

MANUAL DE USUARIO

TABLA DE CONTENIDO

1. OBJETICO
2. DEFINICIONES
3. DESARROLLO DEL MANUAL DE USUARIO

1. OBJETICO

Establecer los pasos específicos para el manejo del sistema.

Con el fin de poder verificar lo positivo y negativo de las olimpiadas y manejar unas aplicaciones tecnológicas.

2. DEFINICIONES

Es una aplicación que permite postear un comentario respecto a una noticia de las olimpiadas utilizando diferentes aplicaciones tecnológicas.

El sistema facilita el registro, actualizaciones y administración de la información.

3. DESARROLLO DEL MANUAL DE USUARIO

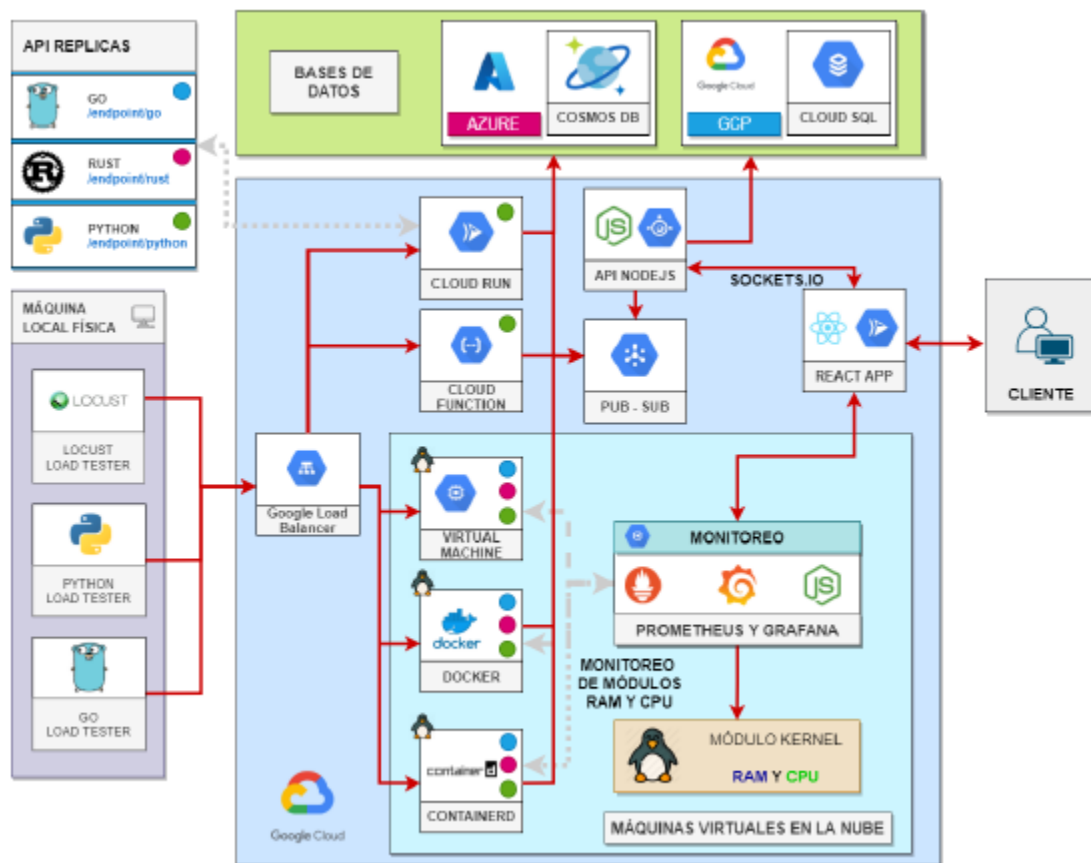
Servicio de Comentarios y Notificaciones.

Es un sistema computacional distribuido, cloud native, utilizando diferentes servicios de Google Cloud Platform, virtualización a nivel de sistema operativo con Docker y Containerd y generadores de tráfico para ser aplicado a un tema actual.

DESCRIPCIÓN GENERAL

A partir de la finalización de los juegos olímpicos Tokio 2021, se tiene un visualizador de las noticias y comentarios de las olimpiadas que están realizando los espectadores.

