

(27/08/2023 – 15/09/1023)

1.Integrantes:

Roman Aichino: Data Engineer

Max Jeffer: Data Engineer – Machine Learning

Diego Campos: Data Engineer

Eduardo Vivar: Data Analytics

2. Introducción:

Este es un proyecto que fue realizado desde la disciplina ‘ciencias de datos’, que busca recolectar datos para procesarlos, analizarlos y poder tomar decisiones en base a ellos. El proyecto cuenta con un EDA, una automatización de el ETL en la nube ‘Google Cloud Platform’, una carga incrementa a partir de los datos de una API, un dashboard con KPIs, y por último, un modelo de Machine Learning que devuelve cual es la mejor opción para una decisión empresarial.

3. Quiénes Somos y Quien es nuestro cliente:

Nosotros somos Drem Data Insights, una empresa dedicada a la ciencia de los datos, y en este proyecto en particular nuestra tarea es crear un sistema que proporcione información sobre los restaurantes en Estados Unidos, en el estado de Florida, con el objetivo de que nuestro cliente, una empresa que quiere abrir un restaurante, pueda tomar buenas decisiones en cuanto a la ubicación y otros detalles del negocio.

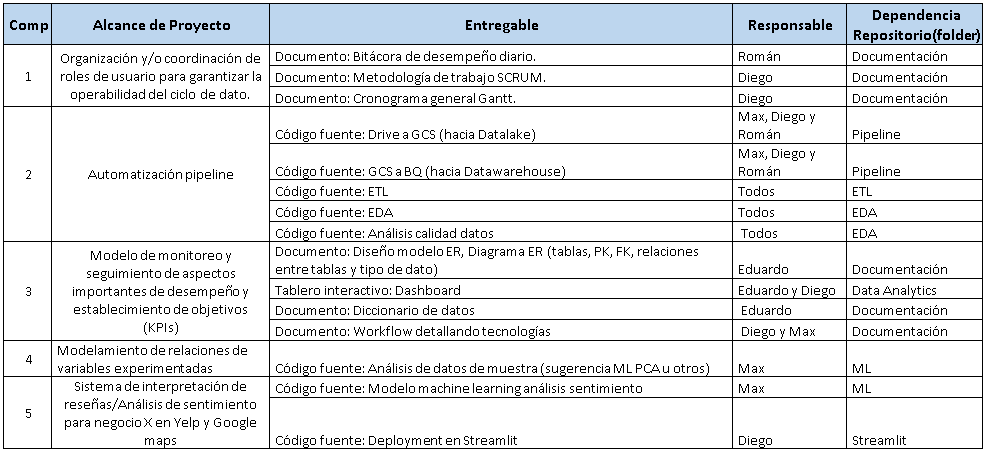
4. Objetivos:

Generales:

* Identificar ubicaciones y atributos estratégicos para abrir un restaurante mediante el análisis de datos a partir de la información recolectada, con el fin de maximizar la rentabilidad y la satisfacción del cliente.

Específicos:

* Definir metodología de gestión general (roles, tareas, diagrama de Gantt, etc).
* Definir el stack tecnológico.
* Automatización de pipelines (carga incremental, etl, envío de datasets al datawarehouse).
* Análisis exploratorio de los datos para la creación de KPIs.
* Diseño de un dashboard interactivo en Power Bi para realizar el seguimiento y monitoreo de los KPIs
* Análisis de sentimiento mediante un modelo de ML.

5. Alcance del Proyecto:

6. Planificación:

Metodología Scrum: Roles y Planificación

**Roles en Scrum**

* Product Owner:

Responsable de definir y priorizar el Product Backlog.

Asegura que el equipo trabaje en las tareas más valiosas para el cliente.

Colabora con interesados y el equipo para mantener la visión clara del producto.

* Scrum Master:

Facilitadora del proceso Scrum.

Garantiza que el equipo siga las prácticas y principios de Scrum.

Elimina obstáculos y fomenta un entorno de trabajo colaborativo.

* Equipo de Desarrollo:

Campos Diego

Aichino Roman

Jeffer Max

Vivar Eduardo

Transforman elementos del Product Backlog en incrementos de producto entregables.

Autónomos y autoorganizados, toman decisiones sobre cómo abordar tareas.

Colaboran para lograr los objetivos del Sprint.

**Planificación en Scrum**

Product Backlog: Lista de tareas priorizadas

1. 4 KPI’s
2. Documentar alcance del proyecto
3. EDA de los datos
4. Repositorio en Github
5. Implementación stack tecnológico
6. Metodología de trabajo
7. Diseño detallado
8. Equipo de trabajo - Roles y responsabilidades
9. Cronograma general - Gantt
10. Análisis preliminar de calidad de datos
11. ETL completo
12. Estructura de datos implementada (DW, DL, etc). Pueden usar algún servicio
13. Pipeline ETL automatizado
14. Diseño del Modelo ER
15. Pipelines para alimentar el DW
16. Datawarehouse
17. Automatización
18. Validación de datos
19. Documentación

Diagrama ER detallado (tablas, PK, FK y tipo de dato)

