# 

# Proyecto Consultoría en análisis de datos para expansión de marca

# 

# 

# 

# Cliente

# The Flavor Company

# 

# 

# índice

[**Blue Consulting Group 1**](#_kvoegyas914a)

[**INTRODUCCIÓN 2**](#_zg8oqs1owv8a)

[**PLANTEAMIENTO DEL PROBLEMA 2**](#_xn8li0r6afey)

[OBJETIVOS DE ANÁLISIS E IMPLEMENTACIÓN 2](#_wv71h04ff3zp)

[KPI’s 3](#_h4k5yj1wece0)

[ALCANCE 3](#_kt808zcqhuxb)

[FUNDAMENTO 3](#_2oazdq5i9qzi)

[**METODOLOGÍA DE TRABAJO 3**](#_l3qdt61pn4z2)

[SPRINTS 3](#_bsiv3u1b22k0)

[Sprint #1: puesta en marcha del proyecto y trabajo con datos 3](#_8n245cgzt51y)

[Sprint #2: data engineering 4](#_xq35aj2k4v6x)

[Sprint #3: data analytics + ML 4](#_6k2h8vktn21f)

[CRONOGRAMA 4](#_v3s9rmhwpy5f)

[EQUIPO DE TRABAJO 5](#_mwsxrafbhums)

[**PROPUESTA SOLUCIÓN 6**](#_xj2swctr043g)

# 

# 

# INTRODUCCIÓN

## PLANTEAMIENTO DEL PROBLEMA

Melina, dueña de la cadena de restaurantes "The Flavor Factory" con presencia a nivel nacional en los Estados Unidos, ha confiado en nosotros para identificar oportunidades estratégicas de expansión "The Flavor Factory". Nuestra estrategia implica el análisis exhaustivo de datos de reseñas de Google Earth y Yelp de los últimos 5 años, lo que permitirá una integración fluida con su plan de marketing. Además, complementaremos esta información con datos sociodemográficos para obtener una comprensión más profunda del mercado objetivo. El resultado será un análisis objetivo y basado en datos que respalde las decisiones de expansión de manera efectiva y respalde el crecimiento de su negocio.

Nuestra misión es proporcionar un enfoque data-driven sólido y una visión clara del panorama competitivo. A través de la combinación de datos de reseñas y datos sociodemográficos, esperamos ofrecer recomendaciones estratégicas precisas que permitan a su cadena de restaurantes tomar decisiones informadas y maximizar su éxito en esta fase de expansión.

Para este problema nos vamos a centrar principalmente en las ciudades y condados principales de Estados Unidos con mayor actividad económica y **como escala temporal de análisis se tomará desde el año 2018**, siendo un plazo adecuado para tener una idea de los cambios en el mercado y tener un diagnóstico de la industria de los restaurantes.

Es importante tener en cuenta que los años de pandemia cambiaron radicalmente el escenario según The Washington Post cerca de 72.200, además que durante y en la post pandemia el aumento de la digitalización aumentó significativamente ([ver fuente](https://www.mordorintelligence.com/es/industry-reports/us-food-service-restaurant-market)), pero ello se analizará los últimos años antes de la pandemia 2018 y 2019 como punto de referencia pero por lo mencionado anteriormente se le dará mayor peso a los datos que se tienen en los años posteriores a la pandemia desde la mitad del 2021.

## OBJETIVOS DEL PROYECTO

## Objetivos Generales

* Hace un diagnóstico de la industria de los restaurantes en los Estados Unidos en base a la información de reseñas de Yelp y Google Maps e información secundaria obtenida.
* Realizar un estudio de los datos disponibles con el fin de identificar patrones, tendencias o lugares óptimos para un restaurante.
* Entregar una herramienta dinámica e interactiva para ayudar en la decisión y seguimiento en un plan de expansión de marca de restaurantes.

## Objetivos Específicos

* Llevar a cabo un análisis comparativo entre las reseñas de Yelp y las de Google Maps.
* Identificar las variables sociodemográficas que influyen en el éxito de un restaurante y proponer recomendaciones con las acciones a tomar para la apertura de nuevos locales de restaurantes u hoteles en ubicaciones específicas.
* Establecer por medio de datos, la ubicación y categoría donde puede ser más viable aperturar un restaurante.
* Establecer herramientas y flujos de trabajo para procesar la gran cantidad de datos brindados de manera efectiva.
* Generar 2 tablas madre de datos que contengan toda la información relevante para nuestro estudio de los establecimientos y usuarios de interés para nosotros y puedan ser consultados por medio de el uso de Big Query.

