

UNIVERSIDAD GALILEO DE GUATEMALA  
CUARTO TRIMESTRE  
PRODUCT DEVELOPMENT  
ING. CARLOS ZELADA  
AUX. OBED ESPINOZA

# MANUAL TECNICO

JHERSON SAZO  
21000341

## INDICE

### Tabla de contenido

|                                      |   |
|--------------------------------------|---|
| MANUAL TECNICO.....                  | 1 |
| 1. Aspectos generales.....           | 3 |
| 2. Tecnologías.....                  | 3 |
| a. Docker: .....                     | 3 |
| b. Streamlit: .....                  | 3 |
| c. Mysql; .....                      | 3 |
| d. Airflow: .....                    | 3 |
| 3. Diagramas.....                    | 4 |
| a. Diagrama ETL.....                 | 4 |
| b. Proceso ETL .....                 | 5 |
| c. Diagrama Relacional.....          | 5 |
| 4. Deployment de la aplicación ..... | 6 |
| a. Estructura del proyecto:.....     | 6 |
| b. Ejecución .....                   | 7 |

## 1. Aspectos generales

En este manual se describen las tecnologías y flujos utilizados para llevar a cabo una aplicación en la que se pueda consultar de forma dinámica los casos de COVID, proporcionados a través de archivos csvs.

## 2. Tecnologías

En esta sección se describen las tecnologías utilizadas en la aplicación con una pequeña descripción de cada una.

### a. Docker:

Docker es una plataforma de software de código abierto para crear, implementar y administrar contenedores de aplicaciones virtualizados en un sistema operativo (SO) común, con un ecosistema de herramientas aliadas

### b. Streamlit:

Streamlit es una biblioteca de Python de código abierto que facilita la creación y el intercambio de hermosas aplicaciones web personalizadas para el aprendizaje automático y la ciencia de datos.

### c. Mysql;

Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo,<sup>12</sup> y una de las más populares en general junto a Oracle y Microsoft SQL Server, todo para entornos de desarrollo web.