Diccionario de datos

Workflow detallando tecnologías

1. Análisis de datos de muestra
2. MVP/Proof of Concept de producto de ML ó MVP/Proof of Concept de Dashboard
3. Diseño de Reportes/Dashboards
4. KPIs
5. Modelos de ML
6. Modelo de ML en producción
7. Documentación

Selección del modelo, feature engineering

Informe de análisis

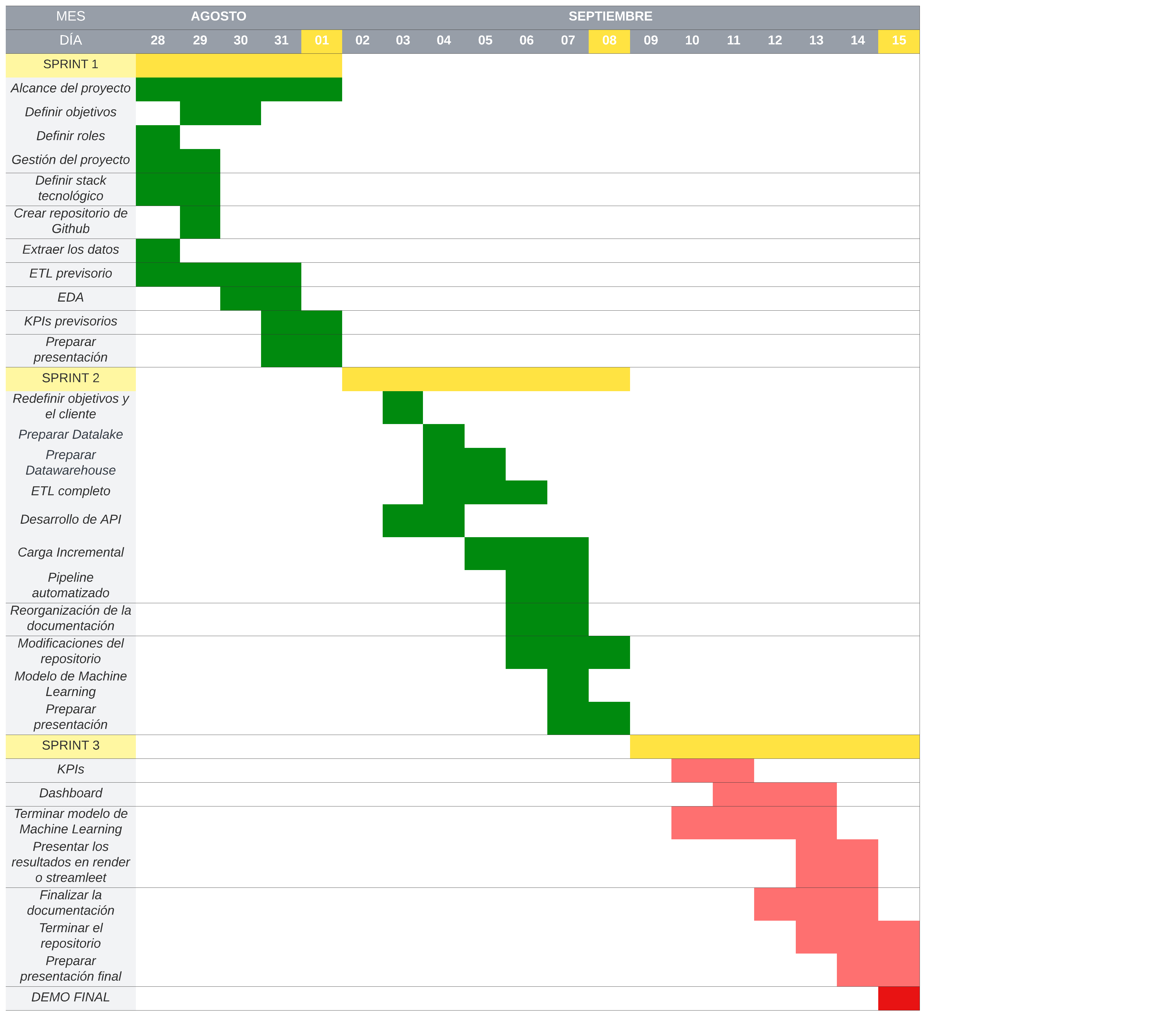
**Sprint Planning: Selección de tareas para el Sprint**

En la primera reunión, el equipo de desarrollo, junto con el Product Owner y el Scrum Master, seleccionó las siguientes tareas para el Sprint:

1. 4 KPI’s
2. Documentar alcance del proyecto
3. EDA de los datos
4. Repositorio en Github
5. Implementación stack tecnológico
6. Metodología de trabajo
7. Diseño detallado
8. Equipo de trabajo - Roles y responsabilidades
9. Cronograma general - Gantt
10. Análisis preliminar de calidad de datos

Estas tareas se han priorizado para este Sprint con el objetivo de lograr un incremento valioso del producto al final del período de desarrollo.

7. Diagrama de Gantt:



8. Stack Tecnológico:

**Parte técnica:**

Nube: Google Cloud Platform

Google Cloud Storage: para nuestro Datalake.

BigQuery: para nuestro Datawarehouse.