## KPI’s

1. **Aumentar la calificación de 3 a 5 para el próximo año (2023)**

Para ello como métricas definimos:

* Calificación Promedio:
* Número reseñas , que porcentaje representa con el total de reseñas de un restaurante. % de las reseñas buenas y % de las reseñas malas.
* Ver si es posible medir Tasa de Respuesta a Reseñas, es decir si a lo largo del tiempo se dio solución a las reseñas negativas o a los tips de usuarios.

1. **Aperturar dos nuevos restaurantes para el próximo año (2023), donde se identifique oportunidades de expansión demográfica.**

Para ello como métricas definimos:

* **Indice de Potencial de Mercado:** Evalúa densidad de población e ingreso mediano o promedio para identificar ubicaciones con concentración de clientes potenciales con la capacidad de gasto adecuada.

1. **Saturación geográfica de competidores:** cuantifica la densidad de competidores en una zona geográfica, calculada como la cantidad de competidores por kilómetro cuadrado

## ALCANCE

Este trabajo tiene por objeto dar una propuesta solución a la problemática planteada haciendo uso de los beneficios de computación en la nube, haciendo uso de google cloud, cumpliendo con los hitos establecidos en cada sprint de la metodología ágil, descritos en l a sección titulada “Metodología de Trabajo”

# METODOLOGÍA DE TRABAJO

Como marco de trabajo se sigue la metodología ágil en específico el scrum. Se ha establecido un plazo de 45 días (mes y medio) dividido en 3 ciclos de desarrollo cortos denominados sprints. Cada sprint tiene definido objetivos e hitos entregables en cada uno de ellos. Se detallan a continuación:

## SPRINTS

### Sprint 1: puesta en marcha del proyecto y trabajo con datos

- Definición del problema, objetivos, alcance y fundamentación del proyecto

- Establecimiento de los KPI’s

- ETL de los datos.

- EDA de los datos

- Repositorio en Github.

- Implementación stack tecnológico

- Metodología de trabajo

- Diseño detallado

- Equipo de trabajo - Roles y responsabilidades

- Cronograma general - Gantt

- Análisis preliminar de calidad de datos

### Sprint 2: data engineering

- ETL completo

- EDA completo

- Estructura de datos implementada (DW, DL, etc).

- Pipeline ETL automatizado

- Diseño del Modelo ER

- Pipelines para alimentar el DW

- Data Warehouse

- Automatización

- Validación de datos

- Documentación

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

- Diccionario de datos

- Workflow detallando tecnologías

- Análisis de datos de muestra

- MVP/ Proof of Concept de producto de ML ó MVP/ Proof of Concept de Dashboard

### Sprint 3: data analytics + ML

- Diseño de Reportes/Dashboards

- KPIs

- Modelos de ML

- Modelo de ML en producción

- Documentación

## CRONOGRAMA

[BCG\_Proyect Timeline](https://docs.google.com/spreadsheets/d/1gwdYvOJ0-7As4o2srmWfDXI75RlLRKSk_1beTSTgZ7E/edit?usp=sharing)

## EQUIPO DE TRABAJO

| **Nombre** | **Rol** |
| --- | --- |
| Juan Diego Angeles | Data Engineer |
| Juan M Buitrago | Business Analyst |
| Melodi Romero | Data Analyst |
| Kevin Davison | Project Manager |

## STACK TECNOLÓGICO

Hasta el presente sprint se ha utilizado un conjunto de herramientas ofrecidos por Google Cloud, lenguajes de programación con sus bibliotecas, motores de procesamiento y herramientas de colaboración y coordinación con el objetivo de afrontar los retos que presentaba el trabajo en cuanto a volumen y variedad de datos, necesidad a satisfacer y coordinación de los involucrados.



