

Curso: Tecnologia em Data Science, Big Data, BI & Data Engineering

Projeto: CHALLENGE (012024)

Turmas do 1o ano: 1TSCPV

Sprint 3

**DataStorm**

Ana Beatriz Azevedo, RM557420

Heloiza Oliveira, RM558881

Isabelle Nahas, RM557405

Matheus Madrid, RM555799

Sara Sitta, RM555113

**DISCIPLINA: ARCHITECTURE ANALYTICS & NOSQL**

**Professores: Leandro Romualdo da Silva, Milton Goya**

São Paulo, Setembro de 2024

## Documentação do Pipeline de Dados

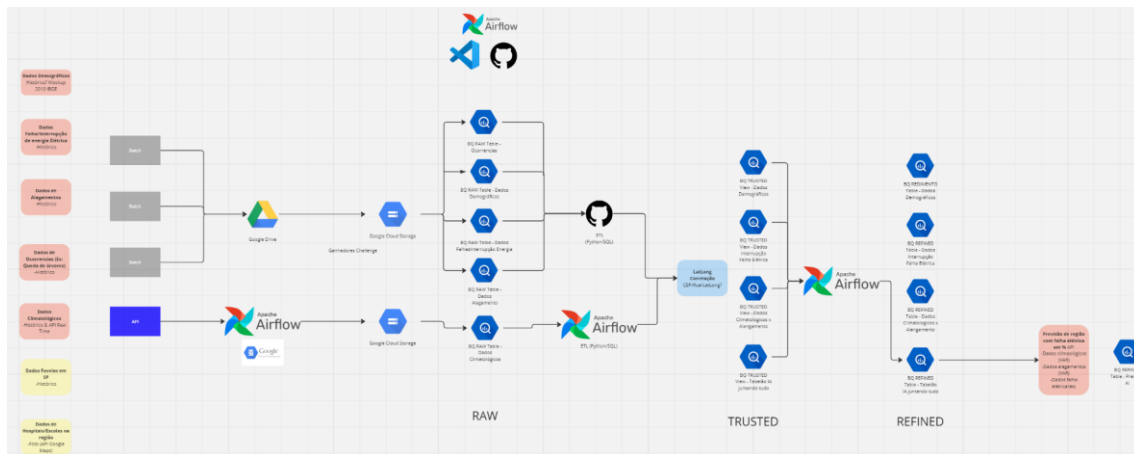
### Visão Geral

Este documento descreve o pipeline de dados implementado para ingestão, transformação, armazenamento e consumo de dados climáticos coletados de uma API, usando Apache Airflow no Google Cloud Composer, Google Cloud Storage como Data Lake, e BigQuery para armazenamento e transformação dos dados. O pipeline é composto por três DAGs principais, cada uma responsável por uma etapa específica do fluxo de dados.

### 1. Ferramenta de ETL/Ingestão

Para atender aos requisitos de coleta, ingestão, armazenamento, transformação e orquestração, foram utilizados Apache Airflow no Google Cloud Composer para orquestração, Google Cloud Storage para armazenamento temporário, e BigQuery para transformação e armazenamento dos dados estruturados.

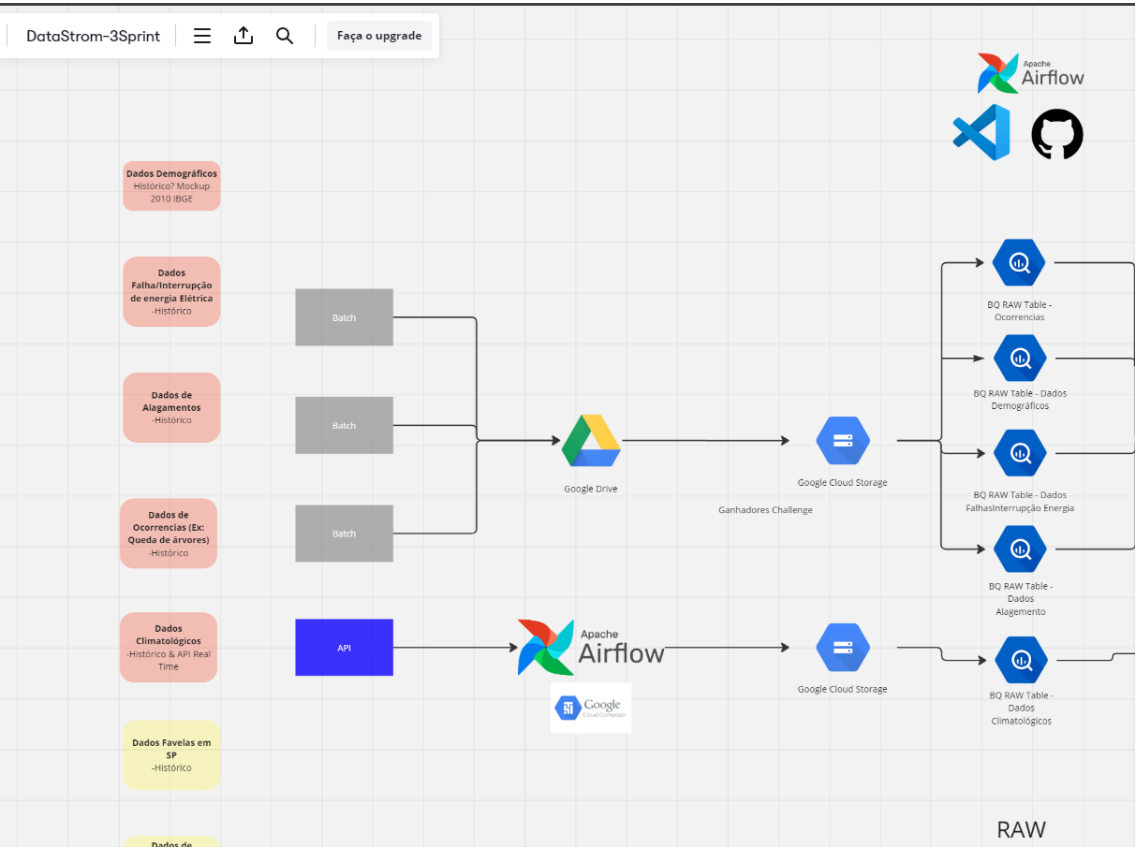
### Arquitetura atualizada:



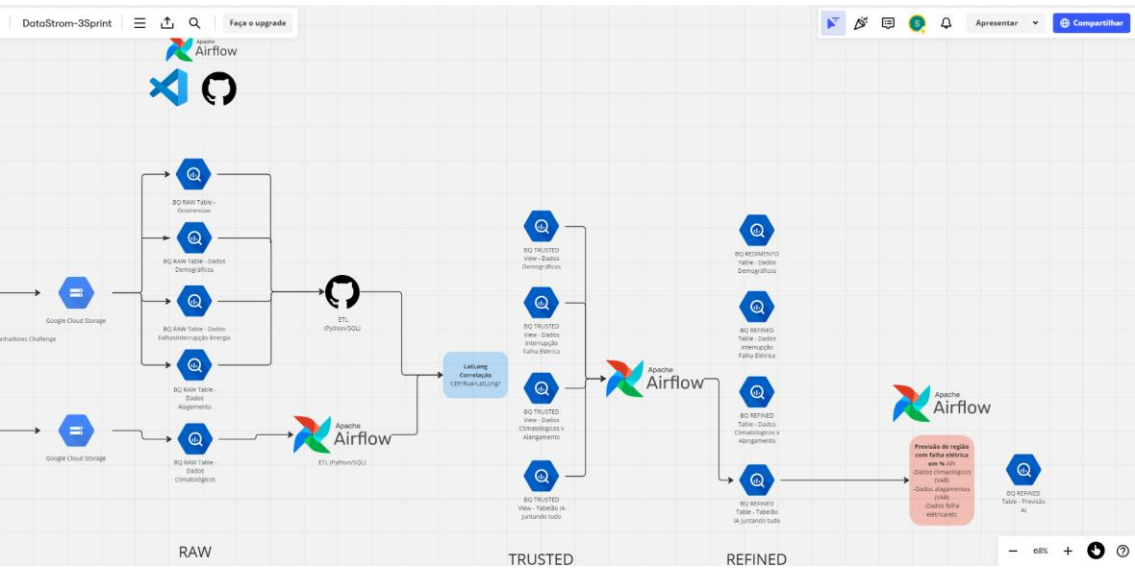
Link: [https://miro.com/app/board/uXjVKmDD4gU=?share\\_link\\_id=695605839813](https://miro.com/app/board/uXjVKmDD4gU=?share_link_id=695605839813)

[https://miro.com/welcomeonboard/dnRiaHI0VEJPOW9KdW84ZUNCanJBTTf4dkpXTkQ2M3ZRNkFOend0cmxsRWntZGIKNm8yYUtlWDBHExF1Qk5CVHwzNDU4NzY0NTc3MzU0NTcyMjQ0fDI=?share\\_link\\_id=512033522479](https://miro.com/welcomeonboard/dnRiaHI0VEJPOW9KdW84ZUNCanJBTTf4dkpXTkQ2M3ZRNkFOend0cmxsRWntZGIKNm8yYUtlWDBHExF1Qk5CVHwzNDU4NzY0NTc3MzU0NTcyMjQ0fDI=?share_link_id=512033522479)

Parte1(ArquiteturaAtt)



Parte2(ArqAtt)



Google Cloud Composer

console.cloud.google.com/composer/environments/detail/us-central1/weather-prod-us-central1/dags?project=elated-drive-432523-s4

Free trial status: R\$1,814.32 credit and 65 days remaining. Activate your full account to get unlimited access to all of Google Cloud—use any remaining credits, then pay only for what you use.

Google Cloud Data Storm Search (/) for resources, docs, products, and more

Navigation menu Environment details OPEN AIRFLOW UI OPEN DAGS FOLDER SAVE SNAPSHOT LOAD SNAPSHOT REFRESH DELETE LEARN

weather-prod-us-central1 This environment is running

MONITORING LOGS DAGS ENVIRONMENT CONFIGURATION AIRFLOW CONFIGURATION OVERRIDES ENVIRONMENT VARIABLES LABELS PYPI PACKAGES

Filter Filter DAGs

DAG id	State	Description	Schedule interval	Last completed run	Active runs	Successful runs (1h)	Failed runs (1h)
airflow_monitoring	Active	liveness monitoring dag	* / 10 * * * *	2 minutes ago	0	6	0
etl_raw_to_trusted_weather	Active	Pipeline to process WEATHER and SUBPREFECTURES tables...	@hourly	2 minutes ago	0	1	0
etl_trusted_to_refined_weather	Active	Pipeline to move WEATHER data from TRUSTED to REFINED...	@hourly	2 minutes ago	0	3	0
weather_data_pipeline	Paused	DAG para coletar dados de clima e enviar para GCS e BigQuery	@hourly	4 hours ago	0	0	0

Apache Airflow:

900bd82ac7824757a102b0b45d253f8-dot-us-central1.composer.googleusercontent.com/home

Airflow DAGs Cluster Activity Datasets Browse Admin Docs Composer 02:03 UTC

weather-prod-us-central1

Active 3 Paused 1 Running 1 Failed 0 Filter DAGs by tag Search DAGs Auto-refresh

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
airflow_monitoring	airflow	1	* / 10 * * * *	2024-09-09, 01:50:00	2024-09-09, 02:00:00	1	▶ 🗑	...
etl_raw_to_trusted_weather	airflow	1	@hourly	2024-09-09, 01:00:00	2024-09-09, 02:00:00	1	▶ 🗑	...
etl_trusted_to_refined_weather	airflow	1	@hourly	2024-09-09, 01:35:10	2024-09-09, 02:00:00	1	▶ 🗑	...
weather_data_pipeline	airflow	1	@hourly	2024-09-08, 21:00:00	2024-09-09, 01:00:00	1	▶ 🗑	...