Google Colab: para la creación de archivos jupyter que automáticamente hagan la carga incremental, el ETL y la subida al Datawarehouse de todos nuestros datasets. Además, también lo utilizamos para los EDAs que quedarán subidos en el repositorio.

Cloud Functions: para la automatización de los archivos jupyter creados en Google Colab.

Pub/Sub y Cloud Scheduler: para el trigger de las funciones creadas en Cloud Functions.

Lenguaje de programación: Python, ya que es el que nuestra empresa utiliza habitualmente para procesar y analizar datos.

Librerías:

* Pandas: la mejor para procesar datasets en forma de dataframes.
* Mathplotlib: para la creación de gráficos que nos permitan mirar los datos desde otra perspectiva más analizable.
* Numpy: para ecuaciones matemáticas.
* Sk-learn: para nuestro modelo predictivo.

**Parte analítica:**

Power BI: para la creación del Dashboard interactivo.

Google Colab: para el análisis de los datos.

9. MODELO DE DATOS:

Se dispone como fuente de información los archivos de las dos plataformas, compuestos por archivos de tipo json, parquet, pkl.

Durante los procesos de ETL y EDA a dichos archivos, y tomando como referencia los diccionarios de las fuentes.

El modelo de datos que contiene el datawarehouse del proyecto contiene tablas de ambas plataformas.

**Tablas de la Plataforma Google Maps**:

**1. dim\_gm\_business**: Esta tabla almacena información detallada sobre los negocios en Google Maps, incluyendo detalles de ubicación, categorías, horarios de operación entre otros.

**2.** **fact\_gm\_review**: Aquí se almacenan las reseñas y calificaciones de los usuarios en relación con los negocios registrados en Google Maps, lo que nos da la posibilidad de analizar la interacción entre negocios y usuarios.

**Tablas de la Plataforma Yelp**:

**1.** **dim\_yelp\_business**: En esta tabla, mantenemos información esencial sobre los negocios en Yelp, como categorías, atributos y valoraciones.

**2. dim\_yelp\_user**: En esta tabla se encuentran registrados los usuarios de Yelp, a diferencia de googlemaps podemos tener mayor información sobre el perfil del usuario, como sus contactos y sus valoraciones.

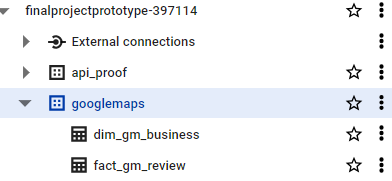
**3. fact\_yelp\_review**: Al igual que en el caso de Google Maps, esta tabla contiene revisiones y calificaciones de usuarios relacionadas con los negocios en Yelp, lo que nos permite evaluar la satisfacción del cliente y realizar análisis de sentimiento.

**4. fact\_yelp\_checkin**: Esta tabla contiene información sobre las visitas realizadas por los usuarios de Yelp en los negocios, lo que puede proporcionar indicios sobre la actividad y la popularidad de los lugares reflejada en la plataforma.

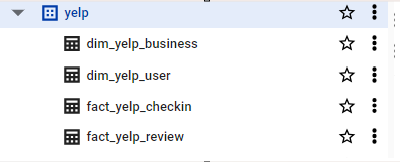
1. **DIAGRAMA DE ENTIDAD RELACION**

Las estructuras de datos para el datawarehouse del proyecto en la plataforma BIGQUERY, se han asignado a dos datasets:

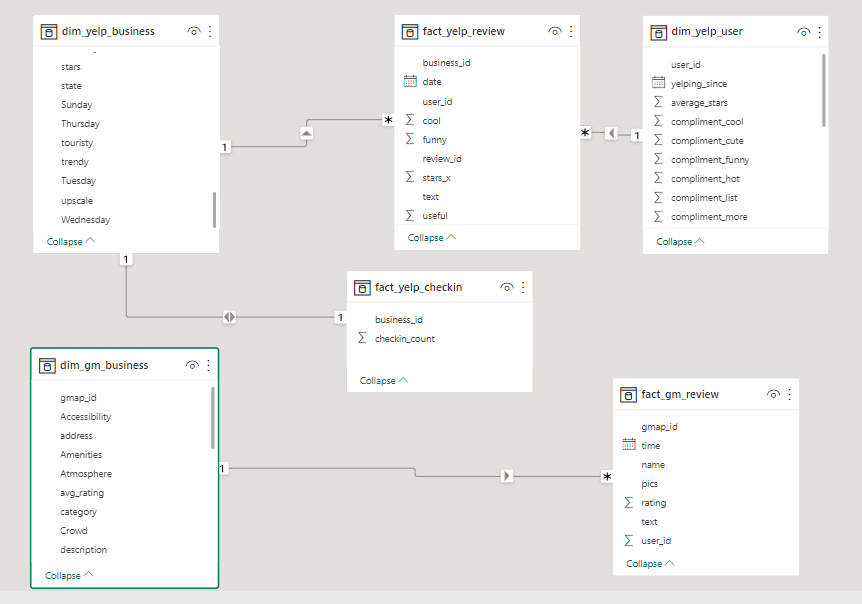
**googlemaps**



**yelp**



A continuación, tenemos el **DER – Diagrama Entidad Relación**

****

1. **DICCIONARIO DE DATOS**

Usamos la nomenclatura dim\_% para tablas de dimensiones, y fact\_% para tablas de hechos.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tabla: dim\_gm\_business** | | | |
| **columna** | **tipo dato** | **descripcion** | **ejemplo** |
| gmap\_id | STRING | Id Negocio google maps | 0x881614ce7c13acbb:0x5c7b18bbf6ec4f7e |
| name | STRING | Nombre | Walgreens Pharmacy |
| address | STRING | Dirección | Walgreens Pharmacy, 124 E North St, Kendallville, IN 46755 |
| description | STRING | Descripción | Department of the Walgreens chain providing prescription medications & other health-related items. |
| latitude | FLOAT | Latitud | 41.45186 |
| longitude | FLOAT | Longitud | -85.2666757 |
| category | STRING | categoría | ['Pharmacy'], |
| avg\_rating | FLOAT | valoración media | 4.2 |
| num\_of\_reviews | INTEGER | numero de valoraciones | 5 |
| price | STRING | nivel de precios | '$$' |
| state | STRING | estado | Closes soon ⋅ 1:30PM ⋅ Reopens 2PM |
| Service options | STRING | detalle de atributo Service options |  |
| Popular for | STRING | detalle de atributo Popular for |  |
| Accessibility | STRING | detalle de horario Accessibility |  |
| Offerings | STRING | detalle de horario Offerings |  |
| Dining options | STRING | detalle de atributo Dining options |  |
| Crowd | STRING | detalle de horario Crowd |  |
| Atmosphere | STRING | detalle de horario Atmosphere |  |
| Amenities | STRING | detalle de horario Amenities |  |
| Monday | STRING | detalle de horario Monday |  |
| Tuesday | STRING | detalle de horario Tuesday |  |
| Wednesday | STRING | detalle de horario Wednesday |  |
| Thursday | STRING | detalle de horario Thursday |  |
| Friday | STRING | detalle de horario Friday |  |
| Saturday | STRING | detalle de horario Saturday |  |
| Sunday | STRING | detalle de horario Sunday |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Tabla: dim\_gm\_review** | | | |
| **columna** | **tipo dato** | **descripción** | **ejemplo** |
| user\_id | FLOAT | Id usuario | id usuario |
| gmap\_id | STRING | Id Negocio | Id Negocio google maps |
| name | STRING | Nombre usuario | nombre usuario |
| time | TIMESTAMP | fecha-hora | 2021-08-03 15:07:30.740000 UTC |
| rating | INTEGER | valoración de la reseña | 5 |
| text | STRING | texto de la reseña | 'Cool place, great people, awesome dentist!', |
| pics | STRING | fotos | [ { 'url': ['https://lh5.googleusercontent.com/p/AF1QipNq2nZC5TH4\_M7h5xRAd61hoTgvY1o9lozABguI=w150-h150-k-no-p']}] |