# TRABAJANDO CON LOS DATOS DEL PROYECTO

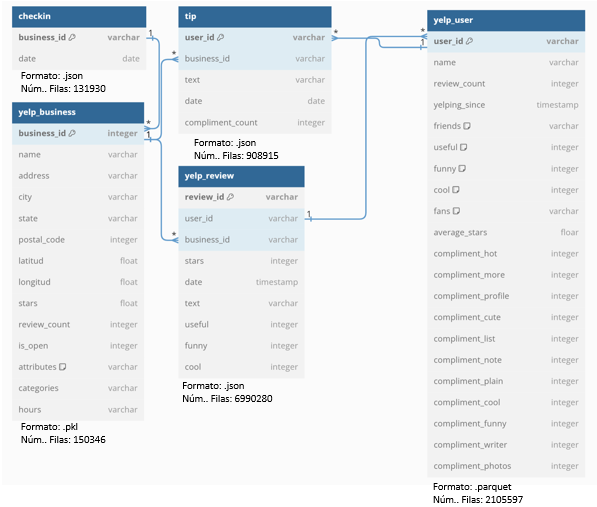
En la etapa de transformación de los datos se abordarán los siguientes aspectos básicos para realizar el tratamiento previo y limpieza de datos para poder llevar a cabo la carga de datos al data wharehouse

* Verificación De tipo de dato de columna
* Dimensionalidad de los datos
* Valores nulos y en creo en caso de las columnas numéricas
* Verificación gráfica de outliers
* Indagación de consistencia de los datos, máximos, mínimos.

## Datos Google reviews y Yelp

Inicialmente se tuvieron diferentes datos provenientes de dos plataformas de reseñas de negocios, google reviews y Yelp, estos datos vienen en diferentes formatos, y archivos. En las imágenes a continuación veremos los esquemas de datos y los formatos de las tablas.