Showing 1-4 of 4 DAGs

CloudStorage – Landing Zone

console.cloud.google.com/storage/browser/us-central1-datastorm-prod-us-a4c8a156-landingzone/LANDING\_ZONE\_WEATHER?project=elated-drive-432523-s4&pageState=({"StorageObjectListTable":{"C":{"%255B%25D50}})

Free trial status: R\$1,810.37 credit and 65 days remaining. Activate your full account to get unlimited access to all of Google Cloud—use any remaining credits, then pay only for what you use.

Google Cloud Data Storm Search (/) for resources, docs, products, and more

Cloud Storage Buckets Bucket details GO TO PATH REF

us-central1-datastorm-prod-us-a4c8a156-landingzone

Location: us (multiple regions in United States) Storage class: Standard Public access: Not public Protection: Soft Delete

OBJECTS CONFIGURATION PERMISSIONS PROTECTION LIFECYCLE OBSERVABILITY INVENTORY REPORTS OPERATIONS

Folder browser

us-central1-datastorm-prod-us-a4c8a156-landingzone

LANDING\_ZONE\_WEATHER/

CREATE FOLDER UPLOAD TRANSFER DATA MORE SERVICES

Filter by name prefix only Filter Filter objects and folders Show Live object

Name	Size	Type	Created	Storage class	Last modified	Public access
historical_weather_data.json	303.6 KB	application/octet-stream	Sep 8, 2024, 11:04:29 PM	Standard	Sep 8, 2024, 11:04:29 PM	Not public

Bigquery GCP

The screenshot shows the Google Cloud BigQuery console. On the left, the 'Explorer' pane displays a project structure with folders for 'elated-drive-432523-p4', 'RAW', 'SUBPRE...', 'WEATHER', and 'REFINED'. The 'WEATHER' folder is selected, showing a table named 'WEATHER'. The main pane displays the 'SCHEMA' tab for the 'WEATHER' table. The schema table lists various weather-related fields with their data types and modes.

Field name	Type	Mode	Key	Collation	Default Value	Policy Tags	Description
SubprefectureName	STRING	NULLABLE	-	-	-	-	Standardized name of the subprefecture from the Open Weather API
Latitude	FLOAT	NULLABLE	-	-	-	-	Subprefecture Latitude
Longitude	FLOAT	NULLABLE	-	-	-	-	Subprefecture Longitude
ObservationTime	TIMESTAMP	NULLABLE	-	-	-	-	Datetime of the weather conditions observation
Temperature	FLOAT	NULLABLE	-	-	-	-	Recorded temperature in Kelvin
FeelsLike	FLOAT	NULLABLE	-	-	-	-	Apparent temperature (feels like) in Kelvin
Pressure	INTEGER	NULLABLE	-	-	-	-	Atmospheric pressure in hectopascals (hPa)
Humidity	INTEGER	NULLABLE	-	-	-	-	Relative humidity percentage
MinTemperature	FLOAT	NULLABLE	-	-	-	-	Minimum temperature recorded in Kelvin
MaxTemperature	FLOAT	NULLABLE	-	-	-	-	Maximum temperature recorded in Kelvin
WindSpeed	FLOAT	NULLABLE	-	-	-	-	Wind speed in meters per second
WindDirection	INTEGER	NULLABLE	-	-	-	-	Wind direction in degrees
WindGust	FLOAT	NULLABLE	-	-	-	-	Wind gust speed in meters per second
CloudCoverage	INTEGER	NULLABLE	-	-	-	-	Percentage of cloud cover
WeatherID	INTEGER	NULLABLE	-	-	-	-	Identifier for the weather type
WeatherMain	STRING	NULLABLE	-	-	-	-	Main meteorology of the weather (e.g. Cloudy, Rain)

## 1.1 Coleta e Ingestão de Dados para o Data Lake e BigQuery (DAG: weather\_data\_pipeline\_dag)

- **Nome da DAG:** weather\_data\_pipeline\_dag
- **Ferramenta Utilizada:** Apache Airflow com Google Cloud Composer para orquestração; Google Cloud Storage para armazenamento dos dados e BigQuery para estruturação do Lake
- **Processo:**
  - **Coleta de Dados:** A DAG coleta dados climáticos da API do OpenWeatherMap em intervalos regulares (a cada 1 hora).
  - **Armazenamento Temporário:** Os dados coletados são salvos no Google Cloud Storage, que atua como o Data Lake.
  - **Carga para BigQuery:** Após salvar no Cloud Storage, os dados são carregados na camada RAW do BigQuery, utilizando operadores específicos do Airflow para transferência de arquivos do Cloud Storage para o BigQuery.