|  |  |  |  |
| --- | --- | --- | --- |
| **dim\_yelp\_business** | | | |
| **columna** | **tipo dato** | **descripción** | **ejemplo** |
| business\_id | STRING | id del negocio, refiere al negocio en business. json | tnhfDv5Il8EaGSXZGiuQGg |
| name | STRING | nombre del negocio | Garaje |
| address | STRING | dirección completa del negocio | 475 3rd St |
| city | STRING | nombre de ciudad |  |
| state | STRING | código del estado | FL |
| postal\_code | STRING | código postal | 94107 |
| latitude | STRING | latitud | 37.78175295 |
| longitude | STRING | longitud | -122.39612197 |
| stars | STRING | rating en estrellas, redondeado a 0 o 0.5 | 4.5 |
| review\_count | INTEGER | numero de reseñas | 1198 |
| is\_open | STRING | abierto | 1 |
| Restaurants | STRING | flag restaurants | True |
| Food | STRING | flag food | True |
| Nightlife | STRING | flag Nightlife | False |
| American\_Traditional | STRING | flag American\_Traditional | False |
| Bars | STRING | flag Bars | True |
| Sandwiches | STRING | flag Sandwiches | False |
| Breakfast & Brunch | STRING | flag Breakfast & Brunch | False |
| Fast Food | STRING | flag Fast Food | True |
| Pizza | STRING | flag Pizza | False |
| American\_New | STRING | flag American\_New | False |
| RestaurantsReservations | STRING | flag RestaurantsReservations | True |
| OutdoorSeating | STRING | flag OutdoorSeating | False |
| Alcohol | STRING | flag Alcohol | False |
| RestaurantsPriceRange2 | STRING | flag RestaurantsPriceRange2 | True |
| BusinessAcceptsCreditCards | STRING | flag BusinessAcceptsCreditCards | False |
| GoodForKids | STRING | flag GoodForKids | False |
| RestaurantsDelivery | STRING | flag RestaurantsDelivery | True |
| RestaurantsGoodForGroups | STRING | flag RestaurantsGoodForGroups | False |
| touristy | STRING | flag touristy | False |
| hipster | STRING | flag hipster | True |
| romantic | STRING | flag romantic | False |
| divey | STRING | flag divey | False |
| intimate | STRING | flag intimate | True |
| trendy | STRING | flag trendy | False |
| upscale | STRING | flag upscale | False |
| classy | STRING | flag classy | True |
| casual | STRING | flag casual | False |
| Monday | STRING | flag Monday | 09AM-11PM |
| Tuesday | STRING | flag Tuesday | 09AM-11PM |
| Wednesday | STRING | flag Wednesday | 09AM-11PM |
| Thursday | STRING | flag Thursday | 09AM-11PM |
| Friday | STRING | flag Friday | 09AM-11PM |
| Saturday | STRING | flag Saturday | 09AM-11PM |
| Sunday | STRING | flag Sunday | 09AM-11PM |
|  |  |  |  |
| **dim\_yelp\_user** | | | |
| **columna** | **tipo dato** | **descripción** | **ejemplo** |
| user\_id | STRING | id de usuario que refiere al usuario en user.json | Ha3iJu77CxlrFm-vQRs\_8g |
| name | STRING | nombre del usuario | Sebastien |
| review\_count | INTEGER | numero de reseñas escritas | 56 |
| yelping\_since | TIMESTAMP | fecha de creación del usuario en Yelp en formato YYYY-MM-DD | "2011-01-01" |
| useful | INTEGER | número de votos marcados como útiles por el usuario | 21 |
| funny | INTEGER | número de votos marcados como graciosos por el usuario | 88 |
| cool | INTEGER | número de votos marcados como cool por el usuario | 15 |
| fans | INTEGER | número de fans que tiene el usuario | 1032 |
| average\_stars | FLOAT | promedio del valor de las reseñas | 4.31 |
| compliment\_hot | INTEGER | total de cumplidos 'hot' recibidos por el usuario | 339 |
| compliment\_more | INTEGER | total de cumplidos varios recibidos por el usuario | 668 |
| compliment\_profile | INTEGER | total de cumplidos por el perfil recibidos por el usuario | 42 |
| compliment\_cute | INTEGER | total de cumplidos 'cute' recibidos por el usuario | 62 |
| compliment\_list | INTEGER | total de listas de cumplidos recibidos por el usuario | 37 |
| compliment\_note | INTEGER | total de cumplidos como notas recibidos por el usuario | 356 |
| compliment\_plain | INTEGER | total de cumplidos planos recibidos por el usuario | 68 |
| compliment\_cool | INTEGER | total de cumplidos 'cool' recibidos por el usuario | 91 |
| compliment\_funny | INTEGER | total de cumplidos graciosos recibidos por el usuario | 99 |
| compliment\_writer | INTEGER | número de cumplidos escritos recibidos por el usuario | 95 |
| compliment\_photos | INTEGER | número de cumplidos en foto recibidos por el usuario | 50 |
| friends\_count | INTEGER | cantidad de amigos | 123 |
| elite\_count | INTEGER | cantidad de anos en elite | 2 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Tabla: dim\_yelp\_checkin** | | | |
| **columna** | **tipo dato** | **descripción** | **ejemplo** |
| business\_id | STRING | Id Negocio | tnhfDv5Il8EaGSXZGiuQGg |
| checkin\_count | INTEGER | conteo de checkins | 13003 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Tabla: dim\_yelp\_review** | | | |
| **columna** | **tipo dato** | **descripción** | **ejemplo** |
| review\_id | STRING | id de reseña | zdSx\_SD6obEhz9VrW9uAWA |
| user\_id | STRING | id único de usuario, refiere al usuario en user.json | Ha3iJu77CxlrFm-vQRs\_8g |
| business\_id | STRING | id del negocio, refiere al negocio en business.json | tnhfDv5Il8EaGSXZGiuQGg |
| date | TIMESTAMP | fecha formato YYYY-MM-DD | "2016-03-09" |
| stars\_x | INTEGER | puntaje en estrellas de 1 al 5 | 4 |
| useful | INTEGER | números de votos como reseña útil | 0 |
| funny | INTEGER | número de votos como reseña graciosa | 0 |
| cool | INTEGER | número de votos como reseña cool. | 0 |
| text | STRING | la reseña en inglés | Great place to hang out after work: the prices are decent, and the ambience is fun..... |

10. Análisis de Datos:

Descripción de las fuentes de datos utilizadas.

Proceso de limpieza, transformación y exploración de datos.

