**Evidencias de documentación**

**Data Warehouse HVR**

# 

Contenido

[**Objetivo 2**](#_ts9sm56zssda)

[Equipo del Proyecto 2](#_1qrlh7vntrvg)

[Metodología 3](#_wcmedcjlc0fn)

[**Alcance y Objetivos del Proyecto 4**](#_vri8tdnqcokf)

[**Cronograma de Sprints 5**](#_7lu12kqr4l8r)

[**Retrospectiva del Sprint 8**](#_jzqgh713nxz9)

[**Métricas del Proyecto 10**](#_q73couns574)

[**Actualización de Documentación Técnica 12**](#_p0cxcttxq819)

[Arquitectura 12](#_ca11uo9eid5s)

[Documentación Sprint 0 12](#_wapy5wbi90p6)

[Configuración de Buckets en Cloud Storage 12](#_qzjcf4sxwtez)

[Creación de conjuntos de datos en BigQuery 17](#_81aehkn1gllw)

[Documentación Sprint 1 28](#_u8bo8j36byi7)

[Crear un catálogo detallado de todas las tablas relevantes del ERP 28](#_6pqep4qnmb3e)

[Creación de Cloud Functions para la carga de datos en BigQuery 33](#_kf5423uq17in)

[Implementación de Dataform para gestión de transformaciones en BigQuery 36](#_uwo4edbfs052)

[Configuración de git en Dataform para gestión de transformaciones en BigQuery 41](#_62xa1bfuqclg)

[Realizar pruebas de conexión desde el servidor local a GCP 43](#_jybgj3lcoa2a)

# 

# 

# Objetivo

Diseñar y desarrollar un Data Warehouse alojado en GCP que optimice la carga diaria de información, entrenar un modelo ML predictivo para un caso de negocio y la generación de informes y dashboards de la empresa HVR.

# Equipo del Proyecto

| **Integrantes** | **Rol** | **Responsabilidad** |
| --- | --- | --- |
| Sebastián Orellana | Data Scientist | Responsable de explorar, analizar y modelar los datos para extraer patrones y generar insights útiles para el negocio. |
| Ignacio Pi | Scrum Master | Responsable de establecer la metodología Scrum y mantener a los miembros del equipo enfocados en los principios y las prácticas de Scrum. |
| Daniela Galvéz | Devops | Responsable de automatizar y gestionar la infraestructura y los despliegues del proyecto. |
| Kiara Caycho | Data Engineer | Diseña, desarrolla y mantiene los sistemas que procesan grandes volúmenes de datos, |

# Metodología

Se utilizará la metodología ágil scrum para entregas iterativas del proyecto, estarán organizadas en 6 sprints con una duración aproximada de 2 semanas cada una, a diferencia del primer y último sprint que serán los más cortos (1 semana), las cuales tienen sus historias asignadas por etapa del proyecto.

Las Dailies serán realizadas 3 veces a la semana y una reunión de refinamiento en caso de que haya que resolver dudas.

Las retrospectivas se realizan al finalizar un sprint, detallando el avance de este para ver posibles mejoras en el próximo.

La evaluación de las historias será con un puntaje en tallas de poleras, una técnica para estimar el esfuerzo de cada una en lugar de la enumeración tradicional.

Puntos de Historia:

* XS (Extra Small): Tarea muy pequeña, que puede completarse rápidamente. (1-2 puntos)
* S (Small): Tarea pequeña, que no es complicada pero lleva algo de tiempo. (3-5 puntos)
* M (Medium): Tarea de tamaño medio, que podría necesitar varias horas o incluso un día de trabajo. (6-8 puntos)
* L (Large): Tarea grande o compleja, que requerirá varios días de trabajo. (9-13 puntos)
* XL (Extra Large): Tarea muy grande y compleja, probablemente requiera más de una iteración o varios días de trabajo con posibles riesgos. (14 puntos o más)

# 

# Alcance y Objetivos del Proyecto

El proyecto busca construir un data warehouse en Google Cloud Platform (GCP) para la carga y transformación de datos a BigQuery. Incluirá la creación de un flujo de datos con capas raw, silver y gold, el uso de BigQuery ML para entrenar un modelo de regresión lineal, y la generación de informes de BI personalizados mediante Looker Studio.

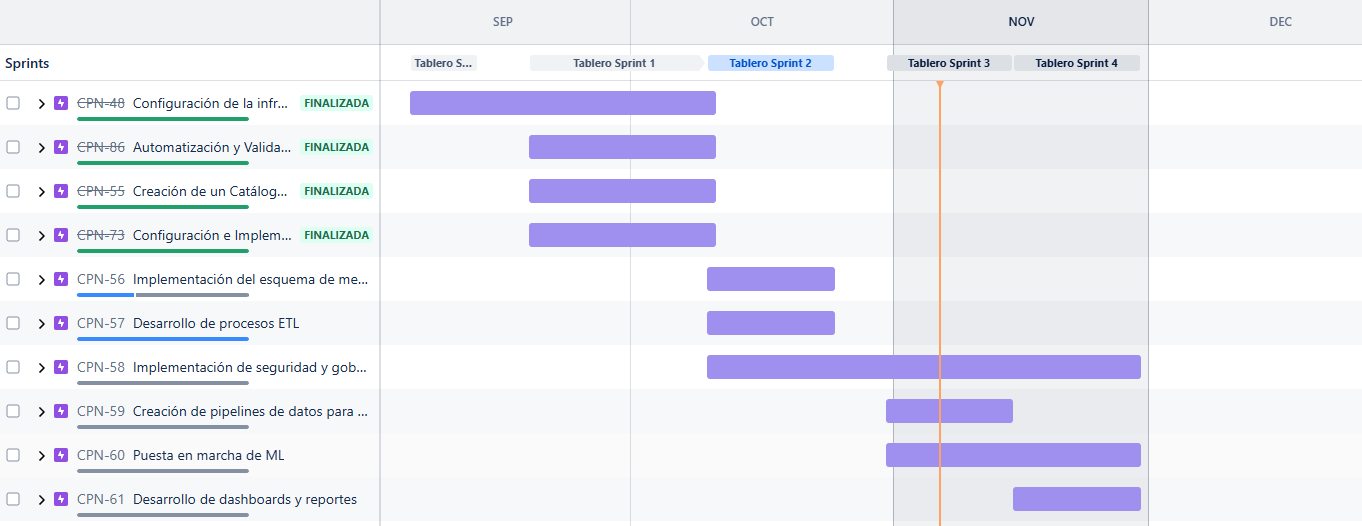
Dentro de los objetivos se encuentran los siguientes puntos:

* Configuración de la infraestructura en GCP: Configuración de buckets, políticas de retención de almacenamiento,conjuntos de datos en BigQuery, permisos en BigQuery, Cloud Functions para la carga de datos, Configuración de git en Dataform, BigQuery ML para Machine Learning, Looker Studio para dashboards interactivos, roles y permisos (IAM), etc.
* Extracción de Datos en CSV: Extraer datos en formato CSV desde un servidor SQL Server externo hacia Google Cloud Storage.
* Optimización del Proceso de ETL: Cargar datos desde los CSV y transformarlos en BigQuery a través de distintas capas (raw, silver y gold).
* Creación de un Modelo Estrella: Organizar los datos en la capa gold usando un modelo estrella que optimice las consultas y el análisis.
* Entrenamiento de un Modelo de ML: Usar BigQuery ML para entrenar un modelo de regresión lineal sobre los datos transformados.
* Visualización y Reporting: Generar informes dinámicos en Looker Studio con roles de acceso controlados.
* Seguridad, Monitoreo y Alertas: Garantizar la protección de los datos y el monitoreo continuo de los flujos de trabajo, con alertas ante fallos o comportamientos anómalos.
* Gestión de Metadatos: Utilizar Data Catalog para organizar, clasificar y documentar los datasets y tablas, facilitando su búsqueda y gestión.
* Automatización y Orquestación: Implementar Google Cloud Workflows para gestionar y automatizar el flujo de trabajo de ETL.
* Almacenamiento Intermedio: Utilizar Google Cloud Storage (GCS) para almacenar temporalmente datos intermedios durante el proceso de ETL.

En este enlace se detallan las historias de usuarios del proyecto junto a su criterio de aceptación

[Sprint - Product Backlog - Datawarehouse HVR](https://docs.google.com/spreadsheets/d/1mYHkbl4eTBivwpUys9payER6KD-f7_ddYFtaPFfWhqs/edit?usp=sharing)

# Cronograma de Sprints



**Sprint 0**

**Fecha de duración:** 9 al 12 de septiembre(1 semana)

**Historias de usuario:**

* Configuración de buckets en Cloud Storage
* Configuración de políticas de retención en Cloud Storage
* Creación de conjuntos de datos en BigQuery
* Configuración de permisos en BigQuery
* Configurar Identity and Access Management (IAM) para los roles y permisos
* Realizar pruebas de conexión desde el servidor local a GCP
* Configuración de roles y permisos en IAM para el data warehouse

**Sprint 1**

**Fecha de duración:** 19 de septiembre al 10 de octubre (4 semanas)

**Historias de usuario:**

* Creación de Cloud Functions para la carga de datos en BigQuery
* Pruebas de ejecución de Cloud Functions
* Crear un catálogo detallado de todas las tablas relevantes del ERP
* Configuración de git en Dataform para gestión de transformaciones en BigQuery
* Implementación de Dataform para gestión de transformaciones en BigQuery

**Sprint 2:**

**Fecha de duración:** 10 al 24 de octubre (3 semanas)

**Historias de usuario:**

* Diseño de la capa Bronze en BigQuery
* Mapeo de datos entre el ERP y la capa Bronze
* Auditoría de accesos en IAM para el data warehouse
* Desarrollo de procesos de limpieza y estandarización en la capa Silver
* Validación de calidad de datos en la capa Silver
* Creación de scripts y pipelines ETL en Dataform
* Ejecución de transformaciones en BigQuery mediante Dataform
* Despliegue de Dataform para orquestación de procesos ETL

**Sprint 3:**

**Fecha de duración:** 31 de octubre 14 noviembre (2 semanas)

**Historias de usuario:**

* Preparación de datasets para Machine Learning
* Partición de datasets para Machine Learning
* Creación de vistas y tablas agregadas en la capa Gold
* Configuración de BigQuery ML para Machine Learning
* Evaluación y ajuste de modelos de Machine Learning
* Optimización del rendimiento de consultas en la capa Gold
* Configuración de BigQuery ML para entrenar modelos
* Despliegue de modelos de Machine Learning

