



*Facultad de Ingeniería*  
*Gestión de datos.*

*Taller Final – Primer Semestre 2023*

Taller Final.

## Preparación de datos, visualización y modelo de recomendación

Joshep Andersson Blanco Reres<sup>a,c</sup>, Laura Estrada Saldarriaga<sup>a,c</sup>

Fabián Camilo Peña<sup>b,c</sup>

*a Estudiante de Maestría en Inteligencia Artificial.*

*b Profesor, Departamento de Ingeniería de Sistemas*

*c Pontificia Universidad Javeriana, Bogotá, Colombia*



## Contenido

SEGUNDA ENTREGA. CONSTRUCCIÓN DEL ETL .....	2
1. Arquitectura ETL .....	3
2. CONSTRUCCIÓN DASHBOARD .....	6
ANEXO 1. Detalles del pipeline de automatización ejecutado en VertexAI .....	9
ANEXO 2. Detalles de la configuración del bucket y de la base de datos. ....	11
ANEXO 3. Conexión de la base de datos en GCP a Google DataStudio (Visualizador.) .....	15

## ENTREGA FINAL

El repositorio se encuentra en el siguiente link. [https://github.com/Joshep1229/spotify\\_project](https://github.com/Joshep1229/spotify_project)  
Dashboard: <https://lookerstudio.google.com/reporting/86e0b128-1eda-4927-ae03-42a04ea3a4d0>

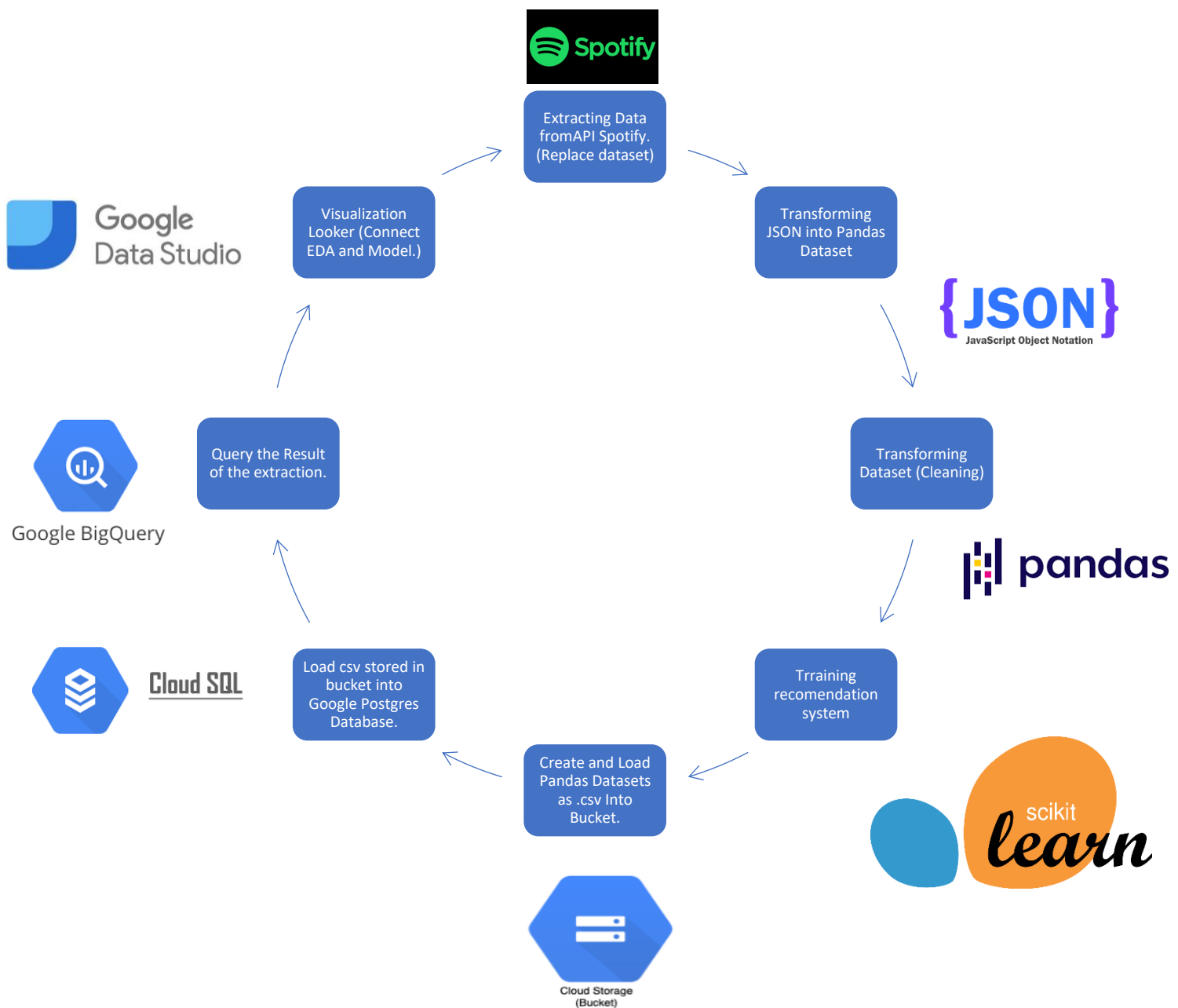
## 1. Arquitectura ETL

Orchestrator:



Vertex AI

Pipeline of Information:



Este proyecto contiene una solución end to end **en la nube** para ejecutar desde la extracción hasta la visualización y el modelo de recomendación de forma automática usando VertexAI.

