

Actividad U3. GCP (x%)

A) Simulación de publicación e ingesta de datos.

Vamos a usar varios recursos de GCP para simular una publicación e ingesta de datos. Los servicios GCP implicados son:

- Servicio de Pub/Sub.
- Google Storage.
- Google Compute.

Simularemos la gestión de canales de publicación y suscripción mediante scripts de Python que permitan tratar la generación de eventos y su consumo.

Se compone de tres partes:

Los eventos.

Los eventos generados por el script *randomEvents.py* se componen de un tiempo, un topic y un mensaje. Hay tres tipos de topics: *search*, *book* y *buy*. Los mensajes contienen la codificación del usuario. Finalmente, el tiempo representa el periodo entre llegadas de dicho evento al sistema.

Nota: Este fichero ya lo tenéis creado. Podéis ejecutar y ver el fichero csv generado. Es **obligatorio** que cambies el valor de la variable *VUESTRA_SIGLA* con vuestras iniciales.

La publicación.

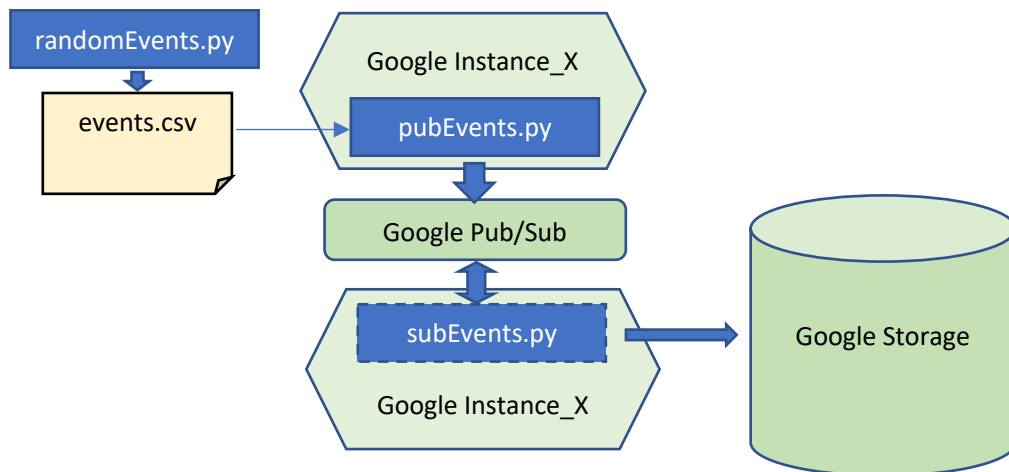
Cada evento será consumido por el script de Python llamado *pubEvents.py*. Este script publicará en Google Pub/Sub cada evento. El script se debe de ejecutar en una instancia de Google Compute y además, tendréis que introducir nuevas líneas de código para que publique el mensaje con su correspondiente topic.

Nota: Este fichero ya lo tenéis creado. Podéis ejecutar y ver el comportamiento de este script en vuestra máquina local.

La suscripción e ingesta.

Finalmente, otro script de Python *subEvents.py* se suscribirá a Google Pub/Sub. Ha de funcionar al mismo tiempo que se publiquen mensajes. No es necesario una sincronización perfecta de inicio entre ambos. Su función es mantener un histórico de cada mensaje recibido. Este histórico de mensajes se guardarán filtrados en tres ficheros correspondientes a cada topic que estarán en un bucket del Google Storage. Es decir, si recibe un mensaje del topic *buy* con un mensaje cualquiera, dicho mensaje se guardará en una nueva línea del fichero */buys* del bucket. Cuando las “publicaciones acaben” tendremos un histórico de todos los mensajes recibidos en los tres ficheros correspondientes.

Nota: Este fichero NO lo tenéis creado.

Arquitectura del proceso.**Observaciones:**

- Los scripts de pub/sub apenas requieren recursos computacionales. Usad la misma instancia al mismo tiempo para ejecutarlos.
- El Google Storage contratado puede tener una capacidad mínima <10Gb. No hace falta más.

Referencias:

- Breve guía de como usar pub/sub mediante Gcloud command:
http://console.cloud.google.com/?tutorial=pubsub_quickstart&hl=es-419
- Código en diferentes lenguajes de como usar pub/sub:
<https://cloud.google.com/pubsub/docs/create-topic-client-libraries?hl=es-419>

Entregables:

- El código en un repositorio de datos. No dejes nunca un password en dicho código. En ningún caso es necesario, pero por si las moscas.
- Un informe con la URL del repositorio y una descripción de las tareas realizadas. Se valorará una pequeña sección con comentarios personales de la práctica.

RECUERDA detener y/o eliminar los
RECURSOS DE PAGO al finalizar la
actividad