### Esquema datos Yelp



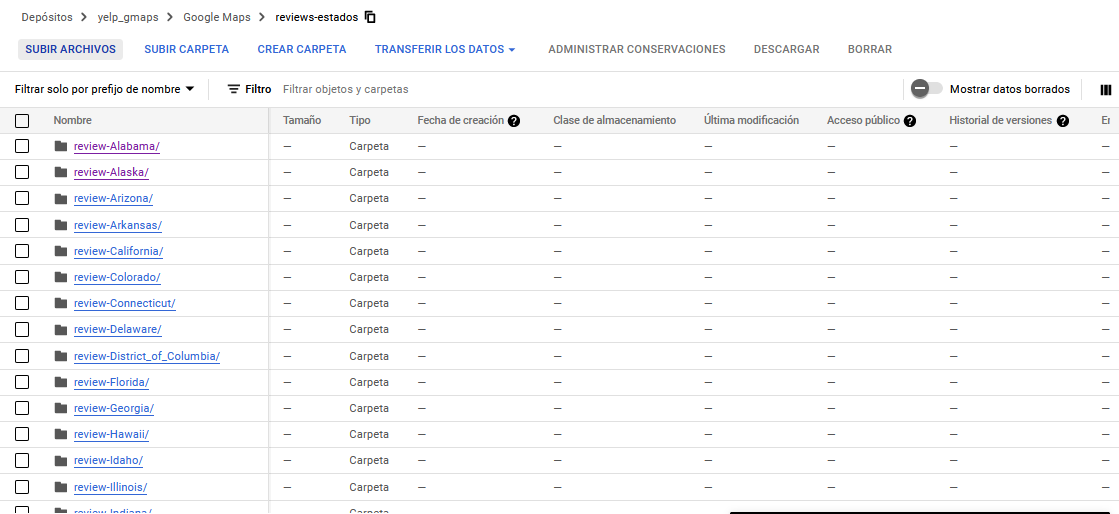
### Esquema de datos Google reviews

****

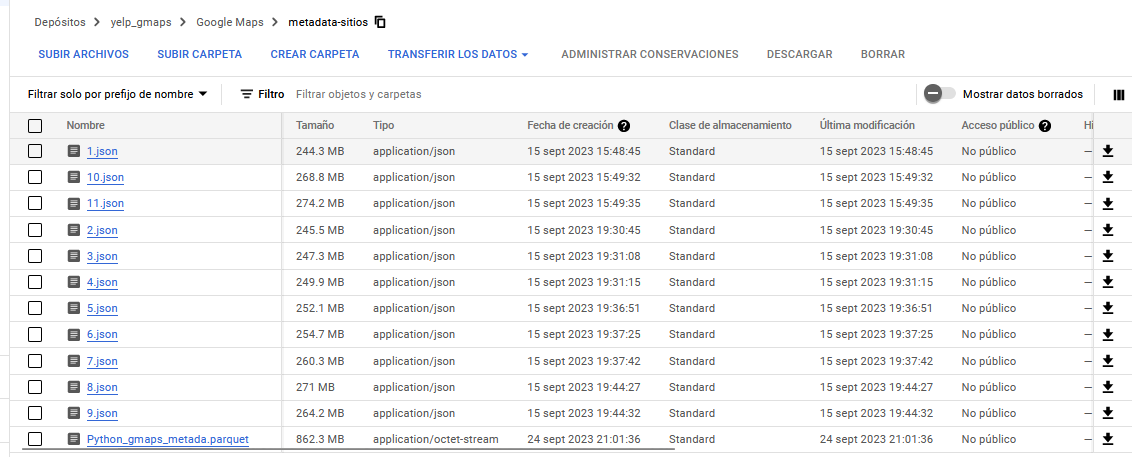
Los datos que se ven en los esquemas fueron almacenados en nuestro datalake de Google Cloud Storage, ya que es una opción escalable y permite almacenar cualquier tipo de datos, y acceder al mismo desde cualquier ordenador con acceso a la cuenta de google.

## Google Reviews

Como se mencionó anteriormente los datos tanto de las reseñas, datos google\_reviews\_estados y metadata\_sitios venían atomizados en json individuales, como se puede ver en las imágen abajo:



Para las reseñas teníamos 611 archivos formato .json en promedio cada uno con 150.000 datos repartidos en carpetas de cada uno de los estados.



En metadata sitios se tenías 11 json con 275001 registros cada uno.

Para solucionar ello se generó un script de Pyspark llamado el cual unifica todos los json y cuyo código se encuentra en el repositorio de github en este [enlace](https://github.com/kdavisong/Google-Yelp/blob/main/Sprint-2/Scripts_gs_josn_unificador.py).

Al ejecutar el script unificador en la carpeta contenedora tanto de las reseñas pore estado como en la carpeta que contiene los 11 json de metadatos, quedamos con dos archivos .parquet

### Google Reviews: Reseñas de usuarios por estados

Empleando la capacidad de procesamiento de Spark obtenemos la siguiente información de las reseñas de google maps a nivel estatal, previo a realizar la transformación:

| Número de filas = 89’946.359 | Número de Columnas = 9 |
| --- | --- |

#### Schema de el dataset con porcentaje de nulos:

| |-- user\_id: double (nullable = true) : Porcentaje de Nulos = **0.00%**  |-- name: string (nullable = true) : Porcentaje de Nulos = **0.00%**  |-- time: long (nullable = true): Porcentaje de Nulos = **0.00%**  |-- rating: long (nullable = true) : Porcentaje de Nulos = **0.00%**  |-- text: string (nullable = true) : Porcentaje de Nulos = **43.70%**  |-- pics: array (nullable = true) : Porcentaje de Nulos = **97.22%**  | |-- element: struct (containsNull = true)  | | |-- url: array (nullable = true)  | | | |-- element: string (containsNull = true)  |-- resp: struct (nullable = true) : Porcentaje de Nulos = **87.74%**  | |-- text: string (nullable = true)  | |-- time: long (nullable = true)  |-- gmap\_id: string (nullable = true) : Porcentaje de Nulos = **0.00%** |
| --- |

En el esquema se observan algunos tipos de datos de columnas que deben ser modificados, adicionalmente podemos ver ya algunos datos interesantes. De los 89 millones de reseñas registradas menos del 3% se les incluye foto, asimismo menos del 13% de las reseñas tienen respuesta por parte del propietario del negocio.

Esta tabla tiene el registro de la fecha, hora, minuto y segundo en el que fue puesta la reseña, por lo que se va a filtrar las reseñas puestas desde el año 2018 hasta el presente año, se crea la columna año.

