

## Arquitectura, conclusiones y consideraciones

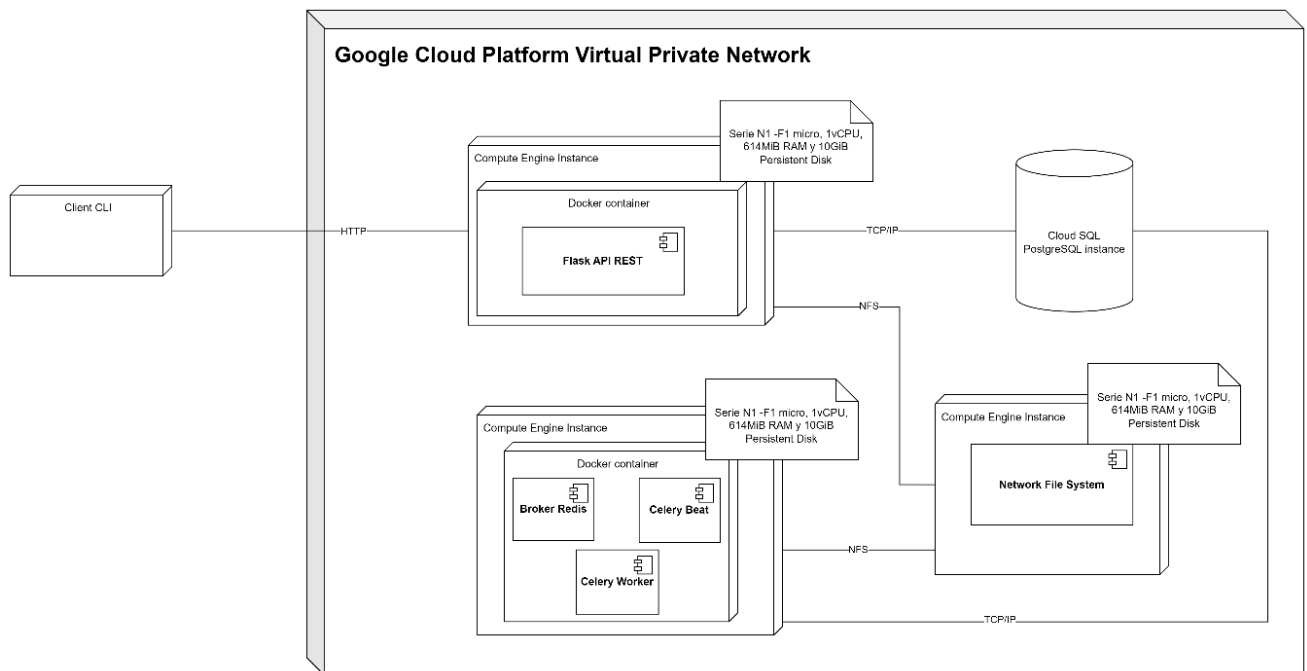
### Entrega 2 – Desarrollo de Software en la Nube

Diego Alejandro Camelo Giraldo, Juan Sebastián Alegría Zúñiga, María Camila

Gómez Hernández, Andrés Felipe Lugo Saavedra

### Grupo 6

#### Diagrama de arquitectura



Respecto a la arquitectura de la aplicación, para esta entrega desplegamos nuestro programa en Google Cloud Platform.

Desplegamos 3 instancias de Compute Engine, cada una con las mismas especificaciones, las cuales pueden ser visualizadas en el diagrama.

También utilizamos una instancia de Cloud SQL con PostgreSQL 14.

Todos los programas utilizados en las instancias de Compute Engine son desplegados en contenedores de Docker, a excepción del Network File System.

### **Conclusiones identificadas en la prueba de estrés**

De las pruebas de estrés logramos concluir que la aplicación tiene una muy buena capacidad para atender solicitudes de varios usuarios al mismo tiempo sin sobrecargar el sistema, esto debido a su despliegue en la nube, ya que las métricas mejoraron considerablemente en comparación a los resultados recolectados en el despliegue anterior. No obstante, al llegar al cuello de botella de la aplicación el rendimiento baja considerablemente, provocando algunos fallos y demoras en los tiempos de espera. Cabe resaltar que llegar a este punto en el despliegue de la nube requirió de más solicitudes que en el despliegue anterior, mostrando como la escalabilidad en la nube permite mejorar la escalabilidad de las aplicaciones sin que estas comiencen a tener fallas totales o tiempos de espera extremadamente largos entre ellos. por lo que hay que tener en cuenta esto a la hora de entender la capacidad de la aplicación.

### **Consideraciones para escalabilidad**

Este proyecto podría tener una mayor escalabilidad si se rompiera su estructura que es similar a la de una aplicación monolítica y la convertimos en una aplicación basada en microservicios. Esto permite tener pequeños servicios que pueden manejar cada característica de la aplicación y ser escalados de manera individual dependiendo de sus necesidades. Con esta propuesta buscamos optimizar el rendimiento de la aplicación ya que esta estrategia debería facilitar un escalamiento horizontal y mejorar el manejo de solicitudes en comparación al que se tiene actualmente

## **Limitaciones de desarrollo**

Durante el desarrollo de esta aplicación, hay varios componentes que se presentan como limitadores en lo que se puede lograr en la aplicación. Uno importante es la potencia de las máquinas virtuales con las que contamos para hacer las respectivas pruebas del software. En algunos casos, sucedía que se sobrecargaba la CPU y las solicitudes colapsaban, o en otros que la memoria RAM se llenaba y era imposible realizar las solicitudes. Otra restricción importante es el alcance que se tiene de la aplicación, pues no se deberían gastar recursos valiosos en desarrollar soluciones para características que no representan algo valioso para el objetivo final. Por último, se debe tomar en cuenta el tiempo que tenemos para desarrollar la aplicación y la velocidad de desarrollo del equipo, pues esto puede afectar la capacidad de soluciones que se pueden realizar.