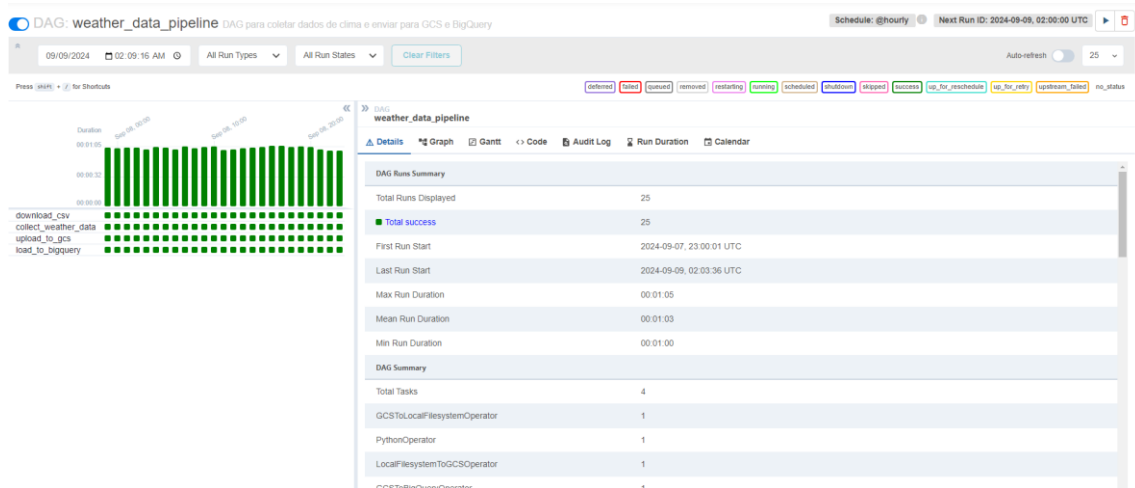
```

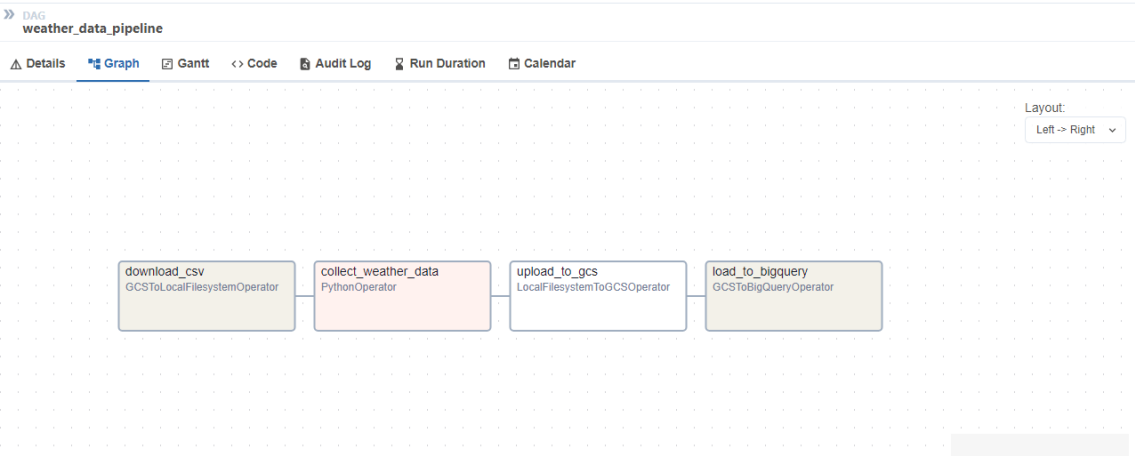
weather_data_pipeline_dag.py > ...
1 from airflow import DAG
2 from airflow.operators.python import PythonOperator
3 from airflow.providers.google.cloud.transfers.local_to_gcs import LocalFileSystemToGCSOperator
4 from airflow.providers.google.cloud.transfers.gcs_to_bigquery import GCSToBigQueryOperator
5 from airflow.providers.google.cloud.transfers.gcs_to_local import GCSToLocalFileSystemOperator
6 from airflow.utils.dates import days_ago
7 import pandas as pd
8 import requests
9 import time
10 import json
11 from datetime import datetime
12
13 # Função para coletar dados das subprefeituras e salvar em formato adequado para o BigQuery
14 def collect_weather_data(**kwargs):
15     # Baixar o CSV das subprefeituras diretamente do Google Cloud Storage
16     subprefeituras = pd.read_csv('/tmp/subprefeituras-sp.csv', sep=';', header=0)
17     subprefeituras.columns = subprefeituras.columns.str.strip().str.lower()
18
19     API_KEY = '-' # Substitua pela sua chave da API
20     BASE_URL = 'https://history.openweathermap.org/data/2.5/history/city'
21
22     results = []
23     for index, row in subprefeituras.iterrows():
24         lat = round(row['latitude'], 2)
25         lon = round(row['longitude'], 2)
26         subprefeitura_nome = row['subprefeitura']
27         url = f'{BASE_URL}?lat={lat:.2f}&lon={lon:.2f}&appid={API_KEY}'
28
29         response = requests.get(url)
30         if response.status_code == 200:
31             data = response.json()
32             if 'list' in data:
33                 for entry in data['list']:
34                     result = {
35                         'subprefeitura': subprefeitura_nome,
36                         'latitude': lat,
37                         'longitude': lon,
38                         'timestamp': datetime.utcfromtimestamp(entry['dt']).isoformat(), # Converter para timestamp ISO
39                         'temperature': entry['main'].get('temp'),
40                         'feels_like': entry['main'].get('feels_like'),
41                         'pressure': entry['main'].get('pressure'),
42                         'humidity': entry['main'].get('humidity'),
43                         'temp_min': entry['main'].get('temp_min'),
44                         'temp_max': entry['main'].get('temp_max'),
45                         'wind_speed': entry['wind'].get('speed'),
46                         'wind_deg': entry['wind'].get('deg')