**Las playlists usadas fueron el top 50 global, el top 50 Brazil y el top 50 Corea del sur.** Los datos generados y el modelo, se actualizan **semanalmente**.

Esto permite que el sistema **siempre generará recomendaciones nuevas y personalizadas de los mejores 50 canciones de los 3 países.** Tomando en cuenta las canciones que ingresen o salgan del top.

#### **Flujo en la nube en detalle:**

1. Cargar Información desde la **\*\*API de spotify\*\***.
2. Preprocesarla y generar datasets (Artistas, canciones, género, datos de modelo y gráfico de similitud.)
3. Procesar **\*\*modelo de recomendaciones\*\*** y guardar los resultados como un csv (Recomendaciones)
3. Guardar los dataset como formato .csv en un **\*\*bucket en GCP\*\***. (Previa conexión configurada en el notebook.)
4. Con SQL cargar el archivo de csv guardado en el bucket en la base de datos de **\*\*Google Bigquery de GCP\*\*** (que está en postgres).
5. Conectar la base de datos de Google Bigquery a **\*\*Google Datastudio (Visualizador)\*\***
6. Visualizar los resultados en Google datastudio para visualizar la información.
  - 6.1 La primera hoja contiene el dashboard.
  - 6.2 La segunda hoja contiene el modelo de recomendaciones generadas por el modelo.

## **2. SISTEMA DE RECOMENDACIÓN.**

Para el sistema de recomendación, se ejecutó el modelo en el flujo preestablecido de los datos

Se ejecuta el modelo con los parámetros de 'danceability', 'energy', 'loudness' y 'valence'.

Se intentó hacer un modelo más complejo agrupando variables como la duración, el género e incluso si la playlist era coreana, brasilera o global, sin embargo sesgaba mucho el resultado y se perdía la variedad en las canciones.

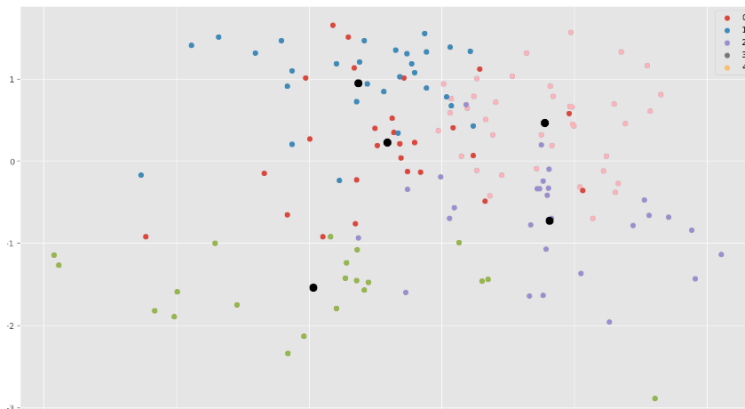
En éste caso quisimos tener un equilibrio entre recomendar música similar, pero también mostrarle elementos distintos al usuario, esa es la razón por la que escogimos 3 mercados muy dispares en la comparación. (Mercado coreano, brasilero y global.)

Por ello, también agrupamos cuatro elementos a nuestro modelo, la energía, la bailabilidad, el ruido y el valence, o el qué tan positiva sonaba la canción. Dejamos por fuera elementos como el género o el país donde se generó la playlist, para darle variedad.

```
[327] from sklearn.cluster import KMeans
import numpy as np

X=np.array(Datos_Modelo[['danceability', 'energy', 'loudness','valence']])
X_norm = (X - X.mean(axis=0)) / X.std(axis=0)

kmeans = KMeans(n_clusters=5).fit(X_norm)
label = kmeans.fit_predict(X_norm)
centroids = kmeans.cluster_centers_
print(centroids)
```



El modelo tiene 5 centroides a los que se les calcula la distancia al centroide y posteriormente la distancia entre los puntos usando la distancia euclidiana.

Esta matriz cuadrada de 143 canciones (50 por cada top, menos las repetidas), se convierte a un formato largo, que contiene:

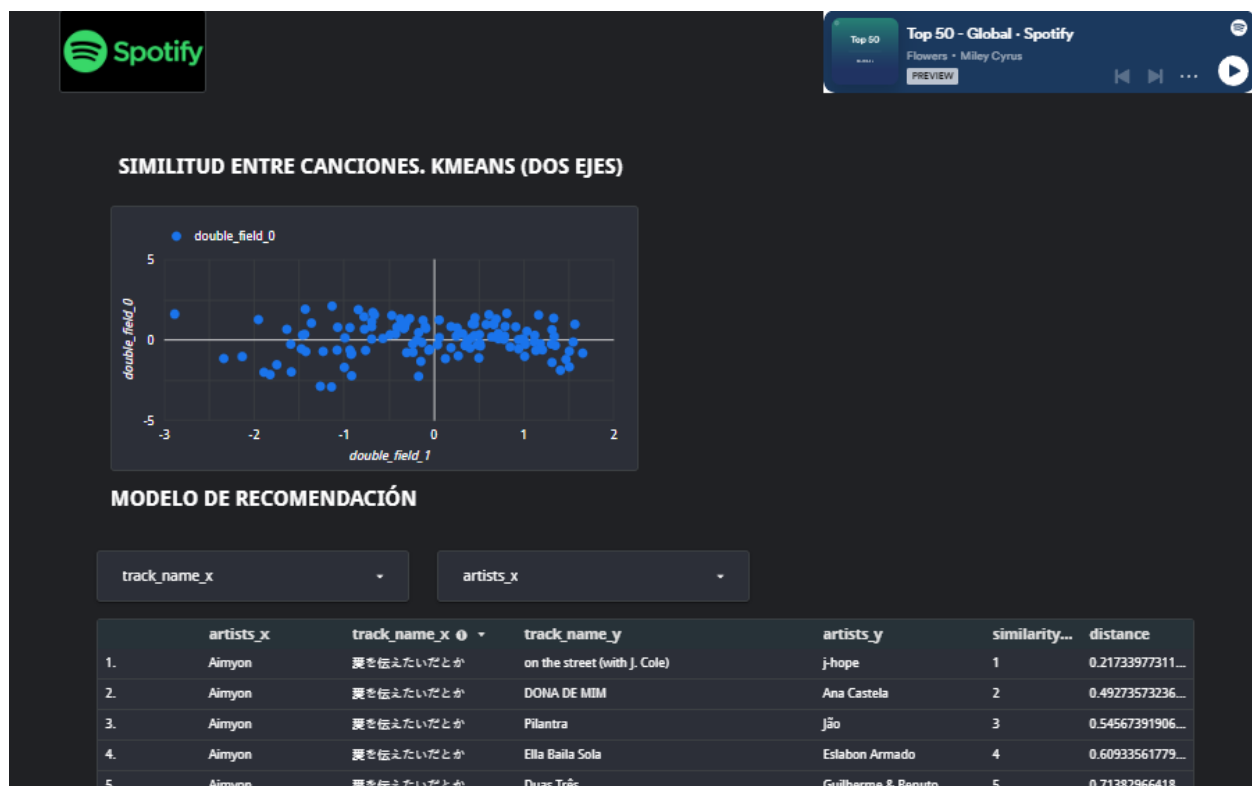
Canción A, Artista	Canción A, Arista	Canción B, Arista	Canción B, Distancia,	Ranking
Bad Bunny	Ojitos lindos	TQG Karol G	0,42	1
Bad Bunny	Ojitos lindos	Mor Ferxxo	0.67	2
Bad Bunny	Ojitos lindos	Para Elisa Mozart	4.52	149

Así sucesivamente hasta tener todos los pares posibles para todas las canciones que contienen el dataset.

Ésta base se carga como una base de datos en bigquery, no se entrena un modelo y se realiza la exportación a un joblib, por la naturaleza de la base de datos, debido a que si entran y salen canciones del top de los tres países, es mejor reentrenar el modelo con los nuevos elementos de datos que se generen.

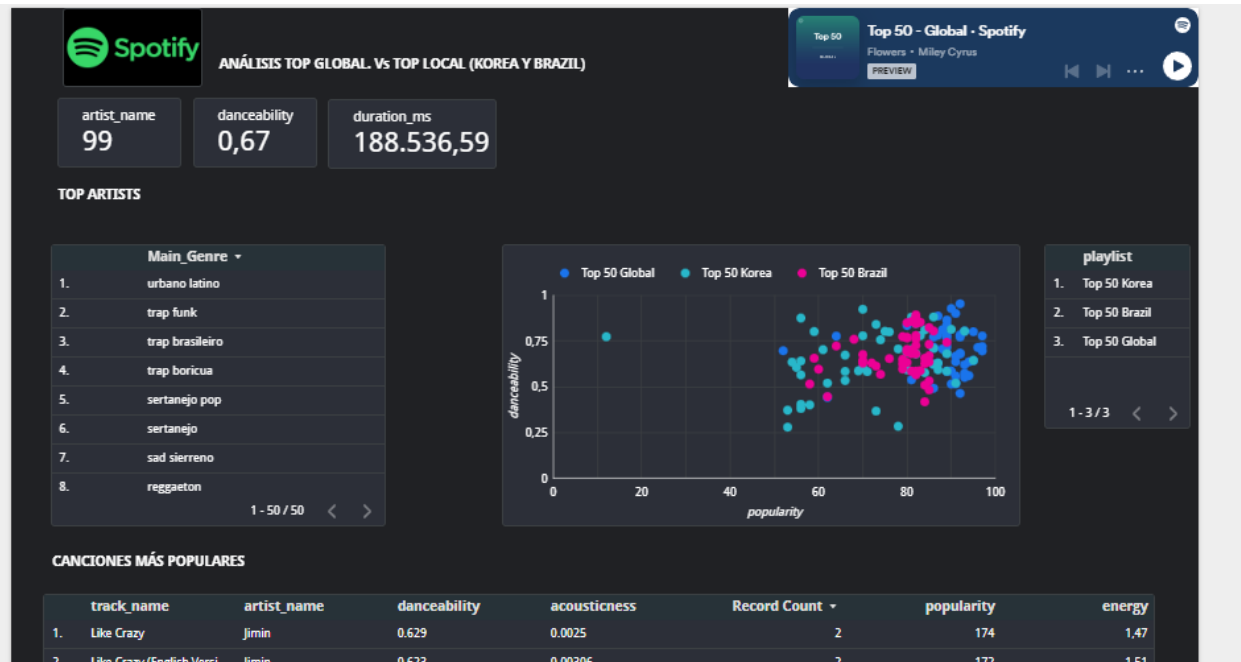
Se crea una base de datos y no un API rest, porque desde el visualizador, podemos conectar el sistema de recomendación como un conjunto de datos en big query y visualizar las recomendaciones de todos los artistas (además de mantener en la nube la solución.).