Asimismo ya que no hay datos representativos la columna “pics” y “resp” se eliminaran del dataframe. Previo a realizar el inner join de nuestros dos conjuntos de datos tendremos estas dimensiones de datos:

| Número de filas = 74’281.753 | Número de Columnas = 8 |
| --- | --- |

### Google Reviews: metadata sitios

Este conjunto de datos era algo más manejable de mucha menor dimensión sin embargo también se empleó PySpark, Spark y Pyarrow para su procesamiento y visualización de datos. Importante que aquí se listan todos los sitios disponibles en google reviews, así que se filtrará únicamente los establecimientos que son relevantes para el problema.

| Número de filas = 3’025.011 | Número de Columnas = 15 |
| --- | --- |

#### Schema de el dataset con porcentaje de nulos:

|-- name: string (nullable = true) : Porcentaje de Nulos = **0.0012%**

|-- address: string (nullable = true) : Porcentaje de Nulos = **2.66%**

|-- gmap\_id: string (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- description: string (nullable = true) : Porcentaje de Nulos = **91.59%**

|-- latitude: double (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- longitude: double (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- category: array (nullable = true) : Porcentaje de Nulos = **0.575%**

|-- avg\_rating: double (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- num\_of\_reviews: long (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- price: string (nullable = true) : Porcentaje de Nulos = **90.90%**

|-- hours: array (nullable = true) : Porcentaje de Nulos = **26.02%**

|-- MISC: struct (nullable = true) : Porcentaje de Nulos = **22.84%**

|-- state: string (nullable = true) : Porcentaje de Nulos = **24.68%**

|-- relative\_results: array (nullable = true) : Porcentaje de Nulos = **9.75%**

|-- url: string (nullable = true) : Porcentaje de Nulos = **0.00%**

#### La columna description tiene menos del 8% de datos que no son nulos, antes de filtrar por categorías de restaurantes, por lo que es muy poca información y no sería relevante por lo que se toma la decisión de eliminar este dataset.

#### Selección de categorías relevantes expansión de marca de restaurantes

Para esta selección se desanidó la columna “category” para analizar una a una, por una parte se exportaron las categorías y el número de repeticiones de estas en un archivo csv ([ver archivo](https://github.com/kdavisong/Google-Yelp/blob/main/Sprint-2/ETL_EDA/Google_Maps/Output-tables_categorias_part-00000-15dd8325-fd24-4c24-b8bb-1b74deb347e9-c000(1).csv)).

Se tenían un total de 4471 categorías reducidas a 44 relacionadas con la industria de los restaurantes, ya que si tomabamos únicamente los que tuviesen “Restaurant” en categoría íbamos a omitir muchos registros importantes de reposterías, cafés, comida étnica y heladerías que son importantes a considerar.

**CATEOGRÍAS SELECCIONADAS DE RESTAURANTES**

| 0 | Pizza restaurant | 12 | Pizza |
| --- | --- | --- | --- |
| 1 | British restaurant | 13 | Burrito restaurant |
| 2 | Asian restaurant | 14 | Cafe |
| 3 | Cake shop | 15 | bakery |
| 4 | Singaporean Restaurant | 16 | Spanish restaurant |
| 5 | Hamburger restaurant | 17 | Takeout restaurant |
| 6 | Grill | 18 | Family Restaurant |
| 7 | Health food restaurant | 19 | Donut shop |
| 8 | Mexican restaurant | 20 | Bistro |
| 9 | seafood | 21 | Hamburger Restaurant |
| 10 | Fast food restaurant | 22 | Sushi restaurant |
| 11 | Delivery restaurant | 23 | Vegetarian RestaurantJapanese restaurant |
| 24 | Tex-mex restaurant | 36 | American restaurant |
| 25 | Bar & grill | 37 | Bar |
| 26 | Restaurant | 38 | Steak house |
| 27 | Sandwich shop | 39 | Southern restaurant (US) |
| 28 | Indian restaurant | 40 | Chinese restaurant |
| 29 | Chicken restaurant | 41 | Diner |
| 30 | Italian restaurant | 42 | Brunch restaurant |
| 31 | ice cream shop | 43 | Juice shop |
| 32 | Dessert Restaurant | 44 | Breakfast restaurant |
| 33 | Coffee shop | 45 | Greek restaurant |
| 34 | New American restaurant |  | |
| 35 | Thai restaurant |