```

Arquivo anexado - weather\_data\_pipeline\_dag.py

weather\_data\_pipeline\_dag. DAG no Airflow





» DAG weather\_data\_pipeline

Details Graph Gantt Code Audit Log Run Duration Calendar

Parsed at: 2024-09-09, 02:07:40 UTC

```
96     task_id='collect_weather_data',
97     python_callable=collect_weather_data,
98     provide_context=True,
99 )
100
101 # Task 3: Enviar JSON para o Cloud Storage na pasta especificada
102 upload_to_gcs = LocalFilesystemToGCSOperator(
103     task_id='upload_to_gcs',
104     src='/tmp/historical_weather_data.json',
105     dst='LANDING_ZONE_WEATHER/historical_weather_data.json',
106     bucket='us-central1-datastorm-prod-us-a4c8a156-landingzone',
107 )
108
109 # Task 4: Carregar dados do GCS para BigQuery
110 load_to_bigquery = GCSToBigQueryOperator(
111     task_id='load_to_bigquery',
112     bucket='us-central1-datastorm-prod-us-a4c8a156-landingzone',
113     source_objects=['LANDING_ZONE_WEATHER/historical_weather_data.json'],
114     destination_project_dataset_table='elated-drive-432523-s4:RAW.WEATHER', # Substitua pelo seu projeto e dataset
115     source_format='NEWLINE_DELIMITED_JSON',
116     write_disposition='WRITE_APPEND',
117 )
118
119 # Definir a sequência das tasks
120 download_csv >> collect_data_task >> upload_to_gcs >> load_to_bigquery
121
```

Toggle Wrap

## LANDING ZONE no Google Cloud Storage

← Bucket details GO TO PATH REF

parent page

**us-central1-datastorm-prod-us-a4c8a156-landingzone**

Location	Storage class	Public access	Protection
us (multiple regions in United States)	Standard	Not public	Soft Delete

OBJECTS CONFIGURATION PERMISSIONS PROTECTION LIFECYCLE OBSERVABILITY INVENTORY REPORTS OPERATIONS

Folder browser ⏮

us-central1-datastorm-prod-us-a4c8a156-landingzone

LANDING\_ZONE\_WEATHER/

Buckets > us-central1-datastorm-prod-us-a4c8a156-landingzone > LANDING\_ZONE\_WEATHER

CREATE FOLDER UPLOAD TRANSFER DATA MORE SERVICES

Filter by name prefix only Filter Filter objects and folders Show Live objects

<input type="checkbox"/>	Name	Size	Type	Created	Storage class	Last modified	Public access
<input type="checkbox"/>	historical_weather_data.json	303.6 KB	application/octet-stream	Sep 8, 2024, 11:04:29 PM	Standard	Sep 8, 2024, 11:04:29 PM	Not public

## Camada RAW.WEATHER

Explorer + ADD IK

Search BigQuery resources

Viewing resources. [SHOW STARRED ONLY](#)

- elated-drive-432523-s4
  - Queries
    - Shared queries
    - ExploratoryDataAnalysis-We...
  - Notebooks
  - Data canvases
  - Data preparations
  - External connections
  - RAW
    - SUBPREFECTURES
    - WEATHER**
    - WEATHER\_HISTORY

WEATHER QUERY SHARE COPY SNAPSHOT DELETE EXPOR

**SCHEMA** DETAILS PREVIEW TABLE EXPLORER **PREVIEW** INSIGHTS LINEAGE DA

Filter Enter property name or value

<input type="checkbox"/>	Field name	Type	Mode	Key	Collation	Default Value	Policy Tags
<input type="checkbox"/>	weather_main	STRING	NULLABLE	-	-	-	-
<input type="checkbox"/>	weather_id	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	clouds_all	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	wind_deg	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	pressure	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	temp_min	FLOAT	NULLABLE	-	-	-	-
<input type="checkbox"/>	wind_speed	FLOAT	NULLABLE	-	-	-	-
<input type="checkbox"/>	humidity	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	feels_like	FLOAT	NULLABLE	-	-	-	-

## 1.2 Estruturação e Preparação dos Dados para a Camada Estruturada (DAG: etl\_raw\_to\_trusted\_weather)

- **Nome da DAG:** etl\_raw\_to\_trusted\_weather
- **Ferramenta Utilizada:** Apache Airflow com Google Cloud Composer para orquestração; Google Cloud Storage para armazenamento dos dados e BigQuery para transformação dos dados.
- **Processo:**
  - Esta DAG transforma e estrutura os dados movendo-os da camada RAW para a camada TRUSTED no BigQuery.
  - **Transformações Incluídas:**
    - Ajuste de precisão de latitude e longitude para garantir a consistência entre os conjuntos de dados.
    - Conversão de CEP de string para inteiro, removendo caracteres indesejados.
    - Criação de views para padronizar os nomes das colunas para o padrão de Data Lake (letra maiúscula, sem underscores).