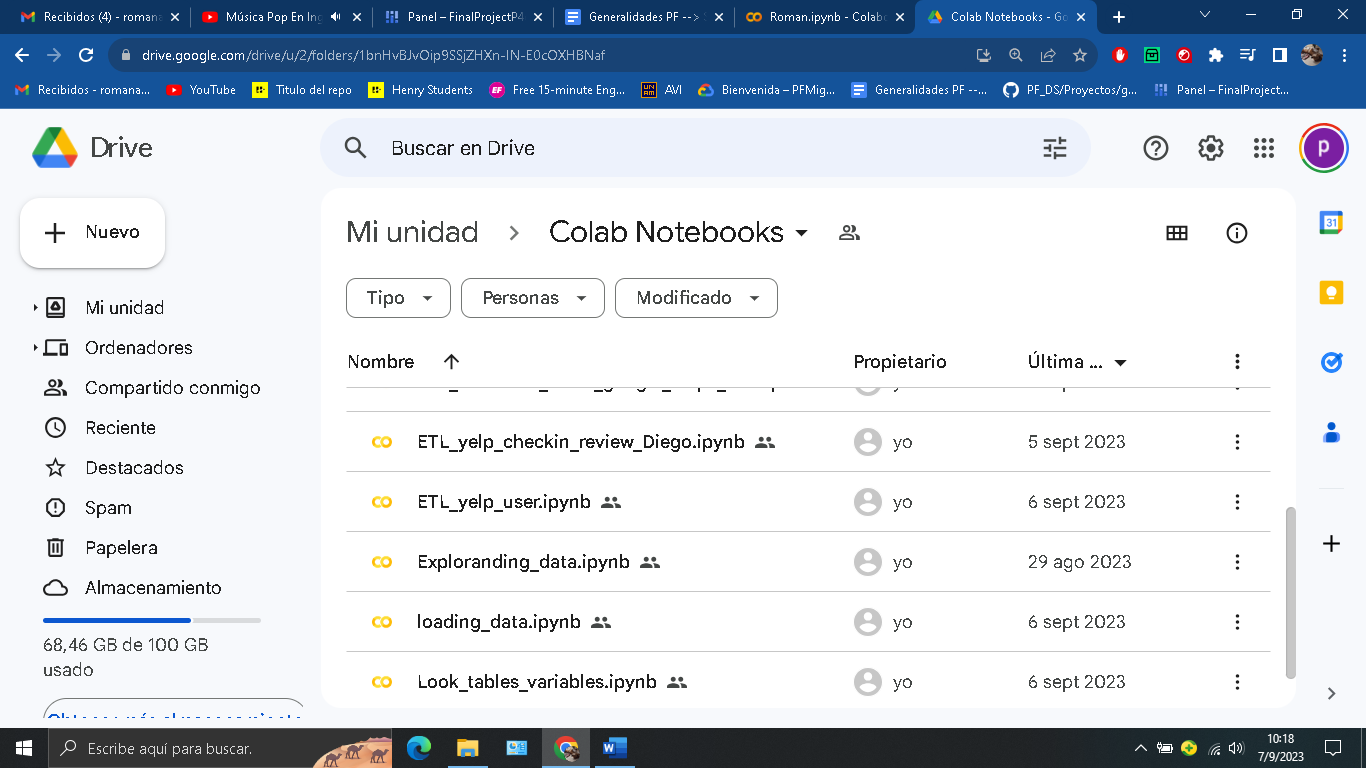
Resumen de los hallazgos clave del análisis de datos.

11. Ingeniería de Datos:

Ya habiendo nombrado anteriormente nuestro stack tecnológico, lo que queda es explicar cómo usamos cada herramienta, y el ciclo de vida del dato.

Nuestros datos principales son extraídos desde un Google Drive que nos aportó el cliente, a partir de ellos lo primero que hicimos fue realizarles el ETL con Google Colab.

Imagen que demuestra la creación de varios colabs en una carpeta e Google Drive:



Luego de realizar tanto los ETLs como los EDAs en Google Colab, subimos los datasets a Google Cloud Storage para poder implementar los ETLs que realizamos en Google Colab, pero de una manera más automatizada a través de Google Cloud Functions.

Imagen que demuestra los datos en varios buckets de GCS:

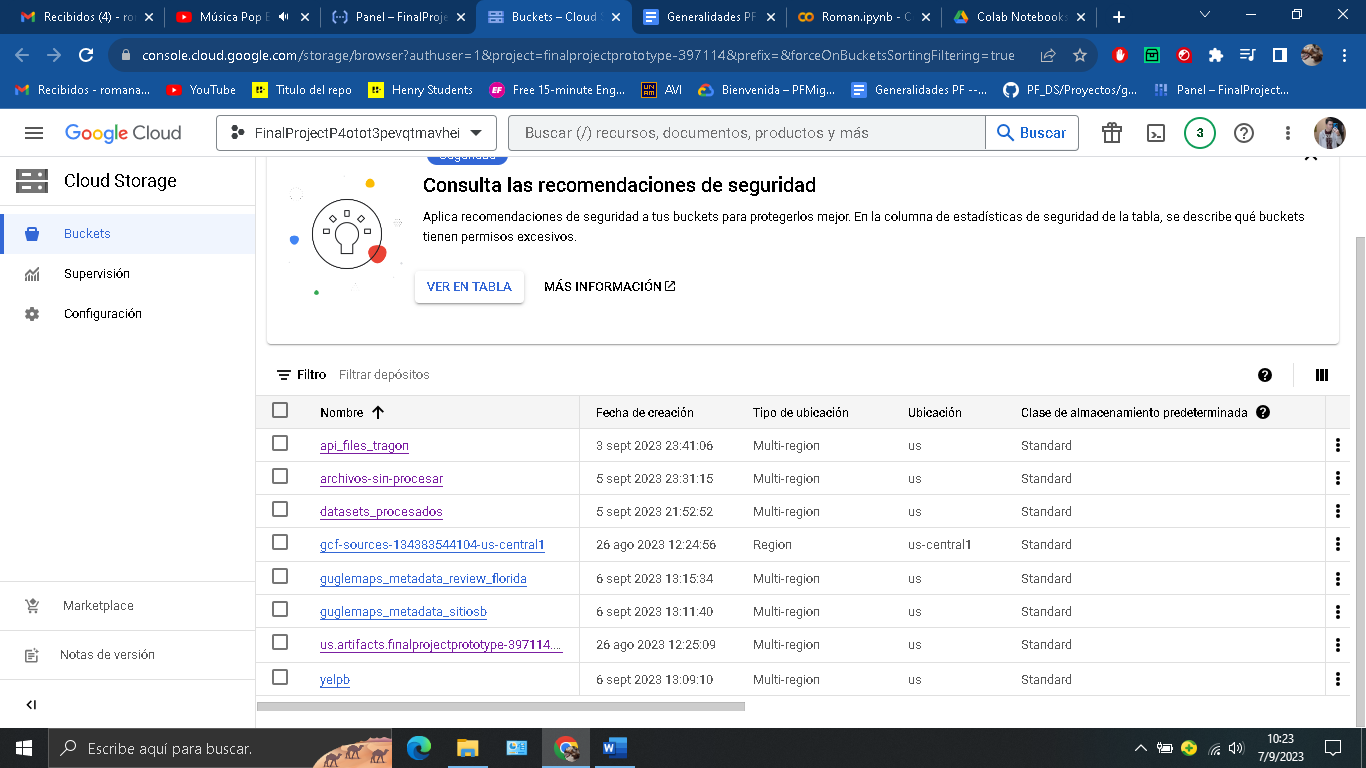
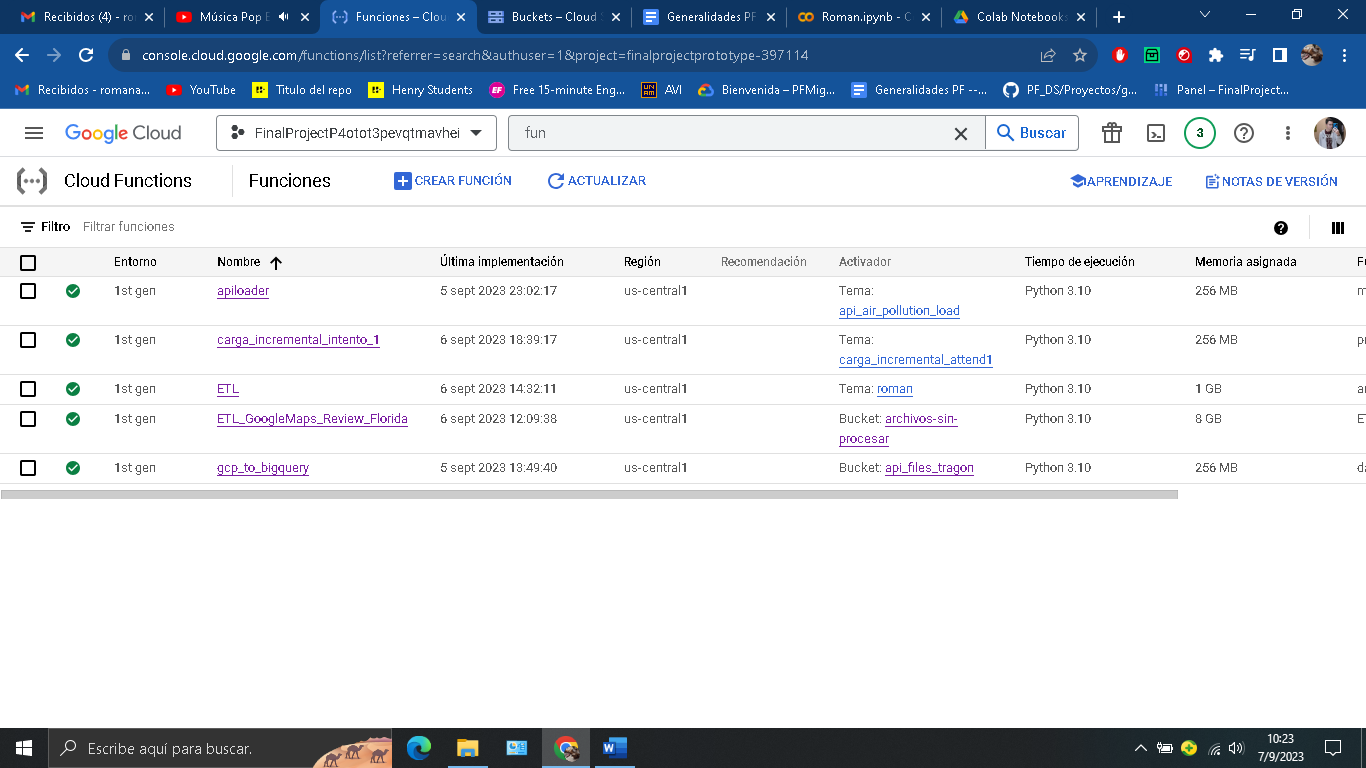