**Sprint 4:**

**Fecha de duración:** 15 al 29 de noviembre (3 semanas)

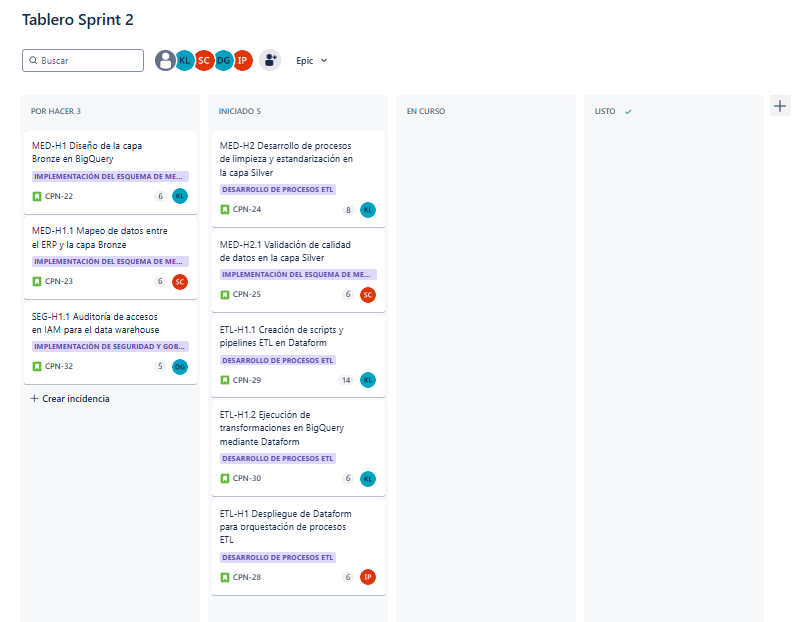
**Historias de usuario:**

* Configuración de Pub/Sub para notificaciones en tiempo real
* Configuración de Looker Studio para dashboards interactivos
* Configuración de Security Command Center en GCP
* Configuración de Cloud Monitoring para la supervisión de infraestructura
* Creación de un dashboard ejecutivo en Looker Studio
* Configuración de actualización automática del dashboard ejecutivo

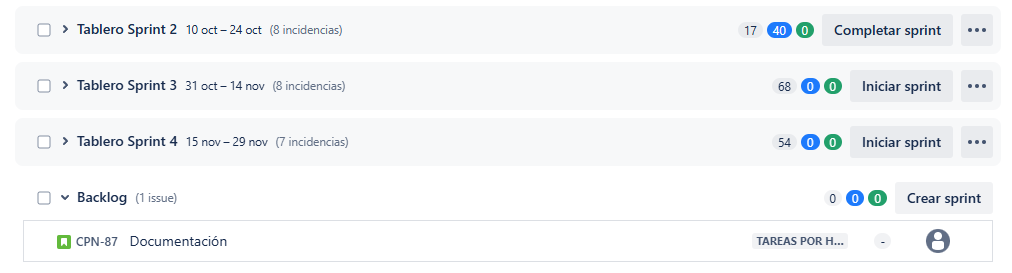
Detalle de cada historia aquí : [Historias de Usuario - Datawarehouse HVR](https://docs.google.com/document/d/1uH3gw1GPqTUcMZyPeZzadH9z1ANXfSDfqMOkzWrPPe8/edit?tab=t.0)

Detalle de cada sprint e historia realizada y por hacer aquí: [Jira](https://hvrinformes.atlassian.net/jira/software/projects/CPN/boards/8?atlOrigin=eyJpIjoiMjA4OWVmMjdjZjU3NGFkZTg2ZjI1NGI4YWIyZmM4YWQiLCJwIjoiaiJ9)

[Sprint - Product Backlog - Datawarehouse HVR](https://docs.google.com/spreadsheets/d/1mYHkbl4eTBivwpUys9payER6KD-f7_ddYFtaPFfWhqs/edit?usp=sharing)



Detalle de Backlog en jira [aquí](https://hvrinformes.atlassian.net/jira/software/projects/CPN/boards/8/backlog)

[Sprint - Product Backlog - Datawarehouse HVR](https://docs.google.com/spreadsheets/d/1mYHkbl4eTBivwpUys9payER6KD-f7_ddYFtaPFfWhqs/edit?gid=254126892#gid=254126892)

# Retrospectiva del Sprint

**Sprint 0**

**Estado: Completo**

En este sprint se realizaron las configuraciones de la infraestructura del proyecto HVR en GCP.

| **Empezar a hacer** | **Dejar de hacer** | **Seguir haciendo** |
| --- | --- | --- |
| - Configuración de la infraestructura del proyecto HVR  - Creación de proyecto en GCP.  - Configuración de Buckets.  - Creación de conjunto de datos en BigQuery.  - Configuración de políticas de retención en Cloud Storage  - Configuración de permisos en BigQuery  - Configurar Identity and Access Management (IAM) para los roles y permisos  - Configuración de roles y permisos en IAM para el data warehouse | - Programación de reuniones diarias, solo 3 veces a la semana.  - Configuración de políticas de retención en Cloud Storage, para evitar bloqueo del flujo de carga de datos en cloud storage. | - Reuniones 3 veces a la semana.  - Avance de las tareas asignadas en el sprint 0. |

Retrospectiva

| **Participantes** | **¿Qué ha ido bien en el sprint?** | **¿Qué se puede mejorar?** | **¿Qué ha fallado?** |
| --- | --- | --- | --- |
| Daniela Gálvez | Se crean los perfiles según cargo y se asignan roles según acceso y permisos | Eficiencia en la ejecución de lo propuesto. | Organización. |
| Ignacio Pi | Se configuran los servicios gcp iniciales y se prepara la carga de datos del servidor hvr. | Planificar las tareas con mayor especificidad | Planificación previa |
| Sebastián orellana | Creación exitosa de la infraestructura de Storage y BigQuery | Organización de reuniones de equipo. | Reestructuración de los Bucket iniciales. |
| Kiara Caycho | Se completaron todas las tareas. | Mejorar el tiempo de ejecución de las tareas. | Ejecución de las tareas a tiempo. |

**Sprint 1:**

**Estado: Completo**

En este sprint se dio inicio al registro de las tablas con las que se trabajará durante el flujo del Data Warehouse de HVR, creación de las cloud function, implementación de dataform y configuración con github.

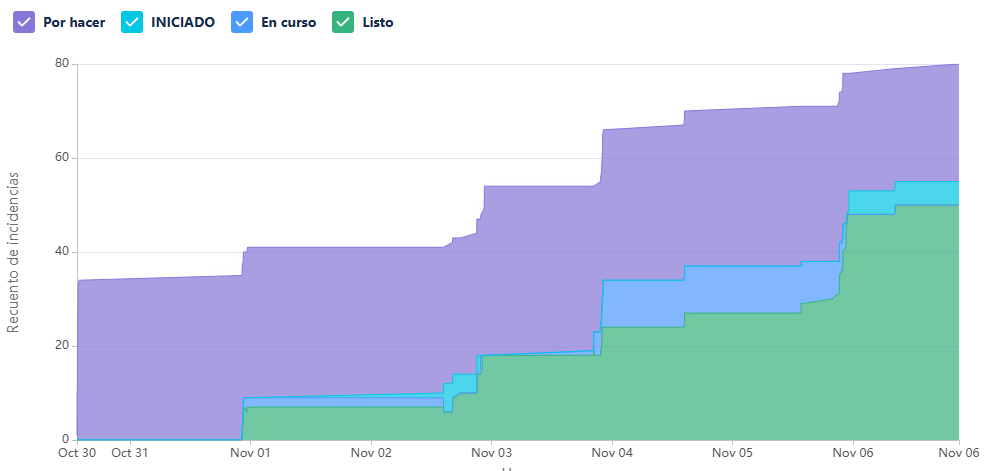
| **Empezar a hacer** | **Dejar de hacer** | **Seguir haciendo** |
| --- | --- | --- |
| - Creación de catálogo de datos de las tablas.  - Creación de Cloud function para la carga de datos en bigquery.  - Implementación de dataform.  - Configuración entre github y dataform. |  | Todas las tareas asignadas |

Retrospectiva

| **Participantes** | **¿Qué ha ido bien en el sprint?** | **¿Qué se puede mejorar?** | **¿Qué ha fallado?** |
| --- | --- | --- | --- |
| Daniela Gálvez | Configuración de políticas de retención y recuperación de datos en bucket. | Sin comentarios. | Retraso en la documentación de las tareas. |
| Ignacio Pi | El flujo de transformación entre las capas Bronze y Silver fue validado sin errores críticos, asegurando una base sólida para la siguiente etapa. | La eficiencia en la entrega de tareas. | Hubo retrasos en la implementación de dataform. Lo que detuvo todo el progreso del proyecto al ser un punto crítico. |
| Sebastián orellana |  |  |  |
| Kiara Caycho | - Documentación completa del registro de las tablas .  - Implementación exitosa de dataform. | El manejo de bloqueos a tiempo. | Hubo bloqueos en la implementación de dataform, pero se soluciona de manera exitosa. |

# Métricas del Proyecto

Diagrama de flujo acumulado, la distancia entre las líneas de cada columna te muestra cuánto tardan las incidencias en pasar de un estado a otro. Busca los puntos en los que una banda crece a una velocidad más rápida que otra para localizar los cuellos de botella

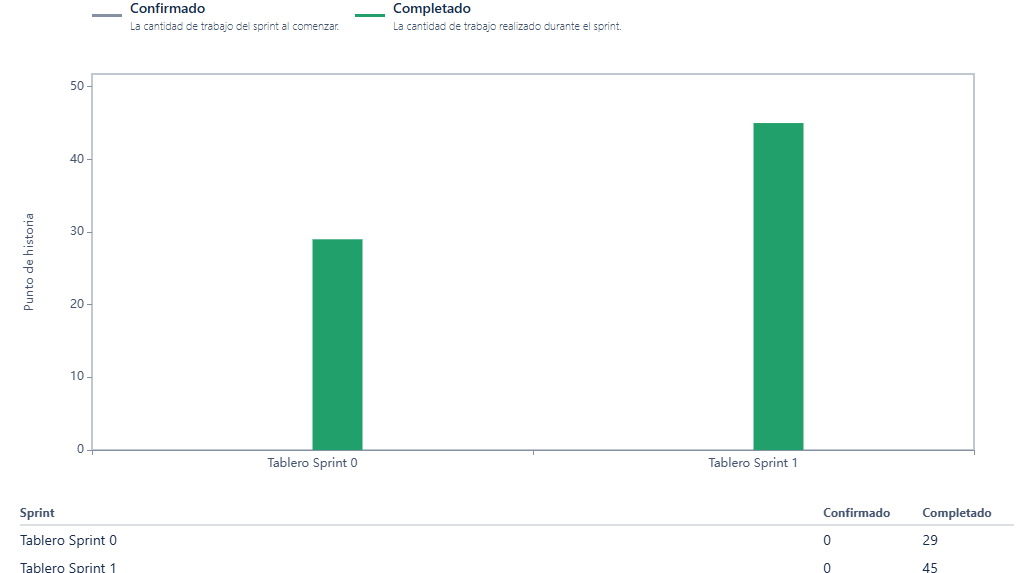


Burndown Chart

Gráfica de la velocidad en relación con el progreso de las tareas del sprint.