Sistema de recomendación integrado a Datastudio.



MODELO DE RECOMENDACIÓN						
track_name_x: Flowers		(1) ▾	artists_x ▾			
	artists_x	track_name_x ▾	track_name_y	artists_y	similarity...	distance
1.	Miley Cyrus	Flowers	Nosso Quadro	AgroPlay	1	0.37743170473...
2.	Miley Cyrus	Flowers	TQG	KAROL G	2	0.53241413611...
3.	Miley Cyrus	Flowers	Haja Corírio (feat. Hugo & Guilherme) - Ao Vivo	Guilherme & Benuto	3	0.53671156512...
4.	Miley Cyrus	Flowers	LOVE DIVE	IVE	4	0.54575767595...
5.	Miley Cyrus	Flowers	Namorando Ou Não - Ao Vivo	Clayton & Romário	5	0.71549410400...
6.	Miley Cyrus	Flowers	VIBE (feat. Jimin of BTS)	TAEYANG	6	0.74606455408...
7.	Miley Cyrus	Flowers	Bones	Imagine Dragons	7	0.77170267535...
8.	Miley Cyrus	Flowers	Yandel 150	Yandel	8	0.80400349420...
9.	Miley Cyrus	Flowers	DONA DE MIM	Ana Castela	9	0.84401003197...
10.	Miley Cyrus	Flowers	Polaroid	Lim Young Woong	10	0.88881661806...

### 3. CONSTRUCCIÓN DASHBOARD



El dashboard se genera usando datastudio en conexión con BigQuery, y contiene información importante sobre el total de artistas analizados, el tipo de canciones examinadas, y métricas. Clave, como el promedio de duración de las canciones.

Nos da una visual interesante de cómo se diferencian las canciones dependiendo del país, su nivel de popularidad y de su bailabilidad, podemos observar cómo Corea del Sur tiene canciones más planas en comparación al top de Brasil o al top global.

Son variables importantes para el negocio, porque la compañía puede indagar sobre las métricas que la podrían hacer triunfar en cada mercado. Y de cómo podría acercarse a cada país. También permite visibilizar artistas que podrían ser muy populares fuera de sus fronteras al encontrarle similitudes con los artistas de otros mercados, como se verá en el sistema de recomendación.

También se inserta la app de Spotify para que el usuario pueda disfrutar de la música mientras realiza la exploración del mismo.



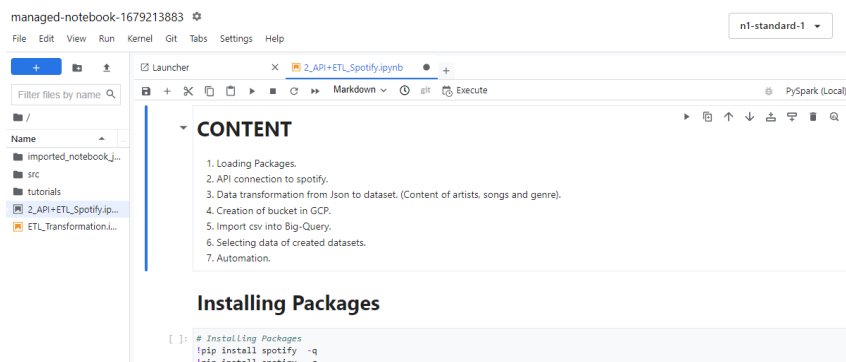
## ANEXO 1. Detalles del pipeline de automatización ejecutado en VertexAI

### AUTOMATIZACIÓN USANDO VERTEX AI.

Vertex AI es un sistema que permite ejecutar, compilar y escalar modelos de datos (MLOps), para el caso nuestro, ha sido usado para automatizar el proceso de ETL, pero también se podría ejecutar el sistema de recomendación.

Éste sistema crea toda la lógica y la ejecución automáticamente en Google Composer. Para tal finalidad, se debe crear una máquina virtual que ejecute el Jupyter notebook dentro de la aplicación, hemos escogido la más básica de las opciones, una CPU de 1 GB sin tarjeta gráfica.

Luego de la creación del entorno, subimos el notebook y se verá se la siguiente manera.



Le damos en el ícono del reloj y mostrará las siguientes opciones de programación de la tarea.