Task 4 da DAG weather\_data\_pipeline\_dag



>> DAG

weather\_data\_pipeline

Details

Graph

Gantt

<> Code

Audit Log

Run Duration

Calendar

Parsed at: 2024-09-09, 02:07:40 UTC

```
96     task_id='collect_weather_data',
97     python_callable=collect_weather_data,
98     provide_context=True,
99 )
100
101 # Task 3: Enviar JSON para o Cloud Storage na pasta especificada
102 upload_to_gcs = LocalFilesystemToGCSOperator(
103     task_id='upload_to_gcs',
104     src='/tmp/historical_weather_data.json',
105     dst='LANDING_ZONE_WEATHER/historical_weather_data.json',
106     bucket='us-central1-datastorm-prod-us-a4c8a156-landingzone',
107 )
108
109 # Task 4: Carregar dados do GCS para BigQuery
110 load_to_bigquery = GCSToBigQueryOperator(
111     task_id='load_to_bigquery',
112     bucket='us-central1-datastorm-prod-us-a4c8a156-landingzone',
113     source_objects=['LANDING_ZONE_WEATHER/historical_weather_data.json'],
114     destination_project_dataset_table='elated-drive-432523-s4:RAW.WEATHER', # Substitua pelo seu projeto e dataset
115     source_format='NEWLINE_DELIMITED_JSON',
116     write_disposition='WRITE_APPEND',
117 )
118
119 # Definir a sequência das tasks
120 download_csv >> collect_data_task >> upload_to_gcs >> load_to_bigquery
121
```

## Camada RAW.WEATHER

Explorer

+ ADD

<

Search BigQuery resources

Viewing resources.

SHOW STARRED ONLY

elated-drive-432523-s4

Queries

Shared queries

ExploratoryDataAnalysis-We...

Notebooks

Data canvases

Data preparations

External connections

RAW

SUBPREFECTURES

WEATHER

WEATHER\_HISTORY

WEATHER

QUERY

SHARE

COPY

SNAPSHOT

DELETE

EXPOR

SCHEMA

DETAILS

PREVIEW

TABLE EXPLORER

PREVIEW

INSIGHTS

LINEAGE

DA

Filter

Enter property name or value

	Field name	Type	Mode	Key	Collation	Default Value	Policy Tags
<input type="checkbox"/>	weather_main	STRING	NULLABLE	-	-	-	-
<input type="checkbox"/>	weather_id	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	clouds_all	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	wind_deg	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	pressure	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	temp_min	FLOAT	NULLABLE	-	-	-	-
<input type="checkbox"/>	wind_speed	FLOAT	NULLABLE	-	-	-	-
<input type="checkbox"/>	humidity	INTEGER	NULLABLE	-	-	-	-
<input type="checkbox"/>	feels_like	FLOAT	NULLABLE	-	-	-	-

## etl\_raw\_to\_trusted\_weather DAG para refinamento

etl\_raw\_to\_trusted\_weather

900b8b2ac7b24757a102b0b45d253f8b-dot-us-central1-composer.googleusercontent.com/dags/etl\_raw\_to\_trusted\_weather/gnd