### Google Reviews: tabla unida output

Una vez realizado el proceso de transformación de nuestros archivos .parquet de reseñas y de metadata\_sitios se procede a realizar un inner join, con el objetivo de relacionar reseñas con información de establecimientos para dejar los datos listos para subir al data warehouse. La tabla resultante de Google Reviews que datos de usuarios relacionados con el establecimiento de comida al ser en formato parquet queda ligera y fácil de realizar las consultas propuestas.

| Número de filas = 3’481.123 | Número de Columnas = 22 |
| --- | --- |

## Yelp

Para Yelp los datos estaban divididos en muchas más tablas, sin embargo el número de datos era mucho menor, sin embargo la calidad de los datos iniciales es menor a la de Google Reviews. Para revisar el detalle de como se trabajaron los archivos de Yelp, revisar este [enlace.](https://github.com/kdavisong/Google-Yelp/tree/main/Sprint-2/ETL_EDA/Yelp)

### Reviews

Esta tabla contiene las primary key de las otras dos tablas principales como lo son la tabla de Bussines que contiene los establecimientos y la de User que son los datos de la cuenta de el usuario como el número de amigos, número de reseñas y los cumplidos que ha dejado en los negocios.

Se observa que los años donde más hay reseñas es en 2018 y 2019 con tendencia a descender. Como se expresó en el planteamiento del proyecto se tomarán las reseñas posteriores al año 2017.

| Número de filas = 6’990.280 | Número de Columnas = 9 |
| --- | --- |

### Business

Al comparar las dimensiones de el dataset con el dataset que contenía la información de los establecimientos comerciales de Google, tenemos a lo largo de los estados unidos únicamente 150346 negocios que tienen reseñas de Yelp, esto previo a filtrar únicamente los que son restaurantes.

| Número de filas = 150.346 | Número de Columnas = 28 |
| --- | --- |

El dataset inicial contenía todas sus columnas repetidas por lo que se eliminaron quedando solo 14 columnas. de la columna attributes se eliminaron los valores nulos.El porcentaje de nulos de las columnas correctas quedó así:  
  
Schema de el dataset con porcentaje de nulos:

|-- busuness\_id: string (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- name: string (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- address: string (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- city: string (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- state: double (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- postal\_code: double (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- latitude: array (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- longitude: double (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- stars: long (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- review\_count: string (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- is\_open: array (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- attributes: struct (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- categories: string (nullable = true) : Porcentaje de Nulos = **0.00%**

|-- hours: array (nullable = true) : Porcentaje de Nulos = **13.9%**

## Datos de la oficina del censo de los Estados Unidos

A través de la API del Bureau de Censo de Estados Unidos, se recopilaron datos de encuestas poblacionales desde el año 2010 hasta el 2023 con el objetivo de obtener información sobre las áreas de las ciudades y los ingresos promedio por ciudad. Estos datos se agruparon por condado, lo que dio como resultado una lista que abarcaba todos los registros de rangos de ingresos. Posteriormente, se calcularon los promedios de ingresos por condado.

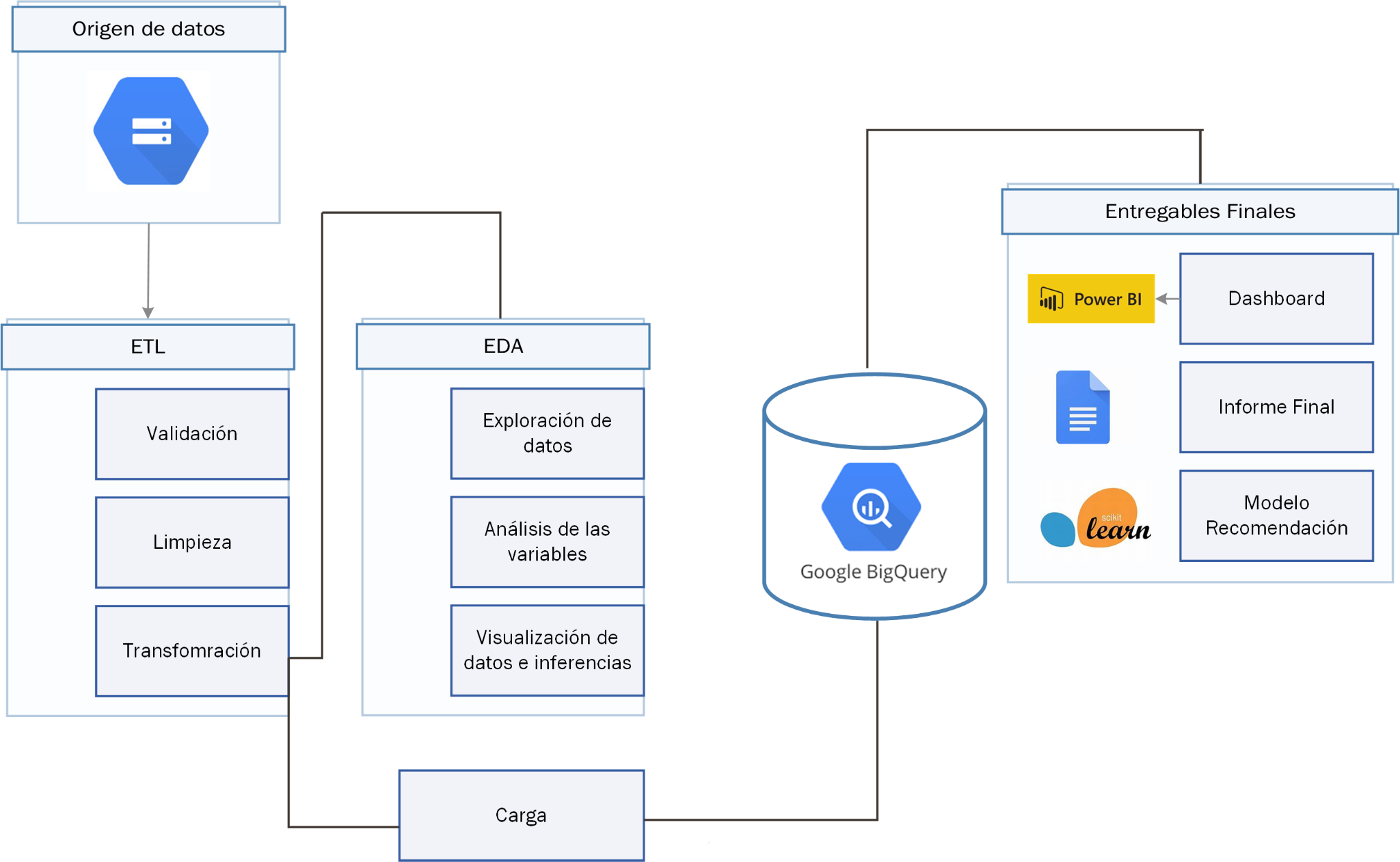
Además, se accedió a la API para facilitar la asignación de identificadores únicos a los registros y así poder mapear la información con cadenas de texto que identificaran los datos de manera más comprensible. Este proceso ayudó a mejorar la claridad y utilidad de la información recopilada, permitiendo un análisis más efectivo de las tendencias y diferencias en los ingresos a nivel de condado en los Estados Unidos.

Para mas detalle referirse al archivo trabajado de e[xtracción de datos de Census API](https://github.com/kdavisong/Google-Yelp/blob/main/Datasets/CensusAPI/EDA_D_ext.ipynb) y la transformación y análisis de estos datos.

| Número de filas = 1.712.307 | Número de Columnas = 8 |
| --- | --- |

## Flujo de Trabajo

El flujo de trabajo definido va a estar todo en Google Cloud y usará los servicios que ofrece Google Cloud, como Google Cloud storage para nuestro Data Lake, clusters para poder trabajar y procesar gran cantidad de datos comodamente y BigQuery para almacenar los datos transformados, acceder a ellos y realizar la conexión directamente a Power BI.



## Data Warehouse

Como almacén de datos se emplea BigQuery de Google Cloud Services, proporciona escalabilidad, facilidad de manejo de grandes volúmenes de datos e interoperabilidad ya que BigQuery ofrece, además BigQuery es apto para manejar datos estructurados y relacionales, como tablas SQL tradicionales, también es muy eficaz para manejar datos no relacionales y semi-estructurados.

Para el caso del presente modelo se tendrá el modelo relacional con 3 tablas de hechos, y se aprovechará la eficiencia del formato columnar de parquet con la facilidad de acceso, consultas y posibilidad de integración del uso de SQL con BigQuery.