Submit notebooks to Executor

Accelerator type  
None

Environment  
Custom Container

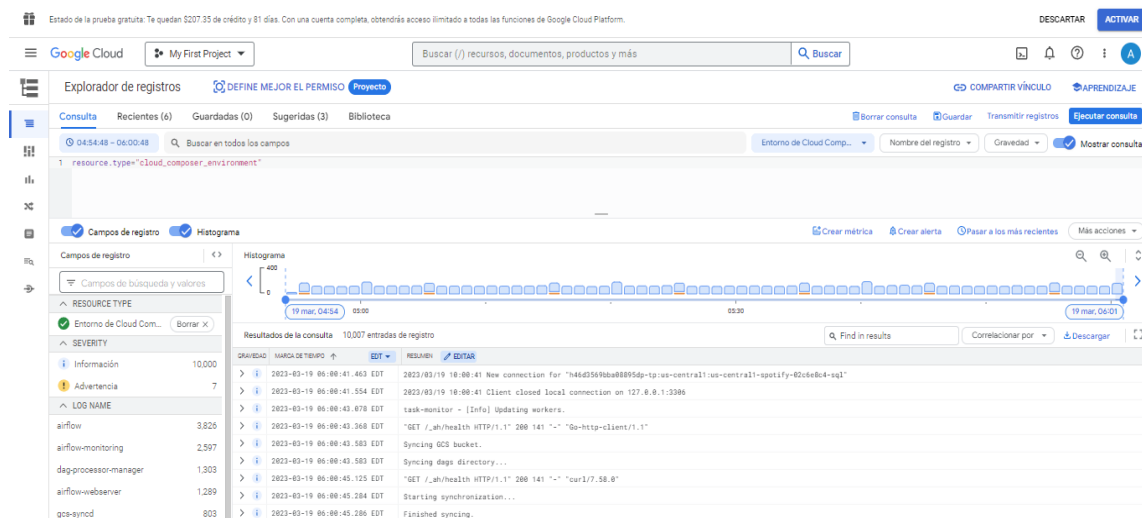
Docker container image  
us-docker.pkg.dev/deeplearning-platform/gcr.io/release.base-cu113:m1  
Note: Custom container must be a [derivative container](#).

Type  
Schedule-based recurring executions

Repeat every  
week

Repeat at  
09:00

Al ejecutarse, genera un log de ejecución que detalla tanto los errores, como las ejecuciones dadas por el programa.



Estado de la prueba gratuita: Te quedan \$207.35 de crédito y 81 días. Con una cuenta completa, obtendrás acceso limitado a todas las funciones de Google Cloud Platform.

Google Cloud My First Project

Explorador de registros

Consulta Recientes (6) Guardadas (0) Sugeridas (3) Biblioteca

Buscar en todos los campos

Entorno de Cloud Com... Nombre del registro Gravedad Mostrar consulta

Campos de registro Histograma

Campos de búsqueda y valores

RESOURCE TYPE

Entorno de Cloud Com... (Borrar X)

SEVERITY

LOG NAME

airflow 3,826

airflow-monitoring 2,597

dag-processor-manager 1,303

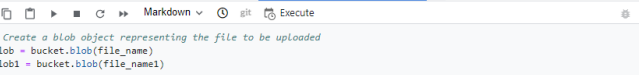
airflow-webserver 1,289

gcp-synod 803

Resultados de la consulta: 10,007 entradas de registro

GRAVEDAD	FECHA DE TIEMPO	RESUMEN
Info	2023-03-19 06:00:41.463 EDT	2023/03/19 16:00:41 New connection for "h46d5569ba88895dp-tp-us-central1-us-central1-spotify-62c6b64-sql"
Info	2023-03-19 06:00:41.554 EDT	2023/03/19 16:00:41 Client closed local connection on 127.0.0.1:3386
Info	2023-03-19 06:00:43.678 EDT	task-monitor - [Info] Updating workers.
Info	2023-03-19 06:00:43.368 EDT	"GET /_ah/health HTTP/1.1" 200 141 "-" "Go-http-client/1.1"
Info	2023-03-19 06:00:43.583 EDT	Syncing GCS bucket.
Info	2023-03-19 06:00:43.583 EDT	Syncing dags directory...
Info	2023-03-19 06:00:45.125 EDT	"GET /_ah/health HTTP/1.1" 200 141 "-" "curl/7.58.0"
Info	2023-03-19 06:00:45.284 EDT	Starting synchronization...
Info	2023-03-19 06:00:45.286 EDT	Finished syncing.

También podemos ver los errores de ejecución del notebook, puntualmente.



The screenshot shows a Jupyter Notebook with two tabs: '2\_API+ETL\_Spotify.ipynb' and '2\_API+ETL\_Spotify.ipynb'. The active tab displays a code cell with the following Python code:

```
# Create a blob object representing the file to be uploaded
blob = bucket.blob(file_name)
blob1 = bucket.blob(file_name)

# Upload the file to the bucket
blob.upload_from_filename(file_path)
blob.upload_from_filename(file_path)
```

Below the code cell, a red error message is displayed:

```
-----
NameError                                Traceback (most recent call last)
/tmp/ipykernel_161/2986729059.py in <module>
      4
      5 # Create a client
----> 6 client = storage.Client.from_service_account_json(key_path)
      7
      8 # Set the name of the bucket to which you want to upload the file

NameError: name 'key_path' is not defined
```

The error message indicates that the variable 'key\_path' is not defined, which is consistent with the code snippet provided in the text above the screenshot.

## ANEXO 2. Detalles de la configuración del bucket y de la base de datos.

## Creación del bucket

Crear un bucket

• Asigna un nombre a tu bucket

Selecciónalo un nombre permanente globalmente único. [Lineamientos para asignar nombre](#)

spotify11

Sugerencia: No incluyas información sensible

✓ ETIQUETAS (OPCIONAL)

CONTINUAR

• Elige dónde almacenar tus datos

Ubicación: us (varias regiones en Estados Unidos)  
Tipo de ubicación: Multi-region

• Elige una clase de almacenamiento para tus datos

Clase de almacenamiento predeterminada: Standard

• Elige cómo controlar el acceso a los objetos

Prevención del acceso público: Activada  
Control de acceso: Uniforme

• Elige cómo proteger los datos de objeto

Herramientas de protección: Ninguno  
Encriptación de datos: Google-managed key

Información útil

Precios de ubicación

Las tarifas de almacenamiento varían según la clase de almacenamiento de los datos y la ubicación de los buckets. [Detalles de precios](#)