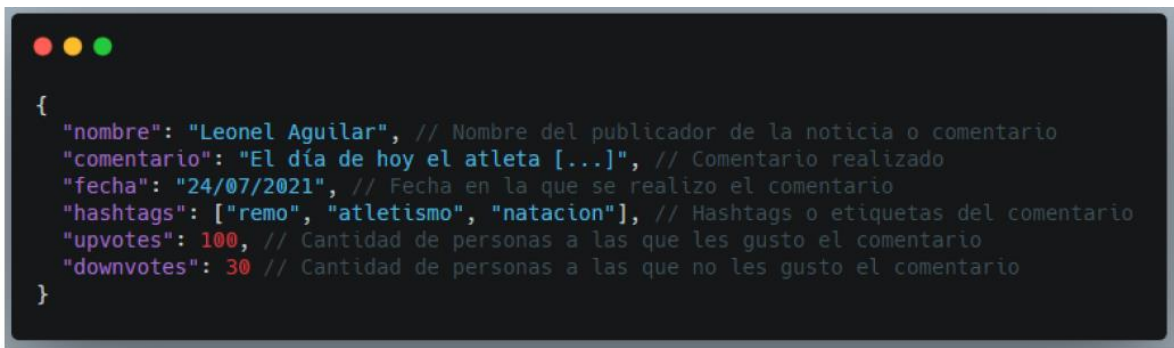
El sistema está totalmente en la nube, utilizando diferentes servicios de Google Cloud Platform y una base de datos de Cosmos DB en Microsoft Azure. cuenta con una carga masiva de datos a partir de diferentes generadores de tráfico, la información a mandar será detallada más adelante. Además de este sistema, se cuenta con un modo “Administrador” en el cual se visualiza gráficas y métricas relevantes de las noticias, y de la información de la RAM y procesos de las máquinas virtuales que se tendrán en la nube.



La finalidad de las aplicaciones que están en esta parte es enviar tráfico al Load Balancer de Google Cloud (detallado más adelante). Básicamente estas aplicaciones tienen las siguientes funcionalidades:

- Leer un archivo .json que contenga un array de datos (los datos respectivos a cada generador de tráfico).
- Procesar el archivo y enviar tráfico a un endpoint.
- Mostrar cuantos datos fueron enviados y cuantos datos tuvieron error.

La información a enviar por los tres generadores de tráfico es la siguiente:



```
{
  "nombre": "Leonel Aguilar", // Nombre del publicador de la noticia o comentario
  "comentario": "El día de hoy el atleta [...]", // Comentario realizado
  "fecha": "24/07/2021", // Fecha en la que se realizo el comentario
  "hashtags": ["remo", "atletismo", "natacion"], // Hashtags o etiquetas del comentario
  "upvotes": 100, // Cantidad de personas a las que les gusto el comentario
  "downvotes": 30 // Cantidad de personas a las que no les gusto el comentario
}
```

API REPLICAS

Se tienen 3 apis de diferentes lenguajes go, Python y rust para poder enviar información las cuales cuentan con diferentes endpoints para que sea de su conocimiento como se esta manejando dicha información.

En cada uno de estos lenguajes deberán crear la misma funcionalidad de la API para conectarse a la base de datos y enviar información enviada desde los generadores de tráfico. Además, se debe crear una notificación de cuantos datos fueron cargados en Google Pub/Sub.

PARA SU CONOCIMIENTO COMO USUARIO

CLOUD RUN – APIS

En Google Cloud Run se cargará el servicio de la API de Python y de Go.

CLOUD FUNCTIONS

En Google Functions se cargará el servicio de la API de Python y de Go.

MÁQUINAS VIRTUALES EN LA NUBE

Se tendrán tres máquinas virtuales en la nube en Google Compute Engine, cada una con las mismas APIs que se detallaron anteriormente.

MÓDULO LISTA DE PROCESOS

Montar un módulo que lea la información del CPU del sistema. Este módulo debe proveer la siguiente información:

- Porcentaje de CPU utilizado
- Cantidad de procesos ejecutándose

MÓDULO DE MONITOREO

En este módulo es un Dashboard utilizando Prometheus y Grafana donde se observe la información de la RAM (toda la que aparece en el Kernel) y la información del CPU y cantidad de procesos ejecutándose.



GOOGLE PUB SUB

Se utilizará Google Pub/Sub para enviar notificaciones acerca de la carga de información desde las APIs hasta la API principal de NodeJS. La información de esta notificación es la siguiente:

```
{
  "guardados": 1000, //cantidad de registros almacenados
  "api": "Python", //desde qué API se guardó la info, puede ser RUST, PYTHON O GO
  "tiempoDeCarga": 10, //Tiempo que tardo la transaccion en segundos
  "bd": "CosmoDB", //Base de datos en la que se guardo la info, CosmoDB o CloudSQL
}
```