Gris: la estimación total de las incidencias de cada sprint cuando éste se inició.

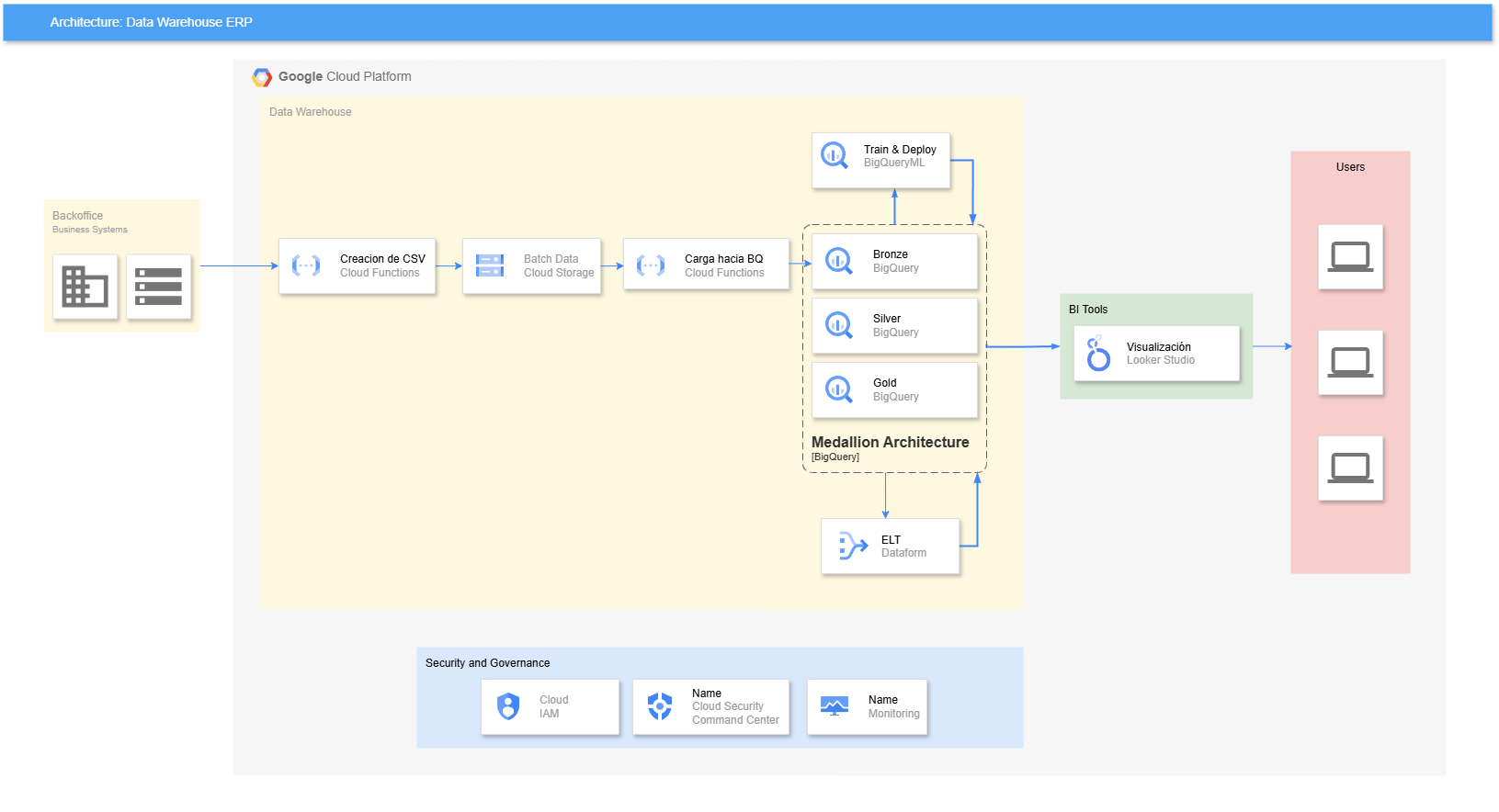
Verde: las estimaciones totales completadas cuando finalizó el sprint.



# Actualización de Documentación Técnica

## 

## Arquitectura



## 

A Continuación se detallarán las documentaciones realizadas por sprint, cada una pertenece a una historia

## Documentación Sprint 0

## Configuración de Buckets en Cloud Storage

Para realizar la creación de los Buckets en Cloud Storage daremos inicio al proyecto HVR en GCP.

Para el proyecto se implementarán 3 ambientes de trabajo con el objetivo de separar bien los roles asignados dentro del proyecto y proteger los datos.

**Ambiente de desarrollo(Dev)**

Aquí los desarrolladores podrán probar las modificaciones o integraciones sin el riesgo de afectar los datos reales del proyecto, realizando ajustes de códigos, configuraciones, etc.

**Ambiente QA**

En este ambiente se realizarán pruebas evaluando si los cambios realizados están bien y no generen errores a futuro.

**Ambiente Producción (Prod)**

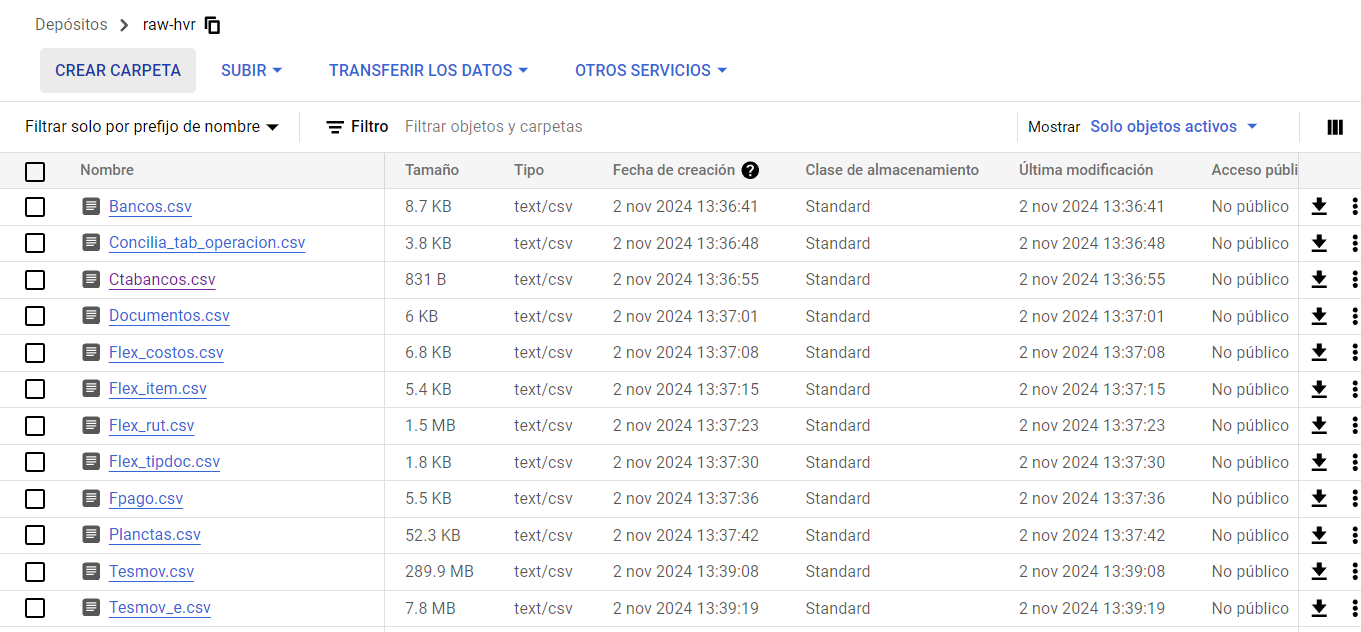
Es el paso final para implementar los cambios realizados en un código, una vez que hayan pasado por Dev y QA finalmente en producción se desplegarán las modificaciones sin riesgo de afectar los datos del proyecto.

Dentro de cada ambiente se crearán los buckets correspondientes a “raw\_data\_hvr", "error\_raw\_data\_hvr", “temp\_data\_hvr“, “clean\_data\_hvr” y “history\_data“ dentro de Cloud Storage

**Bucket raw\_data\_hvr\_dev**

En este bucket se almacenarán todos los datos que llegan desde el servidor de la empresa HVR, los archivos que se cargarán serán en formatos csv.

Al ser cargados automáticamente una cloud function realizará el proceso de limpieza y carga de datos a Big Query dentro del dataset Bronze.



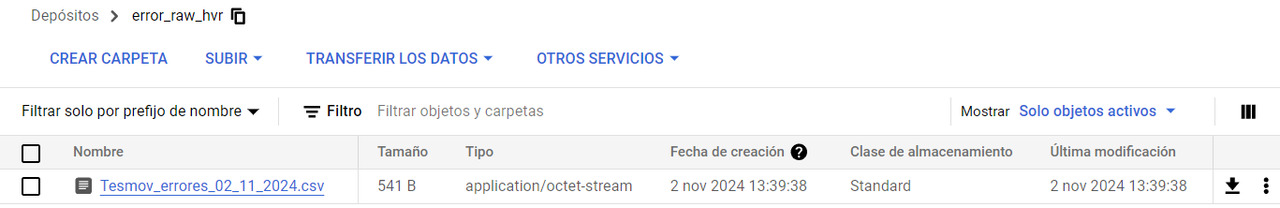
**Bucket error\_raw\_data**

En este bucket se almacenarán los datos erróneos de carga de datos.

Si un archivo tiene un error en el proceso de limpieza automáticamente se genera un archivo csv, el cual es guardado en este bucket con el nombre del documento más la fecha de ejecución.

Dentro de este archivo se registra la fila donde se debería encontrar el dato erróneo que no se pudo cargar a la tabla correspondiente.

En la imagen podemos visualizar que el archivo “Tesmov\_errores\_02\_11\_2024.csv” tiene errores registrados.