Configuración actual: Multi-region / Standard

Elemento	Costo
us (varias regiones en Estados Unidos)	\$0.026 por GB al mes
Con replicación predeterminada	\$0.020 por GB escrito

ESTIMAR COSTO MENSUAL

### Bucket Creado

← Detalles del bucket ACTUALIZAR APRENDIZAJE

**spotify11**

Ubicación: us-east1 (Carolina del Sur) | Clase de almacenamiento: Standard | Acceso público: No público | Protección: Ninguna

OBJETOS CONFIGURACIÓN PERMISOS PROTECCIÓN CICLO DE VIDA OBSERVABILIDAD **NUEVO**

Depósitos > spotify11

SUBIR ARCHIVOS SUBIR CARPETA CREAR CARPETA TRANSFERIR LOS DATOS ADMINISTRAR CONSERVACIONES DESCARGAR BORRAR

Filtrar solo por prefijo de nombre  Filtro: Filtrar objetos y carpetas Mostrar datos borrados

<input type="checkbox"/>	Nombre	Tamaño	Tipo	Fecha de creación	Clase de almacenamiento	Última modificación	Acceso público	Historial de versiones	Encriptación	Fecha de vencimiento de la retención	Conservaciones
No hay filas para mostrar											

Estado de la prueba gratuita: Te quedan \$400.00 de crédito y 91 días. Con una cuenta completa, obtendrás acceso ilimitado a todas las funciones de Google Cloud Platform.

Google Cloud My First Project cuentas de ser Buscar 4 ? J

IAM y administración Cuentas de servicio + CREAR CUENTA DE SERVICIO BORRAR ADMINISTRAR ACCESO ACTUALIZAR APRENDIZAJE

Cuentas de servicio del proyecto "My First Project"

Una cuenta de servicio representa una identidad de servicio de Google Cloud, como el código en ejecución en las VM de Compute Engine, las apps de App Engine o los sistemas que se ejecutan fuera de Google. [Obtén más información sobre las cuentas de servicio.](#)

Las políticas de la organización se pueden usar para asegurar las cuentas de servicio y bloquear sus características riesgosas, como el otorgamiento automático de IAM, la creación y carga de claves, o la creación misma de cuentas de servicio. [Obtén más información sobre las políticas de la organización para cuentas de servicio.](#)

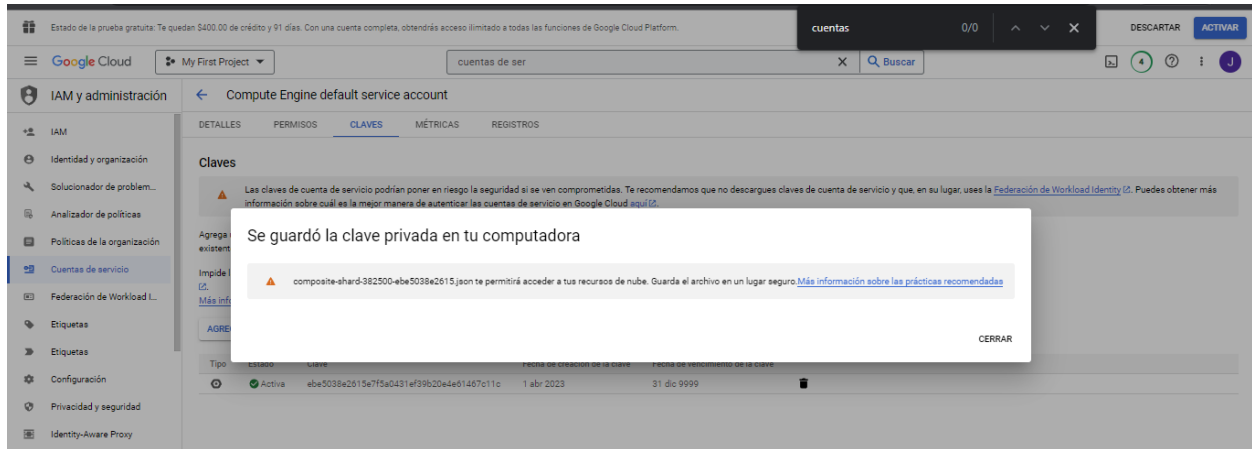
Filtro: Ingresar el nombre o el valor de la propiedad

<input type="checkbox"/>	Correo electrónico	Estado	Nombre	Descripción	ID de clave	Fecha de creación de la clave	ID de cliente de OAuth2	Acciones
<input type="checkbox"/>	<a href="mailto:270888618758-compute@developer.gserviceaccount.com">270888618758-compute@developer.gserviceaccount.com</a>	✓	Compute Engine default service account		No hay claves		11672137609083116234	Acciones <ul style="list-style-type: none"> <li>Administrar detalles</li> <li>Administrar permisos</li> <li>Administrar claves</li> <li>Ver métricas</li> <li>Ver registros</li> <li>Inhabilitar</li> <li>Borrar</li> </ul>

Se copió el URI de guñil al portapapeles

Operaciones de My First Project cargadas COMPETENCIAS CLAVES DE UN Completada

### Creación clave de bucket



Estado de la prueba gratuita: Te quedan \$400.00 de crédito y 91 días. Con una cuenta completa, obtendrás acceso ilimitado a todas las funciones de Google Cloud Platform.

cuentas 0/0

Google Cloud My First Project

cuentas de ser

IAM y administración

Compute Engine default service account

DETALLES PERMISOS CLAVES MÉTRICAS REGISTROS

Claves

Las claves de cuenta de servicio podrían poner en riesgo la seguridad si se ven comprometidas. Te recomendamos que no descargues claves de cuenta de servicio y que, en su lugar, uses la [Federación de Workload Identity](#). Puedes obtener más información sobre cuál es la mejor manera de autenticar las cuentas de servicio en Google Cloud [aquí](#).