APP ENGINE

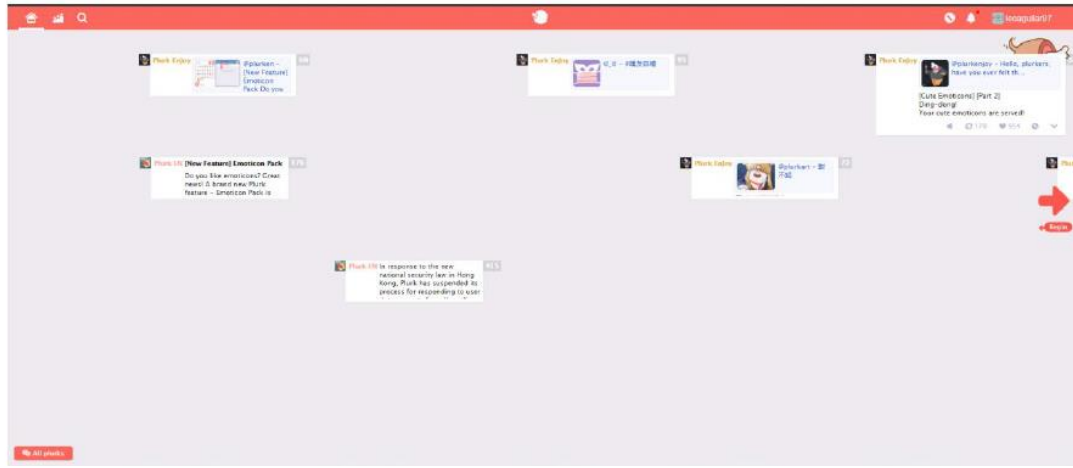
APP WEB

Se realizo una app web utilizando el framework React para que usted como usuario pueda comentar sobre las olimpiadas. La información que usted envia para su conocimiento se almacena en el servidor de CosmosDB y de CloudSQL.

VISTA DE INFORMACIÓN

La información se muestra como por ejemplo.

VISTA DE NOTICIAS Y MENSAJES



Como usted puede publicar su comentario.




VISTA DE REPORTE


- Mostrar el total de noticias, upvotes y hashtags (diferentes) que hay en el sistema.
- Tabla con los datos que están en el sistema.
- Gráfica circular del top 5 de hashtags. Se calculará a partir de la cantidad de upvotes que tiene cada hashtag.
- Gráfica (de barras, líneas, stackeada, etc...) que compare la cantidad de upvotes y downvotes por día.

También se debe agregar un botón o filtro con el cual se pueda decidir desde qué base de datos realizar los reportes (es decir, deben implementar los reportes en ambas bases de datos). Se recomienda utilizar la siguiente


NOTICIAS
21 324

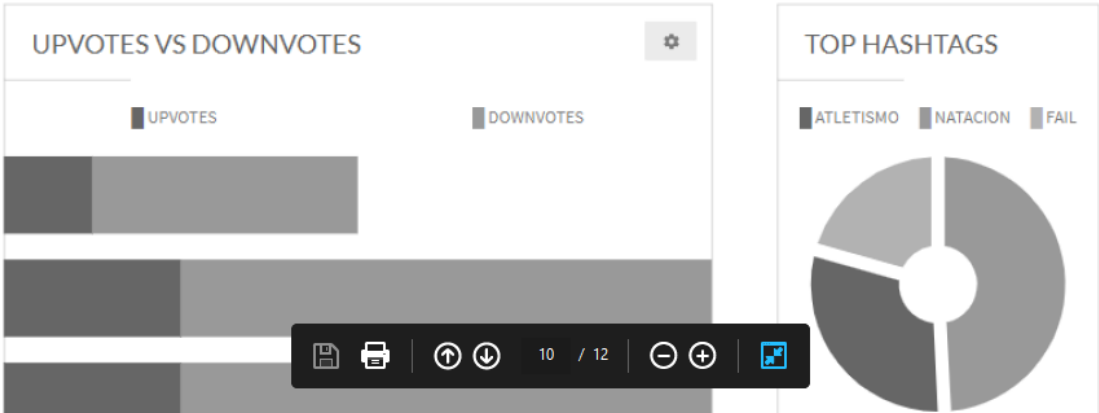


HASHTAGS
300 DIFERENTES



UPVOTES
65 2234



ENTRADAS RECIENTES

Jerry Mattedi	13 aug 2018 Fecha	OLIMPIADAS! Mensaje	noticias, fail Hashtags	Options	VER
Elianora Vasilov	13 aug 2018 Fecha	HOLA A TOD... Mensaje	saludo, hola Hashtags	Options	VER
Marcos Anguiano	13 aug 2018 Fecha	ATLETISMO ... Mensaje	Atletismo, 500m Hashtags	Options	VER
Alvis Daen	13 aug 2018 Fecha	TOKIO ES LO ... Mensaje	Tokio Hashtags	Options	VER
Lissa Shipsey	13 aug 2018 Fecha	HOY JUAN NA... Mensaje	Natacion, noticias Hashtags	Options	VER