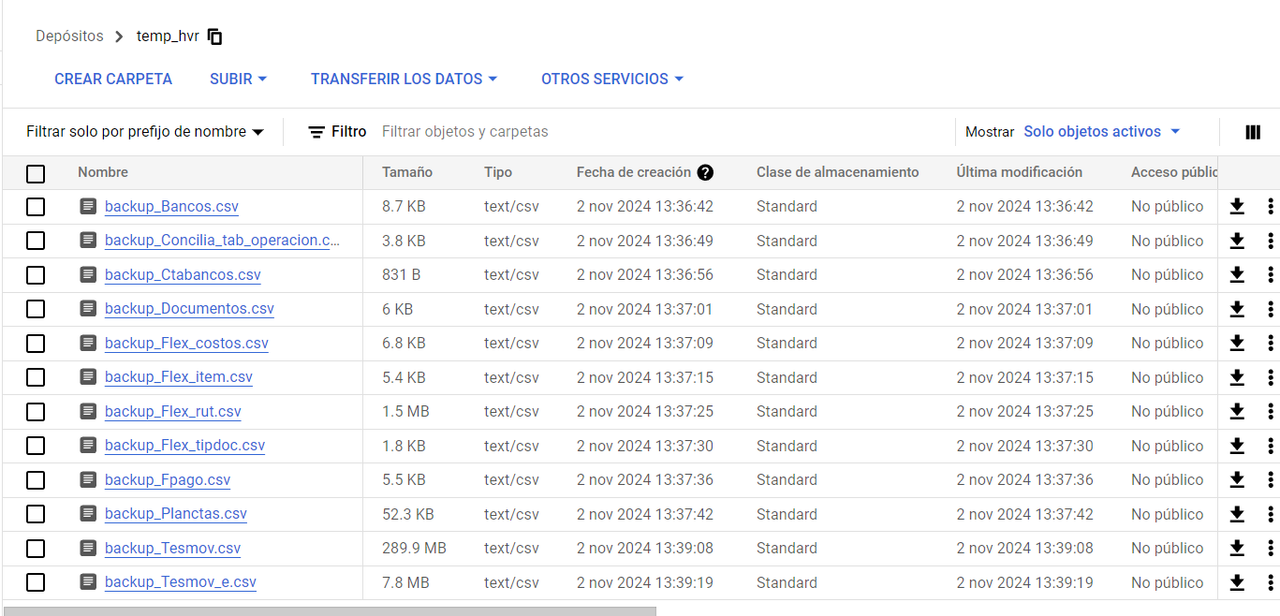


El error registrado se visualiza con los datos de la fila completa, el cual se debería analizar de forma manual para que se solucione y vuelva a pasar por todo el proceso de limpieza y carga de datos.



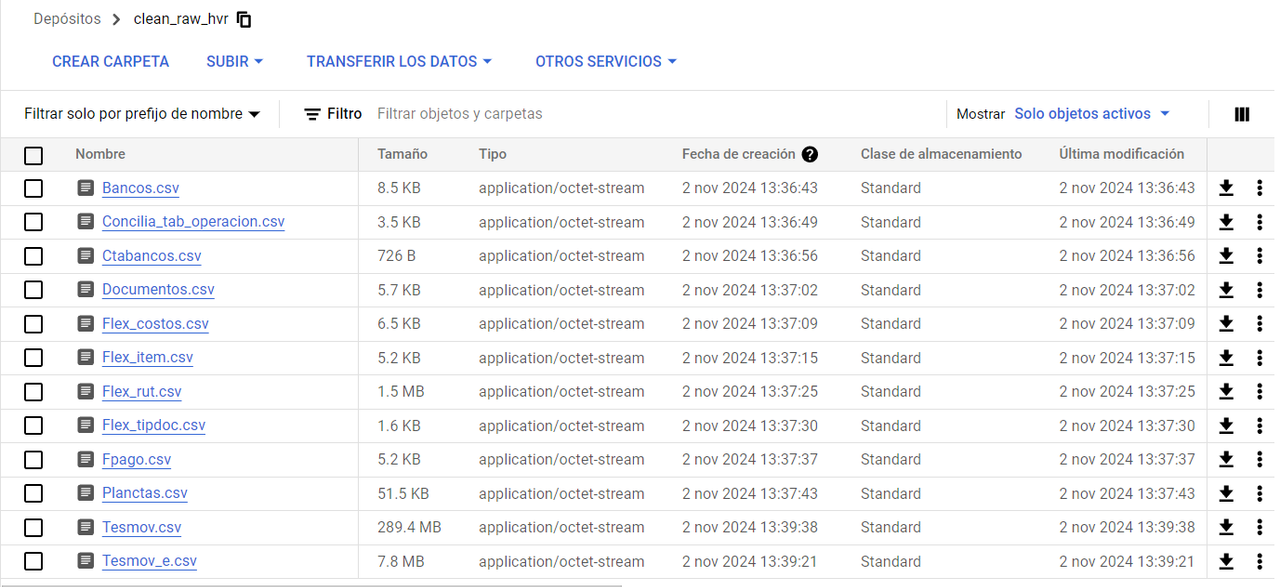
**Bucket temp\_data**

En este Bucket se guardan archivos temporales mientras se va realizando el proceso de limpieza y carga de datos, se crea una tabla temporal en el dataset Bronze “ctabancos\_bronze\_temp“, el cual se elimina cuando está listo el proceso de carga de datos quedando con el nombre “ctabancos\_bronze”.



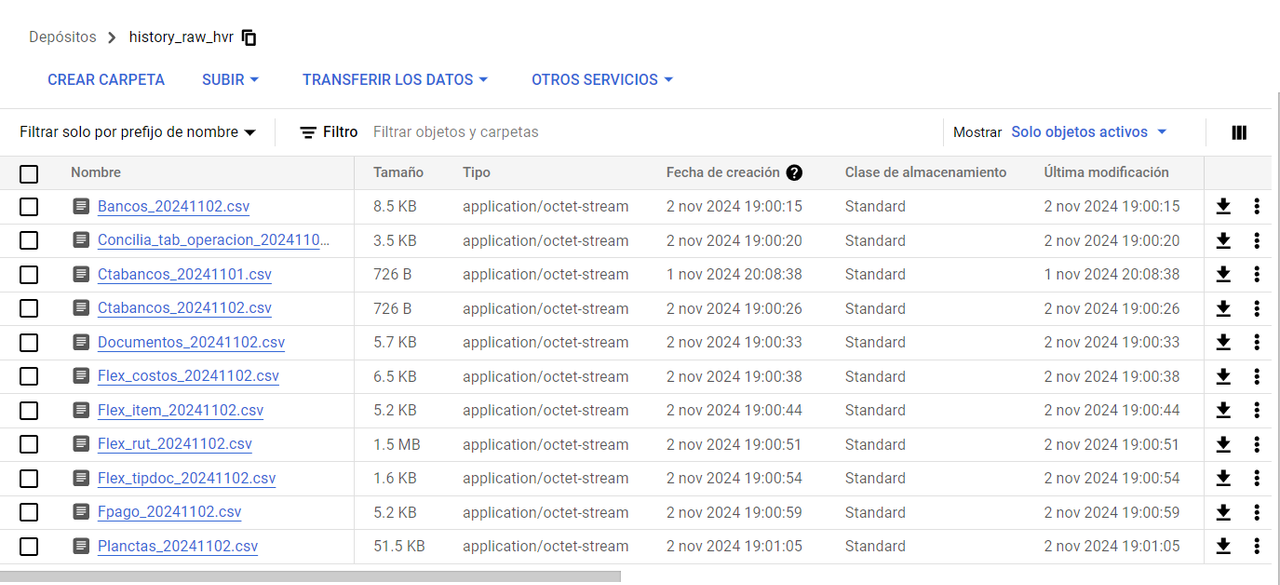
**Bucket clean\_data**

Se almacenan los archivos que hayan pasado por el proceso de limpieza en csv.



**Bucket history\_data**

Se almacenan los csv procesados y almacenados en el dataset bronze con el nombre y fecha de ejecución.



**Cloud Function limpieza\_csv**

El proceso de limpieza es el siguiente:

* Elimina caracteres nulos y verificando el número de columnas
* Elimina caracteres especiales con encoding='ISO-8859-1 esta codificación incluye caracteres especiales como la ñ, acentos (á, é, í, ó, ú) y caracteres de puntuación específicos de las lenguas occidentales europeas, como el español, francés, alemán, italiano, entre otros.
* Elimina segunda línea si contiene "---" y última línea si tiene "(" y ")"
* Mueve la última columna del encabezado al comienzo, columna id de la tabla.
* Guardar archivo limpio y validado en bucket “temp\_hvr”.
* Guardar líneas con errores si existen en el bucket “error\_raw\_hvr”.

## Creación de conjuntos de datos en BigQuery

Para el proyecto se implementarán 3 ambientes de trabajo con el objetivo de separar bien los roles asignados dentro del proyecto y proteger los datos.

Ambiente de desarrollo(Dev)

Aquí los desarrolladores podrán probar las modificaciones o integraciones sin el riesgo de afectar los datos reales del proyecto, realizando ajustes de códigos, configuraciones, etc.

Ambiente QA

En este ambiente se realizarán pruebas evaluando si los cambios realizados están bien y no generen errores a futuro.

Ambiente Producción (Prod)

Es el paso final para implementar los cambios realizados en un código, una vez que hayan pasado por Dev y QA finalmente en producción se desplegarán las modificaciones sin riesgo de afectar los datos del proyecto.

Nomenclaturas para cada ambiente

Estos serán los nombres asignados por cada ambiente de trabajo dentro del proyecto en GCP

| Ambiente | Nombre proyecto GCP |
| --- | --- |
| Desarrollo | hvr-dwh-dev |
| QA | hvr-dwh-qa |
| Producción | hvr-dwh-prod |

El objetivos para su implementación:

* Mejor organización y estructura estándar dentro del proyecto para poder acceder fácilmente a los datos.
* Escalabilidad del proyecto a futuro, esto ayuda a crear nuevos dataset y tablas con un patrón claro y definido.
* Ayuda a poder definir bien los accesos a tablas definidas por IAM
* Fácil de rastrear cambios y movimientos de los datos.
* Evita confusión y error al interactuar con los datos.
* Establece un contexto sobre el contenido de los datos, esto permite a los usuario comprender mejor el significado de cada dataset o tabla.