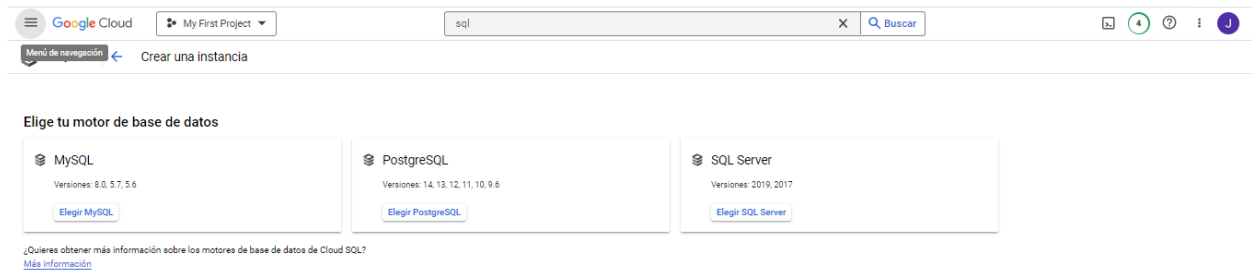
Se guardó la clave privada en tu computadora

composite-shard-382500-eba5038e2615.json te permitirá acceder a tus recursos de nube. Guarda el archivo en un lugar seguro. [Más información sobre las prácticas recomendadas](#)

CERRAR

Tipo	Estado	Nombre	Fecha de creación de la clave	Fecha de vencimiento de la clave
Active	Active	eba5038e2615e77f5e0431ef39b20e4e61467c11c	1 abr 2023	31 dic 9999

## Creación de SQL Cloud



Google Cloud My First Project

sql

Menú de navegación

Crear una instancia

Elige tu motor de base de datos

MySQL

Versiónes: 8.0, 5.7, 5.6

Elegir MySQL

PostgreSQL

Versiónes: 14, 13, 12, 11, 10, 9, 8

Elegir PostgreSQL

SQL Server

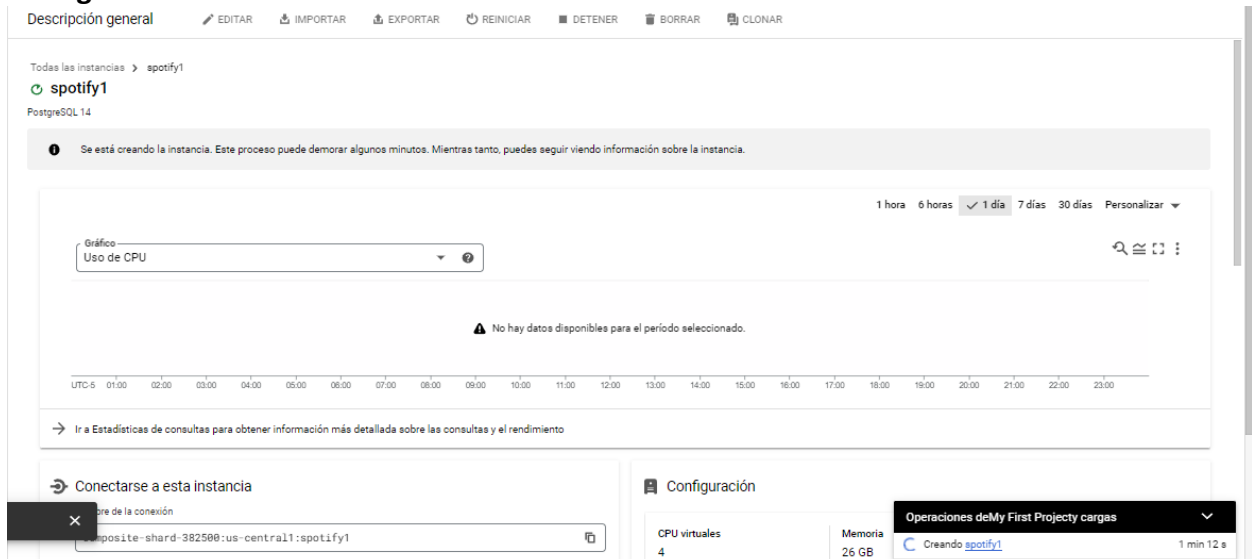
Versiónes: 2019, 2017

Elegir SQL Server

¿Quieres obtener más información sobre los motores de base de datos de Cloud SQL?

[Más información](#)

## Configuración de la instancia



Descripción general EDITAR IMPORTAR EXPORTAR REINICIAR DETENER BORRAR CLONAR

Todas las instancias > spotify1

spotify1

PostgreSQL 14

Se está creando la instancia. Este proceso puede demorar algunos minutos. Mientras tanto, puedes seguir viendo información sobre la instancia.

1 hora 6 horas 1 día 7 días 30 días Personalizar

Gráfico

Uso de CPU

No hay datos disponibles para el periodo seleccionado.