02:13 UTC

DAG: etl\_raw\_to\_trusted\_weather

Pipeline to process WEATHER and SUBPREFECTURES tables from RAW to TRUSTED layer

Schedule: @hourly

Next Run ID: 2024-09-09, 02:00:00 UTC

09/09/2024

02:13:15 AM

All Run Types

All Run States

Clear Filters

Auto-refresh

25

Press <Shift> + J for Shortcuts

etl\_raw\_to\_trusted\_weather

Details

Graph

Gantt

<> Code

Audit Log

Run Duration

Calendar

DAG Run Summary

Total Runs Displayed

8

Total success

5

Total failed

3

First Run Start

2024-09-09, 00:10:48 UTC

Last Run Start

2024-09-09, 02:00:00 UTC

Max Run Duration

00:05:24

Mean Run Duration

00:02:46

Min Run Duration

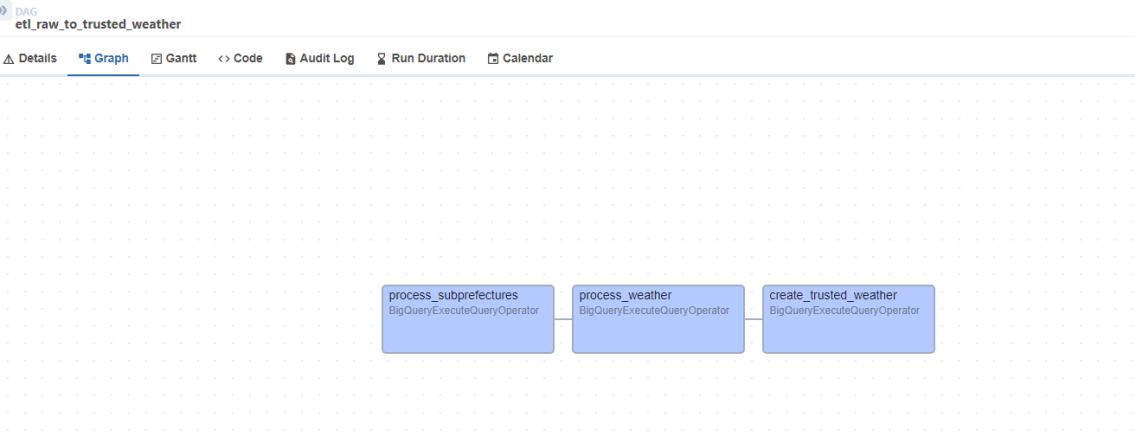
00:00:12

DAG Summary

Total Tasks

3

Tasks da DAG



DAG  
etl\_raw\_to\_trusted\_weather

Details Graph Gantt **Code** Audit Log Run Duration Calendar

Parsed at: 2024-09-09, 02:11:32 UTC

```
19 # Step 1: Create VIEW for SUBPREFECTURES in TRUSTED directly from RAW
20 process_subprefectures = BigQueryExecuteQueryOperator(
21     task_id='process_subprefectures',
22     sql="""
23         CREATE OR REPLACE VIEW `elated-drive-432523-s4.TRUSTED.SUBPREFECTURES` AS
24         SELECT
25             REPLACE(UPPER(SUBSTRING(Subprefeitura, 1, 1)) || LOWER(SUBSTRING(Subprefeitura, 2)), ' ', '') AS SubprefectureName,
26             ROUND(Latitude, 2) AS Latitude, -- Rounding to 2 decimal places
27             ROUND(Longitude, 2) AS Longitude, -- Rounding to 2 decimal places
28             CAST(REPLACE(CEP, '-', '') AS INT64) AS PostalCode
29         FROM `elated-drive-432523-s4.RAW.SUBPREFECTURES`
30     """,
31     use_legacy_sql=False
32 )
33
34 # Step 2: Create VIEW for WEATHER in TRUSTED directly from RAW
35 process_weather = BigQueryExecuteQueryOperator(
36     task_id='process_weather',
37     sql="""
38         CREATE OR REPLACE VIEW `elated-drive-432523-s4.TRUSTED.WEATHER_API` AS
39         SELECT
40             REPLACE(UPPER(SUBSTRING(subprefeitura, 1, 1)) || LOWER(SUBSTRING(subprefeitura, 2)), ' ', '') AS SubprefectureName,
41             ROUND(Latitude, 2) AS Latitude, -- Rounding to 2 decimal places
42             ROUND(Longitude, 2) AS Longitude, -- Rounding to 2 decimal places
43             TIMESTAMP(timestamp) AS ObservationTime,
44             temperature AS Temperature,
```

Dag anexada no zip etl\_raw\_to\_trusted\_weather.py

CAMADA TRUSTED

Explorer + ADD IK

Q Search BigQuery resources

Viewing resources. [SHOW STARRED ONLY](#)

elated-drive-432523-s4

- Queries
- Notebooks
- Data canvases
- Data preparations
- External connections
- RAW
- REFINED
- TRUSTED
  - SUBPREFECTURES
  - WEATHER**
  - WEATHER\_API
- openweathermap\_historic

WEATHER

SCHEMA DETAILS TABLE EXPLORER **PREVIEW** INSIGHTS LINEAGE DATA PROFILE DATA QUALITY

Filter Enter property name or value

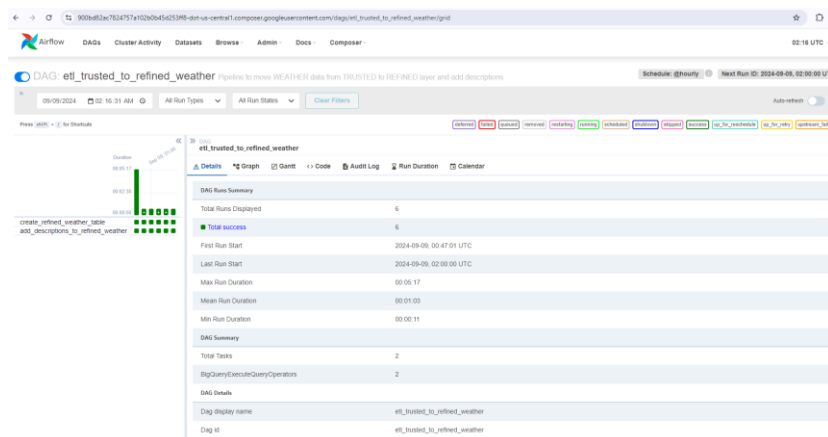
<input type="checkbox"/>	Field name	Type	Mode	Key	Collation	Default Value	Policy Tags	Description
<input type="checkbox"/>	SubprefectureName	STRING	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	Latitude	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	Longitude	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	ObservationTime	TIMESTAMP	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	Temperature	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	FeelsLike	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	Pressure	INTEGER	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	Humidity	INTEGER	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	MinTemperature	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	MaxTemperature	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	WindSpeed	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	WindDirection	INTEGER	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	WindGust	FLOAT	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	CloudCoverage	INTEGER	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	WeatherID	INTEGER	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	WeatherMain	STRING	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	WeatherDescription	STRING	NULLABLE	-	-	-	-	-
<input type="checkbox"/>	WeatherIcon	STRING	NULLABLE	-	-	-	-	-

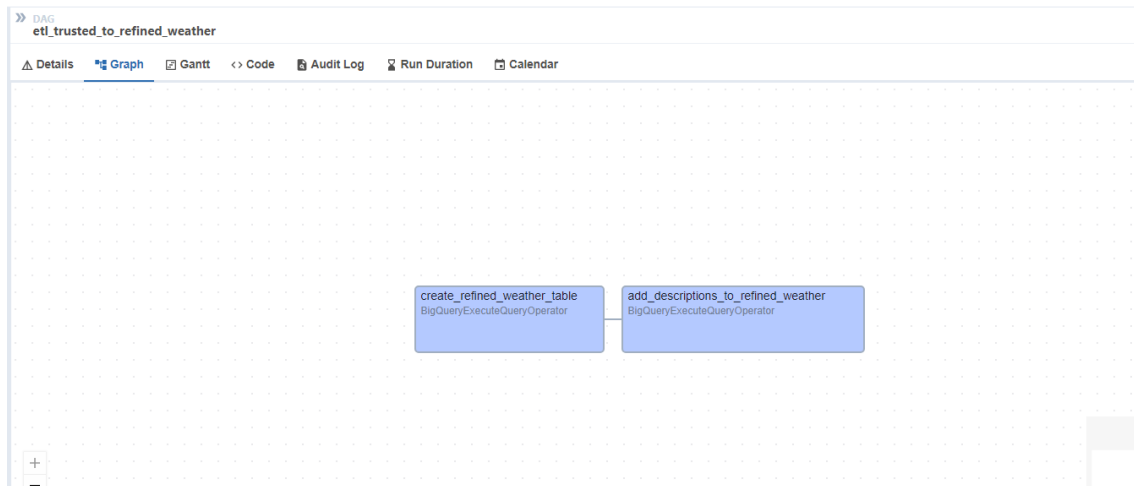
[EDIT SCHEMA](#)