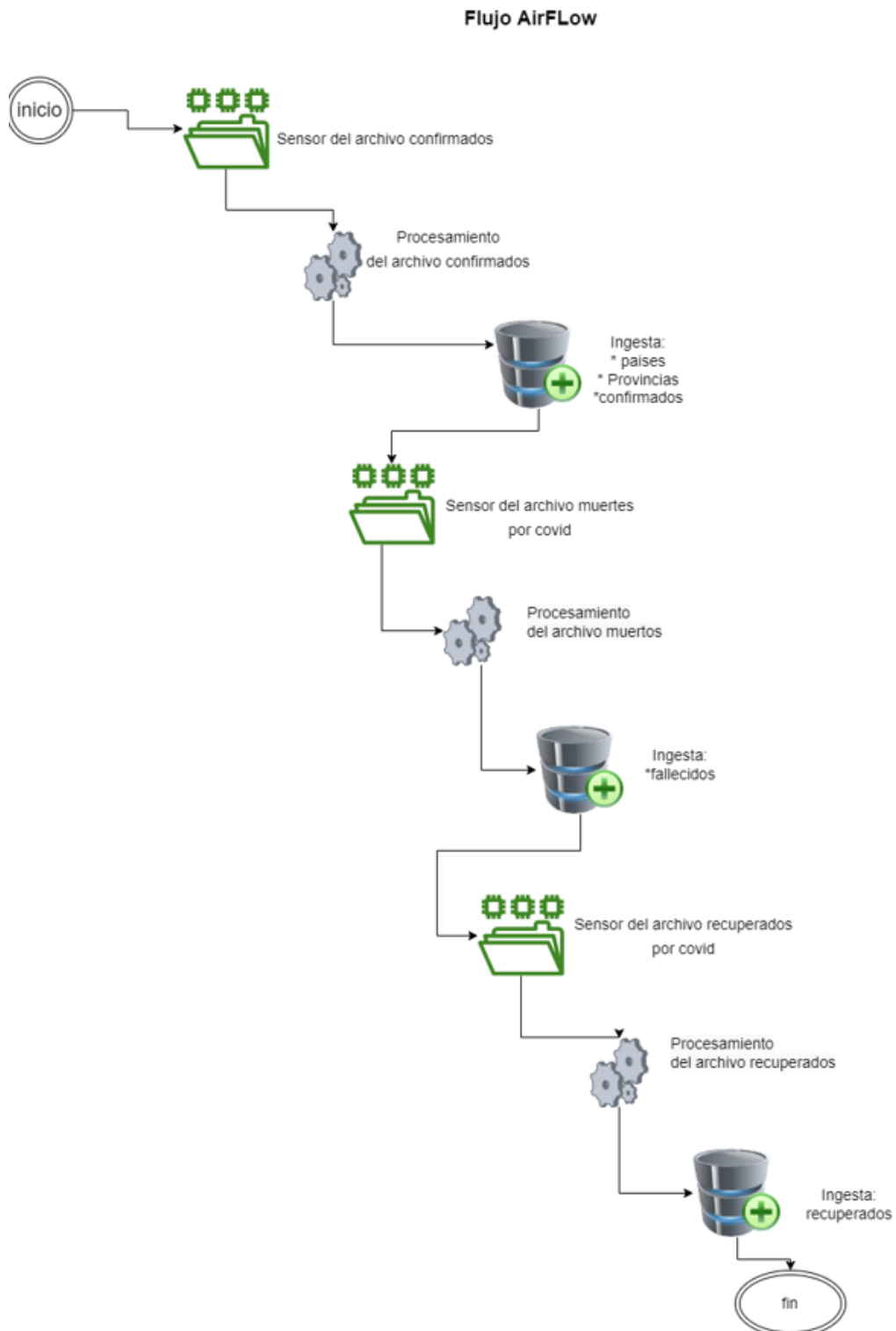
### d. Airflow:

Apache Airflow es una herramienta de tipo workflow manager que utiliza el lenguaje de programación Python, sus funciones principales son: gestionar, monitorizar y planificar flujos de trabajo, usada como orquestador de servicios.

### 3. Diagramas

En esta sección se brinda un detalle del flujo que realiza airflow para realizar el proceso de ETL.

#### a. Diagrama ETL



### b. Proceso ETL

El proceso consiste en tres sensores que se mantienen escuchando una carpeta asignada a una conexión de tipo File, donde cuando llega el archivo especificado, puede avanzar en el flujo.

El primer flujo es el mas importante, debido a que en este se realiza la carga de los países y sus provincias, esta carga a base de datos solo se realiza en este punto, los dos flujos siguientes solo realizan carga a su tabla correspondiente.

Cada procesamiento de archivos realiza las siguientes tareas:

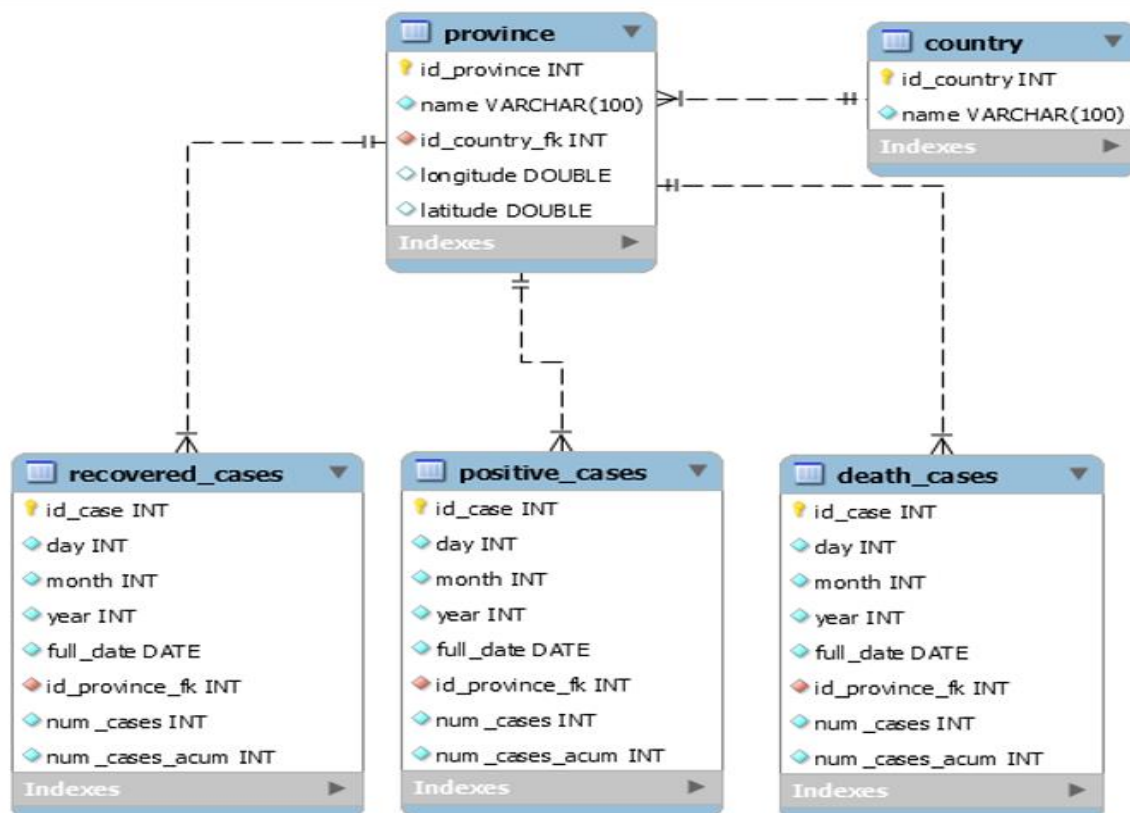
- Elimina registros que tengan latitud y longitud vacíos.
- Que no tengan ningún caso registrado.
- Elimina caracteres especiales que pueden interferir en la relación de las tablas.

Cada ingesta de datos a base de datos para los casos confirmados, fallecidos y recuperados realiza lo siguiente:

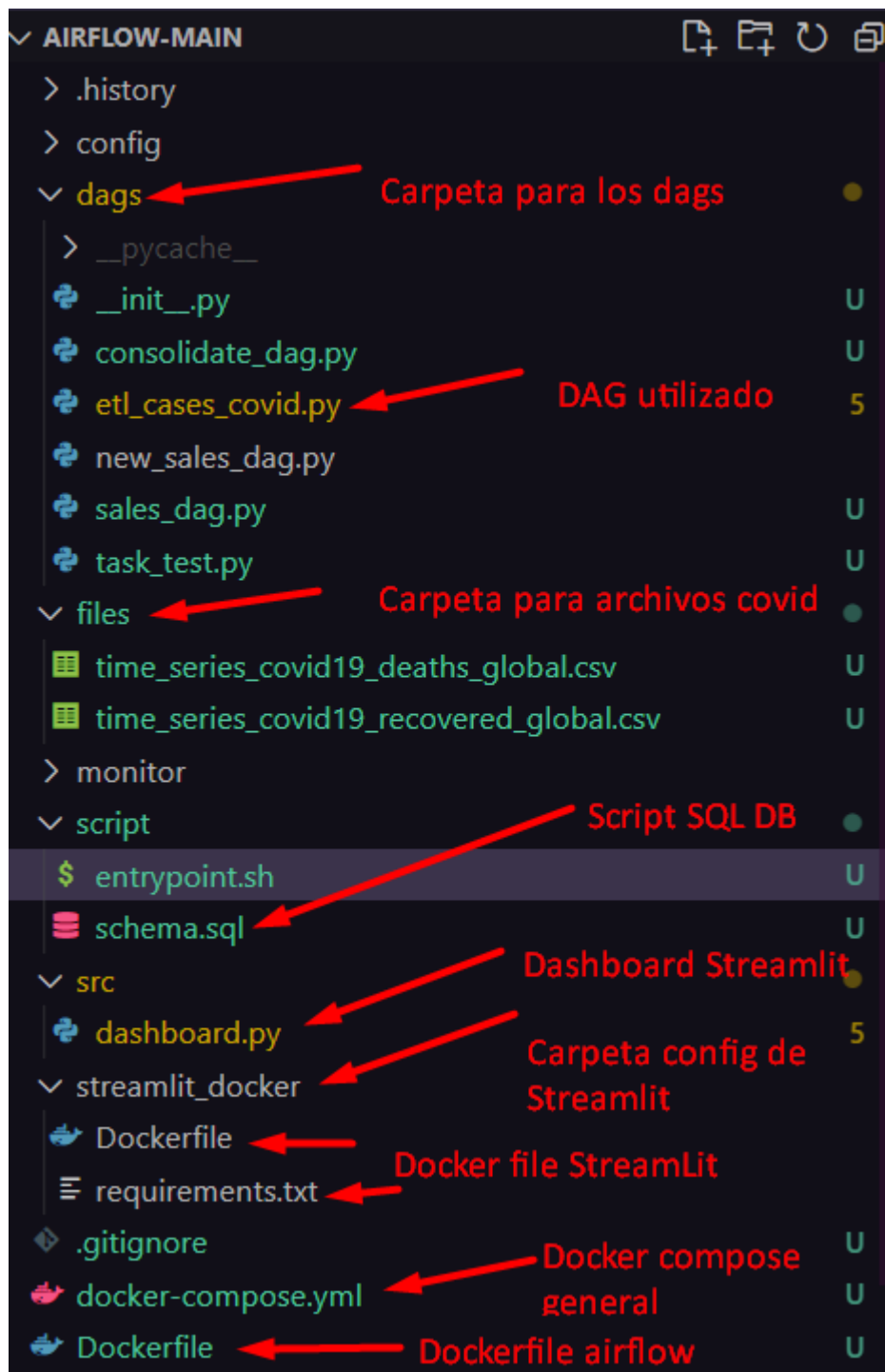
- Se dispone en memoria la llave correspondiente a su país/provincia.
- Consulta a un diccionario por la llave correspondiente, y realiza una resta de los casos detectados el día anterior con el actual para tener el registro correcto de nuevos casos por día.

### c. Diagrama Relacional

Se muestra el modelo relacional utilizado en la aplicación.



4. Deployment de la aplicación
- a. Estructura del proyecto:



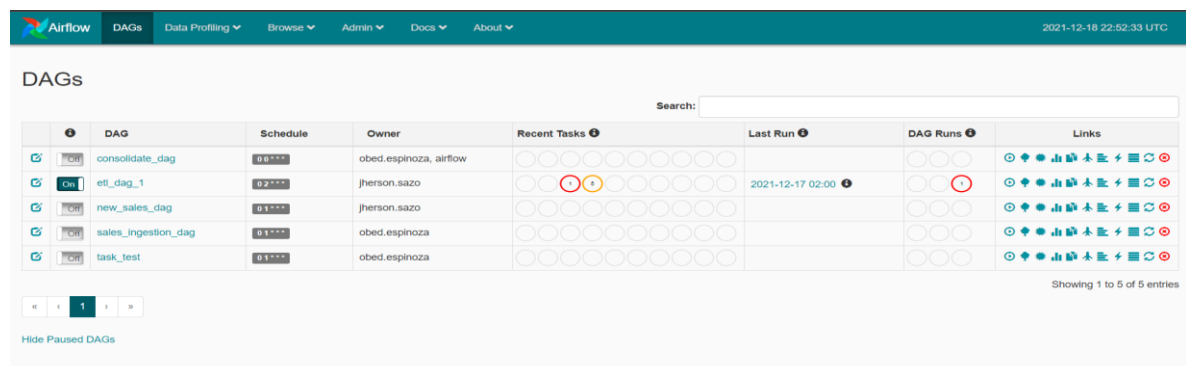
## b. Ejecución

Para ejecutar la aplicación, se requiere abrir una terminal en el directorio del proyecto, y ejecutar `docker-compose up`.

Si se cumplen correctamente la instalación de las dependencias de cada contenedor, deberá aparecer en consola lo siguiente:

```
webserver_1 |
webserver_1 |
webserver_1 |
webserver_1 |
webserver_1 |
webserver_1 | [2021-12-18 21:41:27,419] {{_init_.py:50}} INFO - Using executor LocalExecutor
webserver_1 | /usr/local/lib/python3.7/site-packages/airflow/configuration.py:418: DeprecationWarning: The max_threads option in [s
med to parsing processes - the old setting has been used, but please update your config.
webserver_1 |     return int(self.get(section, key, **kwargs))
webserver_1 | [2021-12-18 21:41:27,430] {{scheduler_job.py:1351}} INFO - Starting the scheduler
webserver_1 | [2021-12-18 21:41:27,430] {{scheduler_job.py:1359}} INFO - Running execute loop for -1 seconds
webserver_1 | [2021-12-18 21:41:27,431] {{scheduler_job.py:1360}} INFO - Processing each file at most -1 times
webserver_1 | [2021-12-18 21:41:27,431] {{scheduler_job.py:1363}} INFO - Searching for files in /usr/local/airflow/dags
webserver_1 |
webserver_1 |
webserver_1 |
webserver_1 |
webserver_1 | /usr/local/lib/python3.7/site-packages/airflow/settings.py:490: DeprecationWarning: `session_lifetime_days` option for
```

Validar en los puertos 8080 del host:



| DAG                 | Schedule  | Owner                  | Recent Tasks  | Last Run         | DAG Runs  | Links  |
|---------------------|-----------|------------------------|---|------------------|---|--|
| consolidate_dag     | 0 0 * * * | obed.espinosa, airflow | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> |                  | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> | <a href="#">🔍</a> <a href="#">📊</a> <a href="#">📅</a> <a href="#">🔧</a> <a href="#">🔄</a> <a href="#">🗑️</a> <a href="#">🔗</a> |
| eti_dag_1           | 0 2 * * * | jherson.sazo           | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> | 2021-12-17 02:00 | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> | <a href="#">🔍</a> <a href="#">📊</a> <a href="#">📅</a> <a href="#">🔧</a> <a href="#">🔄</a> <a href="#">🗑️</a> <a href="#">🔗</a> |
| new_sales_dag       | 0 1 * * * | jherson.sazo           | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> |                  | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> | <a href="#">🔍</a> <a href="#">📊</a> <a href="#">📅</a> <a href="#">🔧</a> <a href="#">🔄</a> <a href="#">🗑️</a> <a href="#">🔗</a> |
| sales_ingestion_dag | 0 1 * * * | obed.espinosa          | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> |                  | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> | <a href="#">🔍</a> <a href="#">📊</a> <a href="#">📅</a> <a href="#">🔧</a> <a href="#">🔄</a> <a href="#">🗑️</a> <a href="#">🔗</a> |
| task_test           | 0 1 * * * | obed.espinosa          | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> |                  | <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> | <a href="#">🔍</a> <a href="#">📊</a> <a href="#">📅</a> <a href="#">🔧</a> <a href="#">🔄</a> <a href="#">🗑️</a> <a href="#">🔗</a> |

Validar en el puerto 8501 para el dashboard en streamlit:



## 5. Requerimientos

Para que la aplicación pueda ser replicada de forma local o en un servidor remoto, este debe cumplir con los siguientes programas instalados:

- Python 3.7 o superior
- Docker cli, Docker server
- Docker compose