UTC-5 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00 12:00 13:00 14:00 15:00 16:00 17:00 18:00 19:00 20:00 21:00 22:00 23:00

→ Ir a Estadísticas de consultas para obtener información más detallada sobre las consultas y el rendimiento

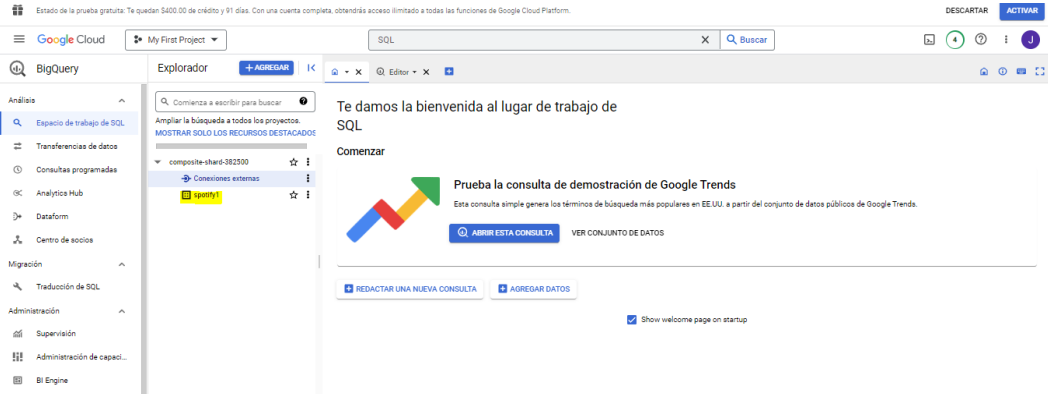
Conectarse a esta instancia

Configuración

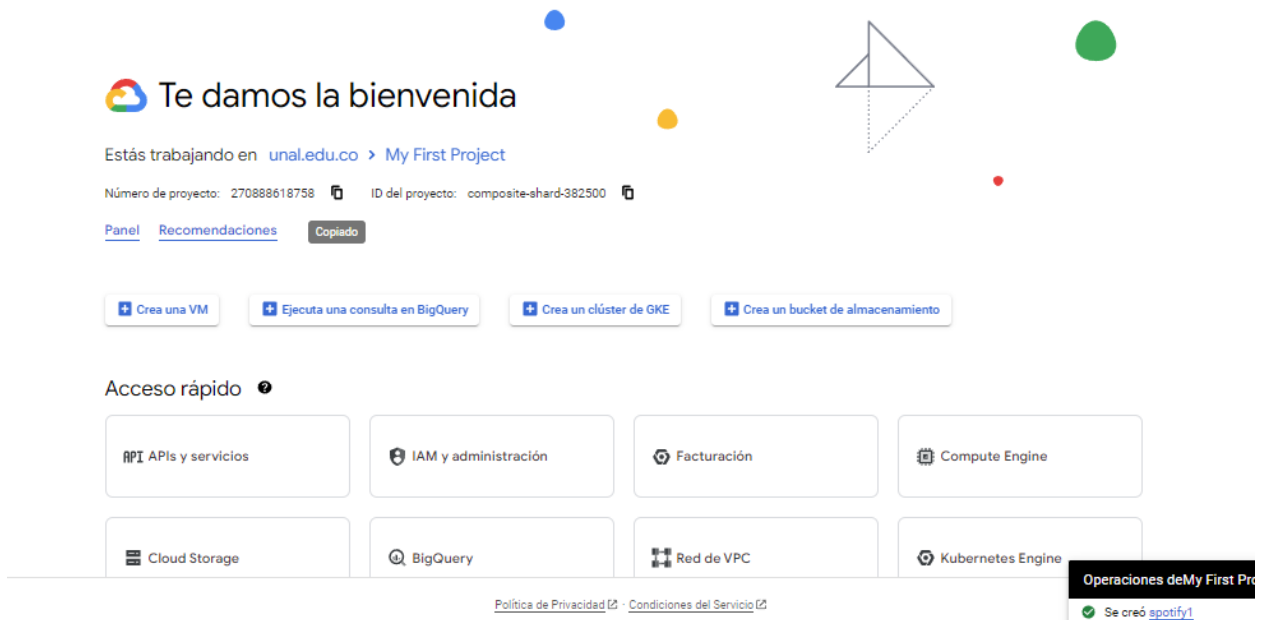
Operaciones de My First Project cargadas

Creando spotify1 1 min 12 s

## Creación BBDD Big Query



### Obtener Id del proyecto:



## ANEXO 3. Conexión de la base de datos en GCP a Google DataStudio (Visualizador.)

Informe sin título

Restablecer
Compartir
Ver

Archivo
Vista
Página
Ayuda

Añadir página
Añadir datos
Añadir un gráfico
Añadir un control

Tema y diseño

Añadir datos al informe

Credenciales de datos: Joshep Andersson Blanco Reyes

BigQuery

De Google

BigQuery es un almacén de datos de análisis de bajo coste administrado íntegramente por Google y en el que se utiliza una escala de petabytes. Este servicio se cobra por consulta o procesamiento de datos, y el importe de esas consultas se carga en la tarjeta de crédito del proyecto de facturación.

MÁS INFORMACIÓN
NOTIFICAR UN PROBLEMA

PROYECTOS RECIENTES	Proyecto	Conjunto de datos	Tabla
MIS PROYECTOS	Introduzca el ID del proyecto manualmente	SPOTIFY	artista
PROYECTOS COMPARTIDOS	My First Project	spotify1	canciones1
CONSULTA PERSONALIZADA			genero