Ambiente Desarrollo: dataset bronze

| Nombre proyecto GCP | Dataset bronze | Tabla |
| --- | --- | --- |
| hvr-dwh-dev-437001 | dataset-bronze | bancos\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | concilia\_tab\_operacion\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | ctabancos\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | documentos\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | documentos\_b\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | flex\_costos\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | flex\_item\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | flex\_rut\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | flex\_tipdoc\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | fpago\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | planctas\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | tesmov\_e\_bronze |
| hvr-dwh-dev-437001 | dataset-bronze | tesmov\_bronze |
| hvr-dwh-dev-437001.dataset\_silver.flex\_rut\_bronze | | |

Ambiente Desarrollo: dataset silver

| Nombre proyecto GCP | Dataset silver | Tabla |
| --- | --- | --- |
| hvr-dwh-dev-437001 | dataset-silver | bancos\_silver |
| hvr-dwh-dev-437001 | dataset-silver | concilia\_tab\_operacion\_silver |
| hvr-dwh-dev-437001 | dataset-silver | ctabancos\_silver |
| hvr-dwh-dev-437001 | dataset-silver | documentos\_silver |
| hvr-dwh-dev-437001 | dataset-silver | documentos\_b\_silver |
| hvr-dwh-dev-437001 | dataset-silver | flex\_costos\_silver |
| hvr-dwh-dev-437001 | dataset-silver | flex\_item\_silver |
| hvr-dwh-dev-437001 | dataset-silver | flex\_rut\_silver |
| hvr-dwh-dev-437001 | dataset-silver | flex\_tipdoc\_silver |
| hvr-dwh-dev-437001 | dataset-silver | fpago\_silver |
| hvr-dwh-dev-437001 | dataset-silver | planctas\_silver |
| hvr-dwh-dev-437001 | dataset-silver | tesmov\_e\_silver |
| hvr-dwh-dev-437001 | dataset-silver | tesmov\_silver |
| hvr-dwh-dev-437001.dataset\_silver.flex\_rut\_silver | | |

Ambiente Desarrollo: dataset gold

| Nombre proyecto GCP | Dataset gold | Tabla |
| --- | --- | --- |
| hvr-dwh-dev-437001 | dataset-gold | bancos\_gold |
| hvr-dwh-dev-437001 | dataset-gold | concilia\_tab\_operacion\_gold |
| hvr-dwh-dev-437001 | dataset-gold | ctabancos\_gold |
| hvr-dwh-dev-437001 | dataset-gold | documentos\_gold |
| hvr-dwh-dev-437001 | dataset-gold | documentos\_b\_gold |
| hvr-dwh-dev-437001 | dataset-gold | flex\_costos\_gold |
| hvr-dwh-dev-437001 | dataset-gold | flex\_item\_gold |
| hvr-dwh-dev-437001 | dataset-gold | flex\_rut\_gold |
| hvr-dwh-dev-437001 | dataset-gold | flex\_tipdoc\_gold |
| hvr-dwh-dev-437001 | dataset-gold | fpago\_gold |
| hvr-dwh-dev-437001 | dataset-gold | planctas\_gold |
| hvr-dwh-dev-437001 | dataset-gold | tesmov\_e\_gold |
| hvr-dwh-dev-437001 | dataset-gold | tesmov\_gold |
| hvr-dwh-dev-437001.dataset\_silver.flex\_rut\_gold | | |

Ambiente QA: dataset bronze

| Nombre proyecto GCP | Dataset bronze | Tabla |
| --- | --- | --- |
| hvr-dwh-qa-437001 | dataset-bronze | bancos\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | concilia\_tab\_operacion\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | ctabancos\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | documentos\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | documentos\_b\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | flex\_costos\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | flex\_item\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | flex\_rut\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | flex\_tipdoc\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | fpago\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | planctas\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | tesmov\_e\_bronze |
| hvr-dwh-qa-437001 | dataset-bronze | tesmov\_bronze |
| hvr-dwh-qa-437001.dataset\_silver.flex\_rut\_bronze | | |

Ambiente QA: dataset silver

| Nombre proyecto GCP | Dataset silver | Tabla |
| --- | --- | --- |
| hvr-dwh-qa-437001 | dataset-silver | bancos\_silver |
| hvr-dwh-qa-437001 | dataset-silver | concilia\_tab\_operacion\_silver |
| hvr-dwh-qa-437001 | dataset-silver | ctabancos\_silver |
| hvr-dwh-qa-437001 | dataset-silver | documentos\_silver |
| hvr-dwh-qa-437001 | dataset-silver | documentos\_b\_silver |
| hvr-dwh-qa-437001 | dataset-silver | flex\_costos\_silver |
| hvr-dwh-qa-437001 | dataset-silver | flex\_item\_silver |
| hvr-dwh-qa-437001 | dataset-silver | flex\_rut\_silver |
| hvr-dwh-qa-437001 | dataset-silver | flex\_tipdoc\_silver |
| hvr-dwh-qa-437001 | dataset-silver | fpago\_silver |
| hvr-dwh-qa-437001 | dataset-silver | planctas\_silver |
| hvr-dwh-qa-437001 | dataset-silver | tesmov\_e\_silver |
| hvr-dwh-qa-437001 | dataset-silver | tesmov\_silver |
| hvr-dwh-qa-437001.dataset\_silver.flex\_rut\_silver | | |

Ambiente QA: dataset gold

| Nombre proyecto GCP | Dataset gold | Tabla |
| --- | --- | --- |
| hvr-dwh-qa-437001 | dataset-gold | bancos\_gold |
| hvr-dwh-qa-437001 | dataset-gold | concilia\_tab\_operacion\_gold |
| hvr-dwh-qa-437001 | dataset-gold | ctabancos\_gold |
| hvr-dwh-qa-437001 | dataset-gold | documentos\_gold |
| hvr-dwh-qa-437001 | dataset-gold | documentos\_b\_gold |
| hvr-dwh-qa-437001 | dataset-gold | flex\_costos\_gold |
| hvr-dwh-qa-437001 | dataset-gold | flex\_item\_gold |
| hvr-dwh-qa-437001 | dataset-gold | flex\_rut\_gold |
| hvr-dwh-qa-437001 | dataset-gold | flex\_tipdoc\_gold |
| hvr-dwh-qa-437001 | dataset-gold | fpago\_gold |
| hvr-dwh-qa-437001 | dataset-gold | planctas\_gold |
| hvr-dwh-qa-437001 | dataset-gold | tesmov\_e\_gold |
| hvr-dwh-qa-437001 | dataset-gold | tesmov\_gold |
| hvr-dwh-qa-437001.dataset\_silver.flex\_rut\_gold | | |

Ambiente Producción: dataset bronze

| Nombre proyecto GCP | Dataset bronze | Tabla |
| --- | --- | --- |
| hvr-dwh-prod-437001 | dataset-bronze | bancos\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | concilia\_tab\_operacion\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | ctabancos\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | documentos\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | documentos\_b\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | flex\_costos\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | flex\_item\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | flex\_rut\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | flex\_tipdoc\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | fpago\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | planctas\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | tesmov\_e\_bronze |
| hvr-dwh-prod-437001 | dataset-bronze | tesmov\_bronze |
| hvr-dwh-prod-437001.dataset\_silver.flex\_rut\_bronze | | |

Ambiente Producción: dataset silver

| Nombre proyecto GCP | Dataset silver | Tabla |
| --- | --- | --- |
| hvr-dwh-prod-437001 | dataset-silver | bancos\_silver |
| hvr-dwh-prod-437001 | dataset-silver | concilia\_tab\_operacion\_silver |
| hvr-dwh-prod-437001 | dataset-silver | ctabancos\_silver |
| hvr-dwh-prod-437001 | dataset-silver | documentos\_silver |
| hvr-dwh-prod-437001 | dataset-silver | documentos\_b\_silver |
| hvr-dwh-prod-437001 | dataset-silver | flex\_costos\_silver |
| hvr-dwh-prod-437001 | dataset-silver | flex\_item\_silver |
| hvr-dwh-prod-437001 | dataset-silver | flex\_rut\_silver |
| hvr-dwh-prod-437001 | dataset-silver | flex\_tipdoc\_silver |
| hvr-dwh-prod-437001 | dataset-silver | fpago\_silver |
| hvr-dwh-prod-437001 | dataset-silver | planctas\_silver |
| hvr-dwh-prod-437001 | dataset-silver | tesmov\_e\_silver |
| hvr-dwh-prod-437001 | dataset-silver | tesmov\_silver |
| hvr-dwh-prod-437001.dataset\_silver.flex\_rut\_silver | | |

Ambiente Producción: dataset gold

| Nombre proyecto GCP | Dataset gold | Tabla |
| --- | --- | --- |
| hvr-dwh-prod-437001 | dataset-gold | bancos\_gold |
| hvr-dwh-prod-437001 | dataset-gold | concilia\_tab\_operacion\_gold |
| hvr-dwh-prod-437001 | dataset-gold | ctabancos\_gold |
| hvr-dwh-prod-437001 | dataset-gold | documentos\_gold |
| hvr-dwh-prod-437001 | dataset-gold | documentos\_b\_gold |
| hvr-dwh-prod-437001 | dataset-gold | flex\_costos\_gold |
| hvr-dwh-prod-437001 | dataset-gold | flex\_item\_gold |
| hvr-dwh-prod-437001 | dataset-gold | flex\_rut\_gold |
| hvr-dwh-prod-437001 | dataset-gold | flex\_tipdoc\_gold |
| hvr-dwh-prod-437001 | dataset-gold | fpago\_gold |
| hvr-dwh-prod-437001 | dataset-gold | planctas\_gold |
| hvr-dwh-prod-437001 | dataset-gold | tesmov\_e\_gold |
| hvr-dwh-prod-437001 | dataset-gold | tesmov\_gold |
| hvr-dwh-prod-437001.dataset\_silver.flex\_rut\_gold | | |

Machine learning

| Nombre proyecto GCP | Dataset gold | Tabla |
| --- | --- | --- |
| hvr-dwh-prod-437001 | dataset-ml | tabla\_ml |
| hvr-dwh-prod-437001 | dataset-ml | tabla2\_ml |
| hvr-dwh-prod-437001.dataset\_silver.tabla\_ml | | |

## Documentación Sprint 1

## Crear un catálogo detallado de todas las tablas relevantes del ERP

Las tablas con las que se trabajarán serán las siguientes

* bancos
* concilia\_tab\_operacion
* ctabancos
* documentos
* documentos\_b
* flex\_costos
* flex\_item
* flex\_rut
* flex\_tipdoc
* fpago
* planctas
* tesmov\_e
* tesmov

Todas las tablas tienen tipo de dato STRING en la capa bronze, las cuales tendrán una transformación de datos en la capa silver, estas serán las que se registrarán a continuación

Tabla bancos

| Nombre de campo | Modo | Tipo de campo | Descripción |
| --- | --- | --- | --- |
| ID\_Bancos | REQUIRED | STRING | Código del banco |
| Empresa | NULLABLE | STRING | Nombre o identificación de la empresa |
| Cod | NULLABLE | STRING | Código del banco o entidad |
| Des | NULLABLE | STRING | Descripción del banco o entidad |
| tipo | NULLABLE | STRING | Tipo de banco o entidad |
| RutLeg | NULLABLE | STRING | RUT legal asociado |
| cod\_inten | NULLABLE | STRING | Código interno |
| suc\_bco | NULLABLE | STRING | Sucursal del banco |
| comuna | NULLABLE | STRING | Comuna en la que se encuentra el banco |
| direccion | NULLABLE | STRING | Dirección física del banco |
| telefono | NULLABLE | STRING | Número de teléfono del banco |
| contacto | NULLABLE | STRING | Persona de contacto en el banco |
| carg\_con | NULLABLE | STRING | Cargo del contacto en el banco |
| item | NULLABLE | STRING | Ítem asociado |
| marca | NULLABLE | STRING | Marca del banco |
| cuenta | NULLABLE | STRING | Número de cuenta bancaria |
| Log\_usr | NULLABLE | STRING | Usuario que realizó el último cambio |
| Log\_fec | NULLABLE | DATETIME | Fecha del último cambio |
| AUDIT\_INSERT\_DTTM | REQUIRED | TIMESTAMP | Fecha de inserción del registro |
| AUDIT\_UPDATE\_DTTM | NULLABLE | TIMESTAMP | Fecha de actualización del registro |