Imagen que demuestra la creación de funciones para los ETLs, la API y la carga incremental:

e

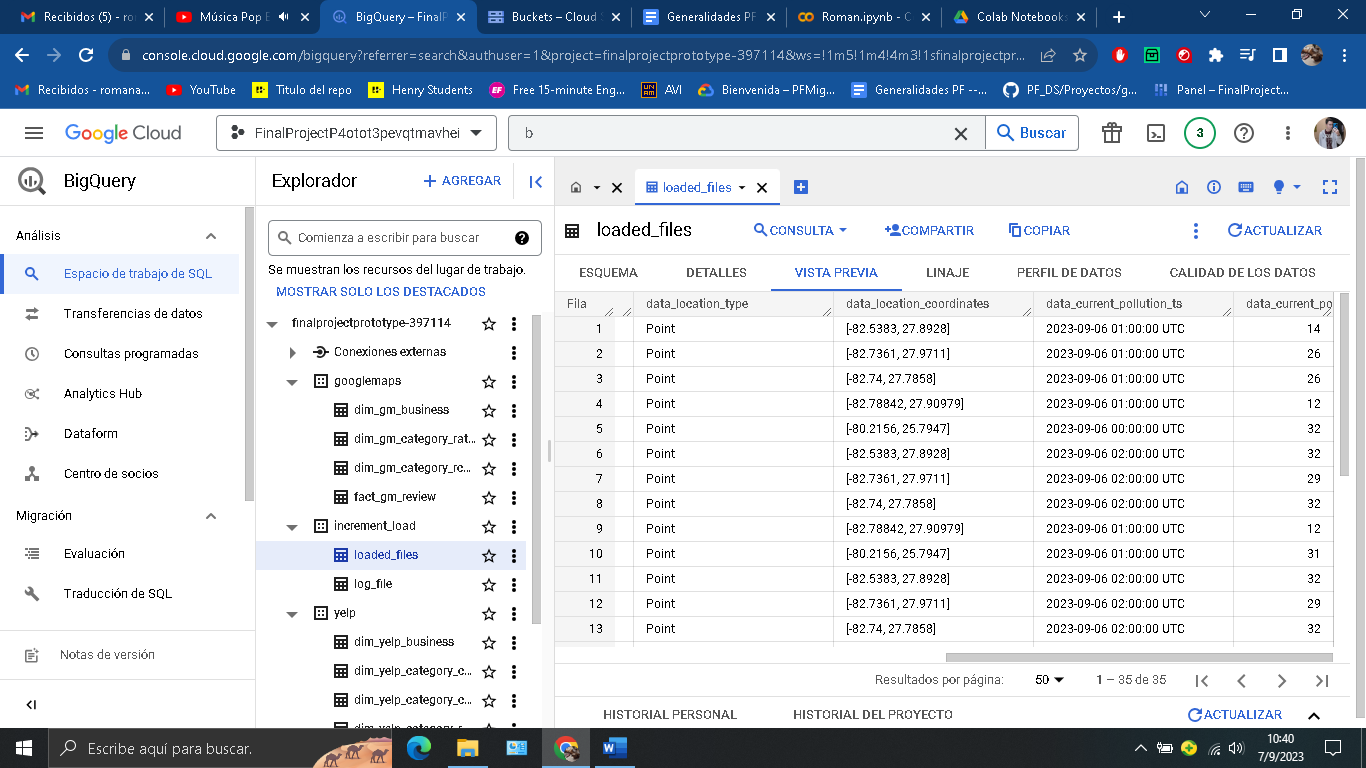
Explicación de lo que hace cada función de la imagen:

* Apiloader: Esta función es la encargada de llamar a la API, extraer los datos y almacenarlos en GSC. Además, se ejecuta cada una hora ya que la API tiene ese tiempo de actualización.
* carga\_incremental\_intento\_1: Esta función cumple varios roles, primero en principal es la encargada de llevar un registro en cuanto a los archivos que ya fueron cargados en BigQuery (nuestro datawarehouse) con el objetivo de que, si la API llegase a fallar, no se almacenen datos repetidos; y, en segundo lugar, la función también es la encargada de llevar los datos a una tabla de BigQuery en donde se van almacenando debajo de los datos ya existentes.
* ETL: Aquí están los ETLs, que como previamente mencioné, fueron creados en Google Colab de forma provisoria. Los objetivos de este pipeline, son el de juntar todos los datasets del mismo tipo para crear uno que pueda ser procesado y enviado a tablas de BigQuery.

La diferencia entre esta función y la de carga incremental, es que los datos de Google Drive jamás serán actualizados, por lo que esta función quedaría automatizada y lista para recibir actualizaciones, pero no ocurrirá a menos que sea manualmente mediante un dataset creado por nosotros.

(Tanto ETL\_GoogleMaps\_Review\_Florida como gco\_to\_bigquery son funciones que implementamos en un principio pero que luego descartamos ante la creación de otras mejores)

Imagen que demuestra la presencia de las tablas con los datos en BigQuery:



Como vemos a la izquierda, tenemos 3 conjuntos de datos llamados ‘googlemaps’, ‘inremental\_load’ y ‘yelp’, y cada uno tiene dentro las tablas que le corresponden.

12. Modelo de Machine Learning:

13. KPIs (Indicadores Clave de Desempeño):

14. Dashboard:

15. Conclusiones:

16. Referencias:

Página donde se encuentra nuestra API: <https://www.iqair.com>

17. Anexos: