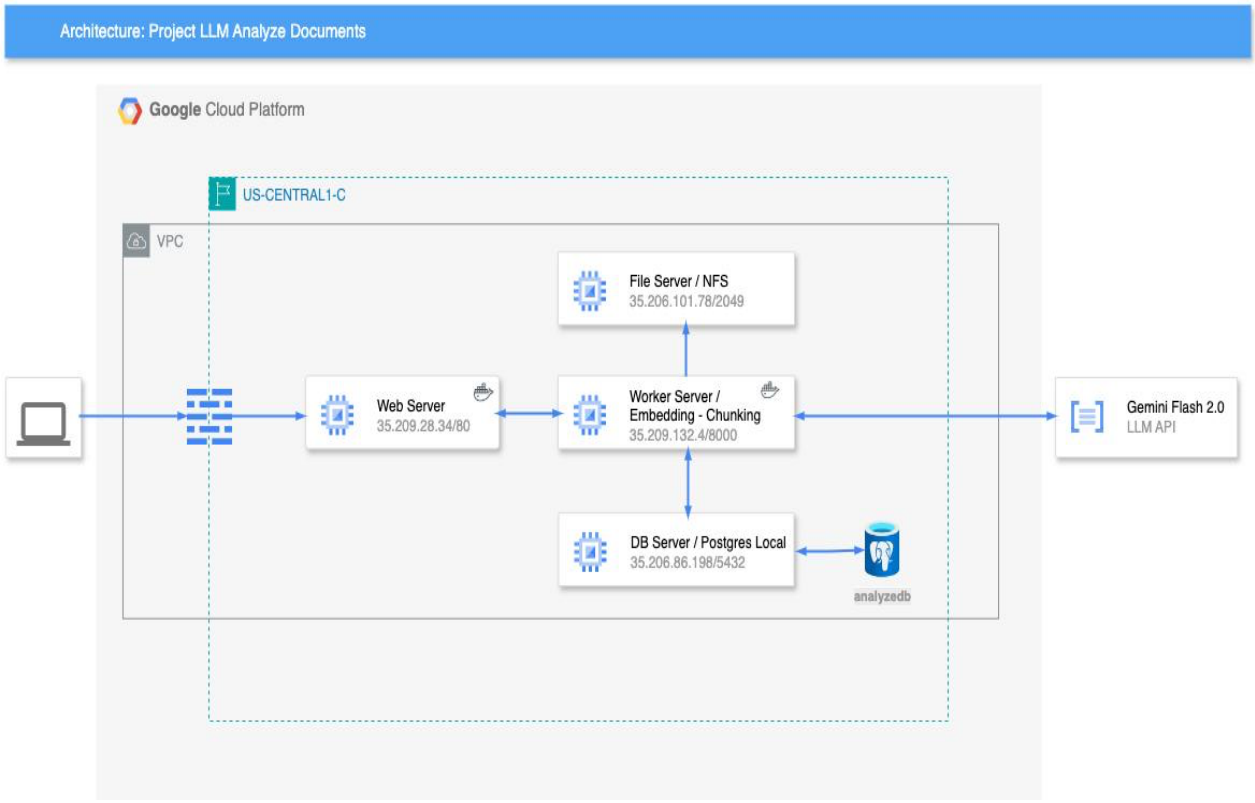
Descripción General de la Arquitectura 3
Configuración Actual del Entorno ..... **¡Error! Marcador no definido.**
Arquitectura de la solución en GCP 3
Recomendaciones para Escalamiento.....4
Limitaciones del Desarrollo Actual 4

Descripción General de la Arquitectura

El proyecto "LLM Analyze Documents" fue implementado sobre Google Cloud Platform, específicamente en la zona **US-CENTRAL1-C**, y se diseñó para permitir la carga, procesamiento y análisis de documentos mediante un LLM externo (Gemini Flash 2.0). La arquitectura se compone de los siguientes elementos desplegados dentro de una **VPC privada de GCP**:

- **Web Server** (Compute Engine - 4 vCPU, 4 GiB RAM, 20 GiB disco): Encargado de exponer la interfaz web y recibir peticiones REST API de los usuarios.
- **Worker Server** (Compute Engine - 4 vCPU, 4 GiB RAM, 50 GiB disco): Realiza procesamiento intensivo, embedding y chunking de los documentos.
- **File Server (NFS)** (Compute Engine - 2 vCPU, 2 GiB RAM, 20 GiB disco): Proporciona almacenamiento compartido para los documentos subidos.
- **DB Server** (PostgreSQL Local - 2 vCPU, 2 GiB RAM, 20 GiB disco): Guarda metadatos y resultados del análisis de documentos.
- **Gemini Flash 2.0 API**: LLM externo que recibe los chunks de texto para su análisis.

Arquitectura de la solución en GCP



Recomendaciones para Escalamiento

Para una futura versión del sistema, con usuarios concurrentes mucho mayores, se recomienda:

- **Autoescalado en Web Server y Worker**: Configurar un grupo de instancias manejado para que escale horizontalmente con base en CPU o número de conexiones.
- **Migración a Cloud SQL**: Para una base de datos administrada con mejor escalabilidad, replicación automática y backup.
- **Carga paralela al LLM**: Implementar un sistema de colas (como Pub/Sub o Cloud Tasks) para manejar múltiples solicitudes al modelo de forma controlada.
- **Monitoreo**: Configurar Cloud Monitoring y Cloud Logging para métricas críticas como CPU, latencia, errores y throughput.

Limitaciones del Desarrollo Actual

- Arquitectura monolítica inicial: Aunque separada por roles, todos los servicios están en una misma zona de disponibilidad y sin redundancia.
- Procesamiento secuencial: Actualmente el worker procesa por lotes, sin orquestación ni pipelines.
- Sin CDN ni control de tráfico global: La capa web no tiene Cloud CDN ni balanceadores regionales.
- Límite en la escalabilidad del File Server NFS: En altos volúmenes puede ser cuello de botella.
- No se contemplan mecanismos de recuperación ante fallos ni alta disponibilidad (HA).