Tabla concilia\_tab\_operacion

| Nombre de campo | Modo | Tipo de campo | Descripción |
| --- | --- | --- | --- |
| ID\_Concilia\_tab\_operacion | REQUIRED | STRING | Código de la operación de Conciliación |
| Empresa | NULLABLE | STRING | Nombre o identificación de la empresa |
| Cod | NULLABLE | STRING | Código del registro |
| Des | NULLABLE | STRING | Descripción del registro |
| Debe\_haber | NULLABLE | STRING | Indica si es debe o haber |
| Centraliza | NULLABLE | BOOLEAN | Indica si el registro está centralizado |
| Atrib1 | NULLABLE | STRING | Atributo adicional 1 |
| Atrib2 | NULLABLE | FLOAT | Atributo adicional 2, valor numérico |
| Atrib3 | NULLABLE | BOOLEAN | Atributo adicional 3, indicador booleano |
| Cuenta | NULLABLE | STRING | Cuenta contable asociada |
| Ccosto | NULLABLE | STRING | Centro de costos asociado |
| Aux1 | NULLABLE | STRING | Campo auxiliar adicional |
| Tipdoc | NULLABLE | INTEGER | Tipo de documento |
| Tipo\_conciliacion | NULLABLE | STRING | Tipo de conciliación asociada |
| Pide\_referencia | NULLABLE | STRING | Indica si se debe solicitar una referencia |
| SSMA\_TimeStamp | NULLABLE | BYTES | Marca de tiempo para control de cambios |
| AUDIT\_INSERT\_DTTM | REQUIRED | TIMESTAMP | Fecha de inserción del registro |
| AUDIT\_UPDATE\_DTTM | NULLABLE | TIMESTAMP | Fecha de actualización del registro |

Tabla ctabancos

| Nombre de campo | Modo | Tipo de campo | Descripción |
| --- | --- | --- | --- |
| ID\_Ctabancos | REQUIRED | STRING | Código de la cuenta bancaria |
| Cod | NULLABLE | STRING | Código del banco o cuenta bancaria |
| Des | NULLABLE | STRING | Descripción del banco o cuenta |
| cuenta\_contable | NULLABLE | STRING | Número de la cuenta contable asociada |
| usuario | NULLABLE | STRING | Usuario responsable de la cuenta o registro |
| fechora | NULLABLE | DATETIME | Fecha y hora del registro |
| Rut\_Empresa | NULLABLE | STRING | RUT o identificación de la empresa asociada |
| AUDIT\_INSERT\_DTTM | REQUIRED | TIMESTAMP | Fecha de inserción del registro |
| AUDIT\_UPDATE\_DTTM | NULLABLE | TIMESTAMP | Fecha de actualización del registro |

Tabla documentos

| Nombre de campo | Modo | Tipo de campo | Descripción |
| --- | --- | --- | --- |
| ID\_Documentos | REQUIRED | STRING | Código del Documento |
| Empresa | NULLABLE | STRING | Nombre o identificación de la empresa |
| Cod | NULLABLE | STRING | Código del documento |
| Centraliza | NULLABLE | BOOLEAN | Indica si el documento está centralizado |
| Atrib1 | NULLABLE | STRING | Atributo 1 del documento |
| Atrib2 | NULLABLE | FLOAT | Atributo 2, valor numérico asociado al documento |
| Atrib3 | NULLABLE | BOOLEAN | Atributo 3, indicador |
| Atrib4 | NULLABLE | BOOLEAN | Atributo 4, indicador |
| Atrib5 | NULLABLE | BOOLEAN | Atributo 5, indicador |
| Atrib6 | NULLABLE | BOOLEAN | Atributo 6, indicador |
| Cuenta | NULLABLE | STRING | Cuenta contable asociada |
| Aux1 | NULLABLE | STRING | Campo auxiliar 1 |
| Origen | NULLABLE | STRING | Origen del documento |
| Clase | NULLABLE | INTEGER | Clase del documento |
| Tipo\_foliacion | NULLABLE | INTEGER | Tipo de foliación del documento |
| SSMA\_TimeStamp | NULLABLE | BYTES | Marca de tiempo |
| Clase\_doc1 | NULLABLE | STRING | Clase del documento 1 |
| Clase\_doc2 | NULLABLE | STRING | Clase del documento 2 |
| Clase\_doc3 | NULLABLE | STRING | Clase del documento 3 |
| Cod\_clase1 | NULLABLE | INTEGER | Código de la clase 1 |
| Cod\_clase2 | NULLABLE | INTEGER | Código de la clase 2 |
| Cod\_clase3 | NULLABLE | INTEGER | Código de la clase 3 |
| Extracontab1 | NULLABLE | FLOAT | Valor contable extra asociado al documento |
| AUDIT\_INSERT\_DTTM | REQUIRED | TIMESTAMP | Fecha de inserción del registro |
| AUDIT\_UPDATE\_DTTM | NULLABLE | TIMESTAMP | Fecha de actualización del registro |

Más detalle de la documentación de las tablas aquí : <https://hvrinformes.atlassian.net/wiki/spaces/F1PDT/pages/edit-v2/21889123>

## Creación de Cloud Functions para la carga de datos en BigQuery

Para la carga de datos se crearon 3 Cloud Function, cada una con una función distinta, las cuales se explicarán a continuación:

**Cloud Funtion limpieza\_csv\_dev**

**Descripción General**

Esta Cloud Function se activa automáticamente cuando un archivo CSV se carga en un Bucket específico de Google Cloud Storage (GCS). Su propósito principal es realizar una limpieza y validación de los datos en el archivo y guardar el archivo ya procesado en un Bucket final clean\_raw\_data\_hvr\_dev. Además, Almacena todas las filas inválidas detectadas que no cumpla con los requisitos de formato en un archivo separado alojado en el Bucket error\_raw\_data\_hvr\_dev.

**Configuración de la Función**

**Información General**

* Región: us-central1
* Memoria asignada: 2 GiB
* CPU: 1
* Tiempo de espera: 60 segundos
* Cantidad mínima de instancias: 0
* Cantidad máxima de instancias: 100
* Simultaneidad: 2
* Cuenta de servicio: 43780507605-compute@developer.gserviceaccount.com
* Actualización de política: Actualizaciones automáticas habilitadas

**Configuración de Red**

* Configuración de entrada: Tráfico interno
* Conector de VPC: No configurado

**Activador de Eventarc**

La función se activa mediante Eventarc, configurado para responder a eventos de carga de objetos en GCS.

* Nombre del activador: limpieza-csv-dev-083043
* Proveedor del evento: Cloud Storage
* Tipo de evento: google.cloud.storage.object.v1.finalized
* Recibir eventos desde: Bucket raw\_data\_hvr\_dev (región us-central1)
* Cuenta de servicio: 43780507605-compute@developer.gserviceaccount.com
* Reintento en caso de error: Habilitado

**Entorno de Ejecución**

* Entorno de ejecución: Python 3.10
* Punto de entrada: process\_csv
* Archivo principal: main.py
* Dependencias: Se encuentran en el archivo requirements.txt e incluyen:
  + functions-framework>=3.\*
  + gcsfs
  + google-cloud-functions

**Flujo General del Proceso**

1. Activación: La función se activa mediante un evento automático de subida de archivo en el bucket raw\_data\_hvr\_dev.
2. Copia de seguridad: El archivo original se guarda en un bucket de respaldo temp\_data\_hvr\_dev para poder ser recuperado en caso de errores inesperados.
3. Limpieza del archivo: Se eliminan caracteres o líneas innecesarias y se reordenan columnas para estandarizar los datos.
4. Validación: Cada línea se verifica para asegurar que cumpla con el número correcto de columnas tomando como referencia el encabezado, esto en el caso de que exista un elemento separador de columnas “;” adicional entre los datos.
5. Guardado: El archivo ya procesado se guarda en el bucket clean\_raw\_data\_hvr\_dev. Y en el caso de encontrar errores de formato, se genera un archivo de mismo nombre con la adición de la fecha en el bucket error\_raw\_data\_hvr\_devs.

**Configuración**

Variables y Entornos

* Project ID: 'hvr-dwh-dev-437001' (actualizar según el entorno).
* Trigger de activación asociado al bucket: clean\_raw\_data\_hvr\_dev
* Buckets de Google Cloud Storage:
  + temp\_data\_hvr\_dev: Bucket temporal para almacenar archivos intermedios.
  + clean\_raw\_data\_hvr\_dev: Bucket donde se guardan los archivos validados y limpios.
  + error\_raw\_data\_hvr\_dev: Bucket para almacenar los archivos con líneas que contienen errores.

**Detalles Técnicos**

Dependencias

La función utiliza las siguientes bibliotecas:

* gcsfs: Para leer y escribir archivos en los buckets de GCS.
* datetime: Para generar nombres de archivo únicos.
* time: Para implementar la espera exponencial en los intentos de escritura.

Más detalles de la documentación aquí: <https://hvrinformes.atlassian.net/wiki/spaces/F1PDT/pages/22020494/2.+Creaci+n+de+Cloud+Functions+para+la+carga+de+datos+en+BigQuery>

## Implementación de Dataform para gestión de transformaciones en BigQuery

El objetivo de implementar Dataform es automatizar las transformaciones de los datos en bigQuery, optimizando el flujo de trabajo del proceso de datos, estructurando el manejo de transformaciones en las capas Bronze, Silver y Gold, para mejorar la calidad y consistencia de los datos.

**Capas de almacenamiento:**

* Bronze: Capa de datos sin procesar.
* Silver: Capa de datos limpios, con transformaciones estándar.
* Gold: Capa de datos enriquecidos para análisis avanzado.

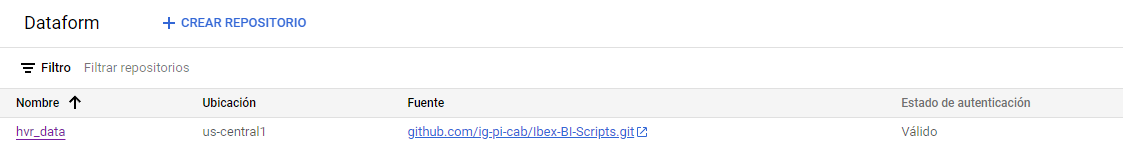
En Dataform se realizarán los siguiente:

* Se emplean scripts SQLX para gestionar las transformaciones en cada capa.
* Particiones: Se implementa la partición diaria en las tablas de la capa silver para optimizar el rendimiento en consultas.
* Monitoreo: Dataform genera reportes de ejecución y errores, facilitando la revisión de calidad y ajuste de procesos.

**Creación de repositorio**

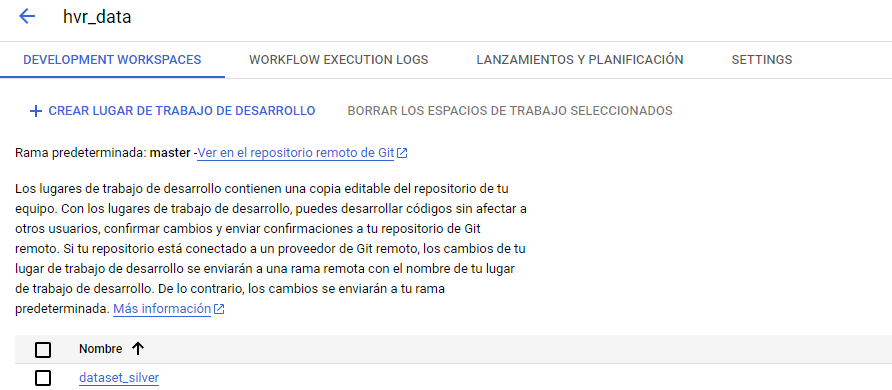
Se crea un repositorio dentro de Dataform

* Nombre: hvr\_data
* Ubicacion: us-central
* Fuente : [github.com/ig-pi-cab/Ibex-BI-Scripts.git](http://github.com/ig-pi-cab/Ibex-BI-Scripts.git)

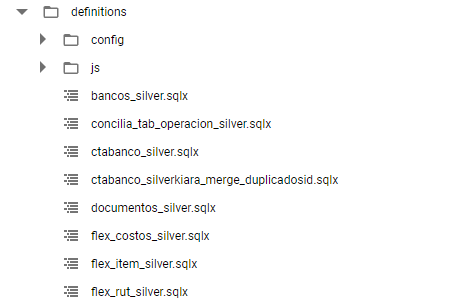


**Workspace**

Se crea el workspace dataset\_silver dentro del repositorio hvr\_data, donde se realizarán las transformaciones de los datos.



Creación de archivos en formato sqlx de las tablas silver



**Ejemplo de transformación de datos**

**Tabla ctabanco\_silver**

Se crea un archivo sqlx donde se crea una Query con la siguiente estructura.

Configuración de la Query

* type: "incremental" indica que esta tabla es incremental, es decir, solo se cargarán los datos nuevos o modificados desde la última ejecución.
* schema: "dataset\_silver" define el esquema o dataset en BigQuery donde se almacenará la tabla final.
* name: "ctabancos\_silver" es el nombre de la tabla final en la capa silver.
* bigquery: Configuración específica para BigQuery:
* partitionBy: "AUDIT\_INSERT\_DTTM" establece la partición de la tabla en la capa silver por la columna AUDIT\_INSERT\_DTTM, lo cual permite consultas más rápidas en función de fechas de inserción.
* updatePartitionFilter: "AUDIT\_INSERT\_DTTM >= TIMESTAMP\_SUB(CURRENT\_TIMESTAMP(), INTERVAL -31 DAY)" establece un filtro de actualización para cargar solo los datos de los últimos 31 días.
* uniqueKey: ["id\_Ctabancos"] define la clave única para la tabla, utilizada para identificar cada registro de forma exclusiva, lo cual es fundamental en operaciones incrementales.

config {

type: "incremental",

schema: "dataset\_silver",

name: "ctabancos\_silver",

bigquery: { partitionBy: "AUDIT\_INSERT\_DTTM",

updatePartitionFilter: "AUDIT\_INSERT\_DTTM >= TIMESTAMP\_SUB(CURRENT\_TIMESTAMP(), INTERVAL -31 DAY)" },

uniqueKey: ["id\_Ctabancos"]

}

**Selección de Campos**

Cada campo en el SELECT representa una columna que se extrae y transforma antes de insertarse en la capa silver :

SELECT

id\_Ctabancos, *--Selecciona el identificador único de la cuenta bancaria*

Cod, *-- Selecciona el código asociado al registro*

Des, *-- Selecciona la descripción de la cuenta bancaria*

cuenta\_contable, *-- Selecciona la cuenta contable asociada*

usuario, *-- Selecciona el usuario que realizó la operación*

SAFE\_CAST(fechora AS DATETIME) AS fechora, *-- Convierte 'fechora' a DATETIME, usando SAFE\_CAST para manejar valores inválidos*

Rut\_Empresa, *-- Selecciona el identificador de la empresa*

*CURRENT\_TIMESTAMP() AS AUDIT\_INSERT\_DTTM, -- Registra la fecha y hora de la operación de inserción CASE -- Establece 'AUDIT\_UPDATE\_DTTM' a la fecha actual si el registro fue modificado, considerando solo la fecha sin la hora*

WHEN AUDIT\_INSERT\_DTTM < CURRENT\_TIMESTAMP() THEN CURRENT\_TIMESTAMP() ELSE NULL END AS AUDIT\_UPDATE\_DTTM

Tabla de Origen y Filtro

FROM

`hvr-dwh-dev-437001.dataset\_bronze.ctabancos\_bronze`

WHERE

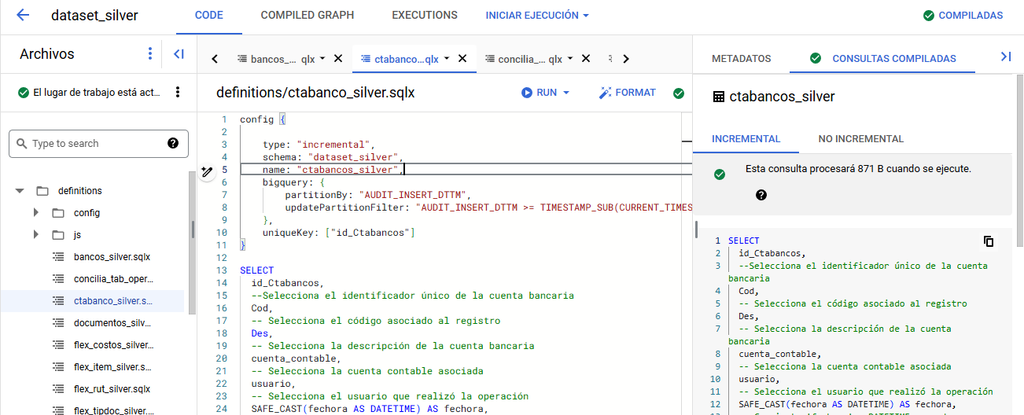
AUDIT\_INSERT\_DTTM >= TIMESTAMP\_ADD(CURRENT\_TIMESTAMP(), INTERVAL -31 DAY)

*-- Tabla de origen desde la cual se están leyendo los datos*

* FROM: Especifica la tabla en la capa bronze desde la cual se están leyendo los datos, en este caso dataset\_bronze.ctabancos\_bronze.
* WHERE: Filtra los datos para seleccionar solo aquellos registros que fueron insertados o actualizados en los últimos 31 días. Esto asegura que solo se procesen los registros recientes o relevantes en cada ejecución incremental.

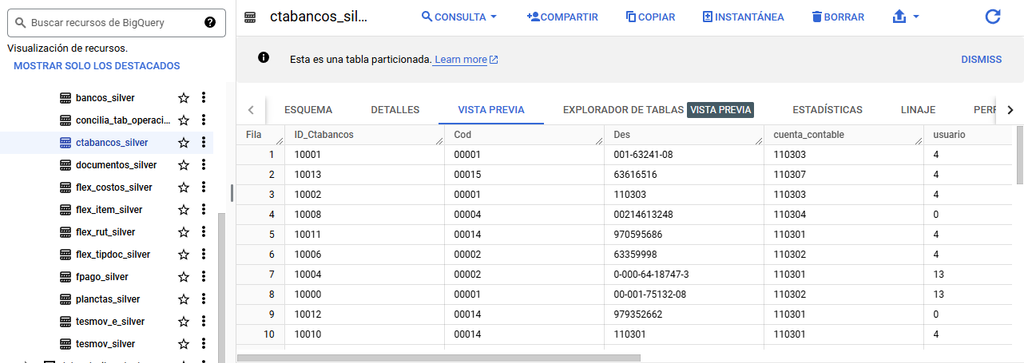
Esta query realiza una carga incremental de datos desde la capa bronze a la silver en BigQuery, aplicando transformaciones para convertir formatos de fechas, agregando auditoría con campos de inserción y actualización, y aplicando una partición por fechas de inserción para optimizar el rendimiento en consultas.

**Aprobación de datos transformados correctamente**

****

**Implementación de transformación de datos**

Los datos transformados se visualizan en la tabla

****

## 

## Configuración de git en Dataform para gestión de transformaciones en BigQuery

El objetivo es configurar GitHub en Dataform para facilitar la gestión y versionamiento de scripts de transformación de datos en BigQuery, permitiendo un desarrollo colaborativo y control de versiones.

Los beneficios que nos da es controlar cambios, nos permite revisiones de código, y asegura una sincronización efectiva en el equipo de desarrollo.

**Conexión con Github**

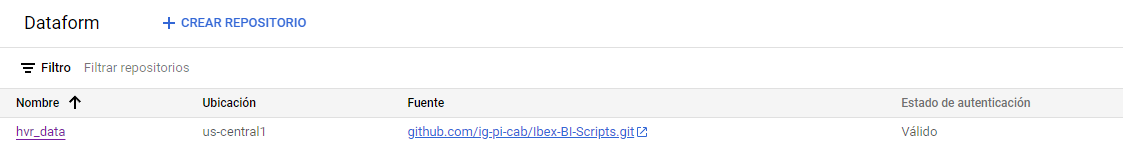
Se creó un repositorio en GitHub y se genera un token de acceso

* <https://github.com/ig-pi-cab/Ibex-BI-Scripts>

**Configuración de conexión en Dataform**

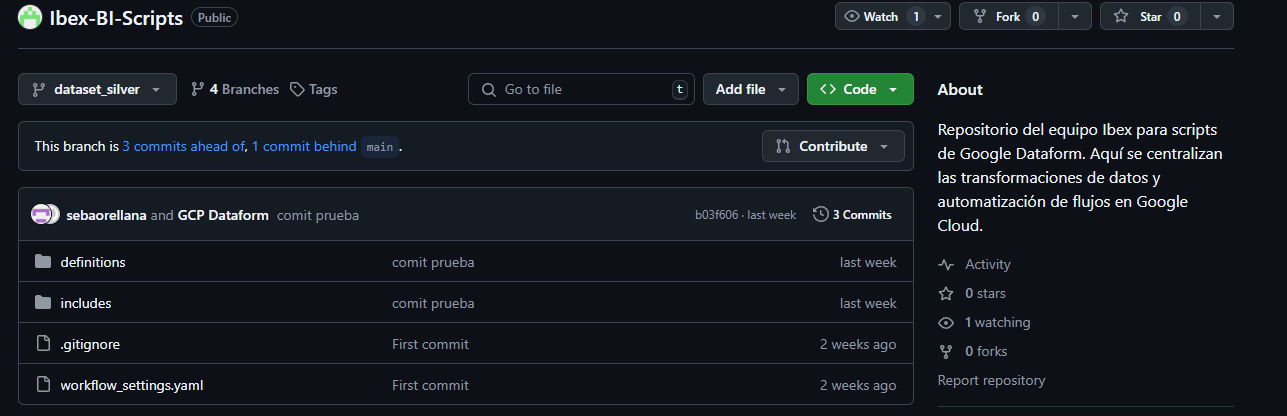
Se crea un repositorio dentro de Dataform y se da acceso con el token creado desde github

* Nombre: hvr\_data
* Ubicacion: us-central
* Fuente : [github.com/ig-pi-cab/Ibex-BI-Scripts.git](http://github.com/ig-pi-cab/Ibex-BI-Scripts.git)

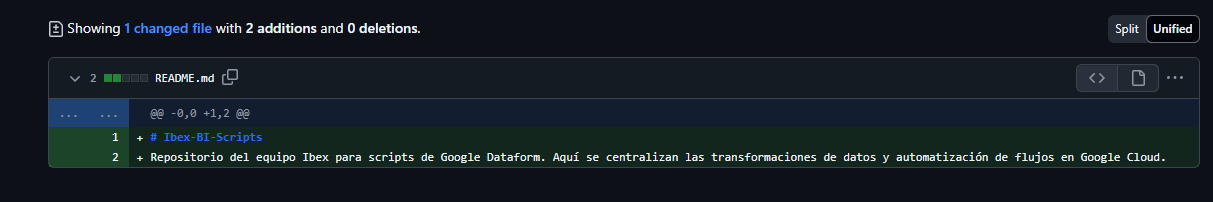


**Pruebas de commits**

Se realizan commits de prueba desde Dataform para compartir los scripts realizados



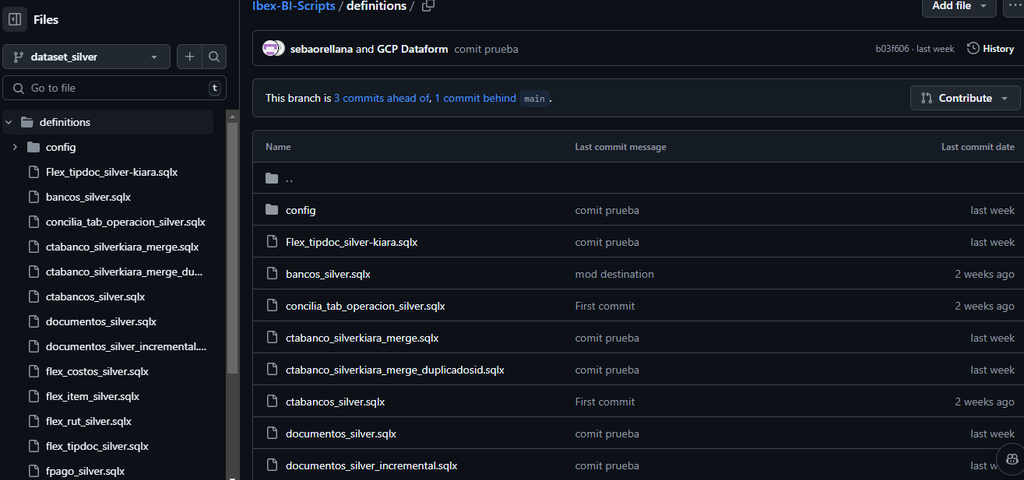
Creación de archivo README.md



Creación de Branch



**Registro de commits desde Branch dataset\_silver**

****

## Realizar pruebas de conexión desde el servidor local a GCP

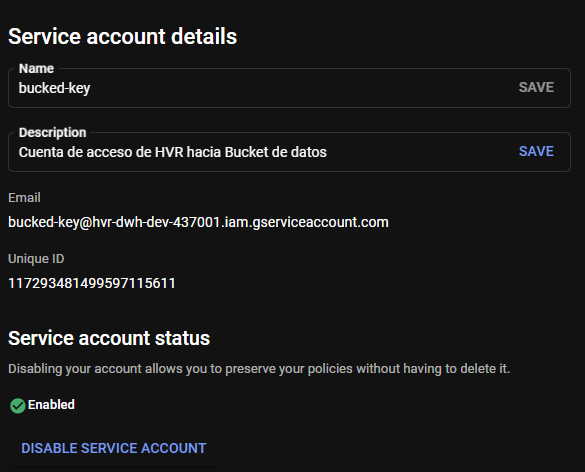
**Descripción general**

Este documento describe el proceso de configuración de una conexión segura y estable entre el servidor del cliente y el bucket de Google Cloud Platform (GCP) raw\_data\_hvr\_dev, con el objetivo de automatizar la carga diaria de archivos CSV. La tarea incluyó la creación de una cuenta de servicio en GCP con permisos específicos, la configuración de una key en el servidor del cliente para acceso seguro, y la verificación de la correcta transferencia de archivos.

El objetivo principal fue asegurar un flujo de datos confiable y seguro para la carga de documentos CSV desde el servidor hacia el bucket en GCP. Para esto, se llevaron a cabo varias actividades, incluyendo la creación y configuración de una cuenta de servicio, la generación de una key de acceso, y la ejecución de pruebas de carga automática de archivos.

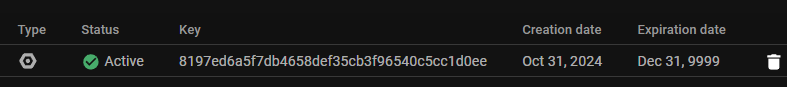
**Creación de la cuenta de servicio:**

Se crea una cuenta de servicio con permisos de administrador sobre el bucket raw\_data\_hvr\_dev.



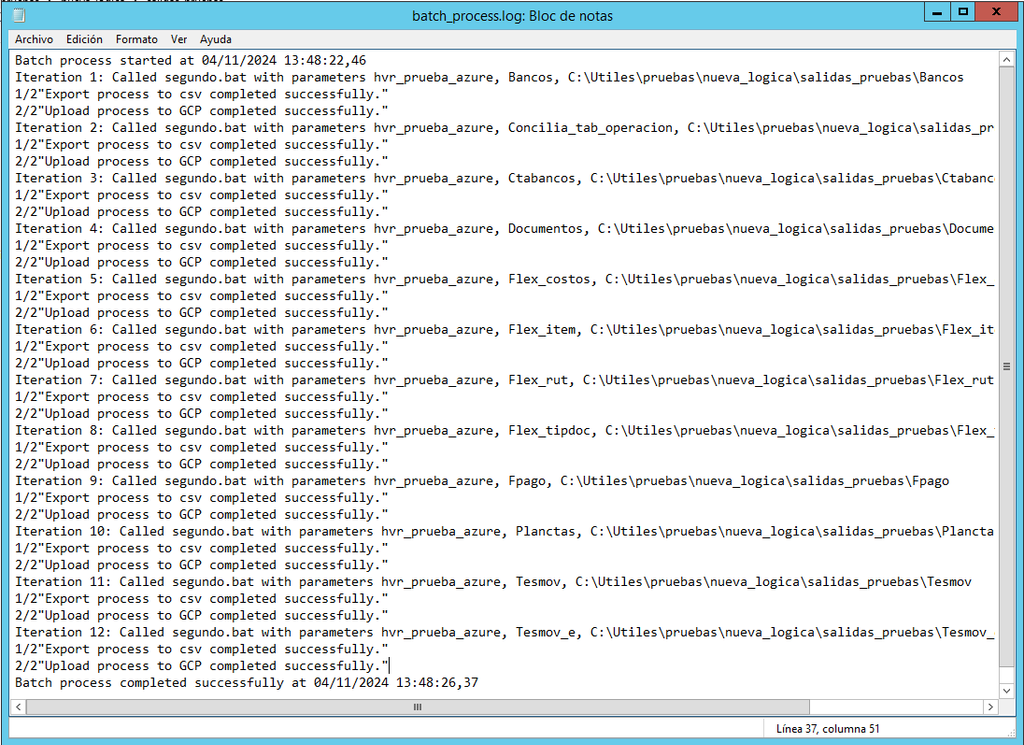
**Creación de Key de acceso**

Se crea una Key para conexión segura desde el servidor de cliente HVR hacia el bucket raw\_data\_hvr\_dev

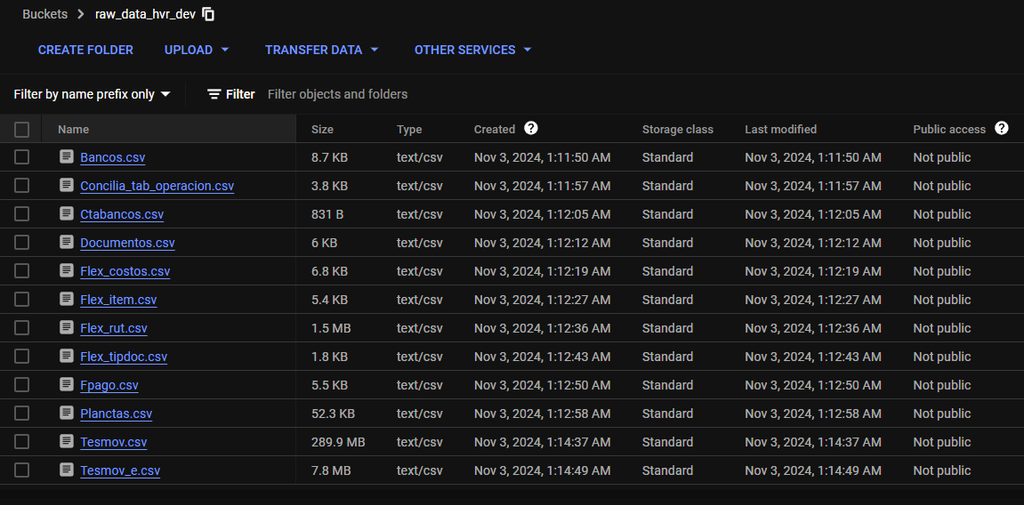


**Evidencia de log entregados por el cliente:**

Archivo log con la secuencia de carga de datos hacia el Bucket



Evidencia de documentos alojados en el bucket:



Documentación completa del proyecto aquí: <https://hvrinformes.atlassian.net/wiki/spaces/F1PDT/pages/29491201/Proyecto+Data+Warehouse+HVR>