### 1.3 Refinamento e Enriquecimento dos Dados (DAG: etl\_trusted\_to\_refined\_weather)

- **Nome da DAG:** etl\_trusted\_to\_refined\_weather
- **Ferramenta Utilizada:** Apache Airflow com Google Cloud Composer para orquestração; BigQuery para refinamento e enriquecimento dos dados.
- **Processo:**
  - A última DAG move os dados da camada TRUSTED para a camada REFINED no BigQuery.
  - **Refinamento Inclui:**
    - Criação de tabelas finais na camada REFINED.
    - Adição de descrições das colunas para melhorar a documentação e a compreensão dos dados.
    - Junção de dados climáticos com informações de subprefeituras para fornecer um conjunto de dados enriquecido.

DAG etl\_trusted\_to\_refined\_weather no Airflow





» DAG  
etl\_trusted\_to\_refined\_weather

Details Graph Gantt **Code** Audit Log Run Duration Calendar

Parsed at: 2024-09-09, 02:15:25 UTC

```
29 # Step 2: Consolidated ALTER TABLE to Add Descriptions to REFINED.WEATHER Table
30 add_descriptions_to_refined_weather = BigQueryExecuteQueryOperator(
31     task_id='add_descriptions_to_refined_weather',
32     sql="""
33         ALTER TABLE `elated-drive-432523-s4.REFINED.WEATHER`
34         SET OPTIONS (
35             description = 'Refined table with enriched weather and subprefecture data'
36         );
37
38         ALTER TABLE `elated-drive-432523-s4.REFINED.WEATHER`
39         ALTER COLUMN SubprefectureName SET OPTIONS (description = 'Standardized name of the subprefecture from the Open Weather API')
40         ALTER COLUMN Latitude SET OPTIONS (description = 'Subprefecture Latitude'),
41         ALTER COLUMN Longitude SET OPTIONS (description = 'Subprefecture Longitude'),
42         ALTER COLUMN ObservationTime SET OPTIONS (description = 'Datetime of the weather conditions obsevation'),
43         ALTER COLUMN Temperature SET OPTIONS (description = 'Recorded temperature in Kelvin'),
44         ALTER COLUMN Feelslike SET OPTIONS (description = 'Apparent temperature (feels like) in Kelvin'),
45         ALTER COLUMN Pressure SET OPTIONS (description = 'Atmospheric pressure in hectopascals (hPa)'),
46         ALTER COLUMN Humidity SET OPTIONS (description = 'Relative humidity percentage'),
47         ALTER COLUMN MinTemperature SET OPTIONS (description = 'Minimum temperature recorded in Kelvin'),
48         ALTER COLUMN MaxTemperature SET OPTIONS (description = 'Maximum temperature recorded in Kelvin'),
49         ALTER COLUMN WindSpeed SET OPTIONS (description = 'Wind speed in meters per second'),
50         ALTER COLUMN WindDirection SET OPTIONS (description = 'Wind direction in degrees'),
51         ALTER COLUMN WindGust SET OPTIONS (description = 'Wind gust speed in meters per second'),
52         ALTER COLUMN CloudCoverage SET OPTIONS (description = 'Percentage of cloud cover'),
53         ALTER COLUMN WeatherID SET OPTIONS (description = 'Identifier for the weather type'),
```

Arquivo anexado no zip etl\_trusted\_to\_refined\_weather.py

CAMADA REFINED com a tabela WEATHER final com as Descrições

Explorer

+

ADD

◀

Viewing resources.

SHOW STARRED ONLY

elated-drive-432523-s4

Q

Queries

📓

Notebooks

📊

Data canvases

🔗

Data preparations

🔗

External connections

RAW

REFINED

WEATHER

TRUSTED

openweathermap\_historic

WEATHER

QUERY

SHARE

COPY

SNAPSHOT

DELETE

EXPORT

SCHEMA

DETAILS

PREVIEW

TABLE EXPLORER

PREVIEW

INSIGHTS

LINEAGE

DATA PROFILE

DATA QUALITY

Filter

Enter property name or value

<input type="checkbox"/>	Field name	Type	Mode	Key	Collation	Default Value	Policy Tags	Description
<input type="checkbox"/>	SubprefectureName	STRING	NULLABLE	-	-	-	-	Standardized name of the subprefecture from the Open Weather API
<input type="checkbox"/>	Latitude	FLOAT	NULLABLE	-	-	-	-	Subprefecture Latitude
<input type="checkbox"/>	Longitude	FLOAT	NULLABLE	-	-	-	-	Subprefecture Longitude
<input type="checkbox"/>	ObservationTime	TIMESTAMP	NULLABLE	-	-	-	-	Datetime of the weather conditions observation
<input type="checkbox"/>	Temperature	FLOAT	NULLABLE	-	-	-	-	Recorded temperature in Kelvin
<input type="checkbox"/>	FeelsLike	FLOAT	NULLABLE	-	-	-	-	Apparent temperature (feels like) in Kelvin
<input type="checkbox"/>	Pressure	INTEGER	NULLABLE	-	-	-	-	Atmospheric pressure in hectopascals (hPa)
<input type="checkbox"/>	Humidity	INTEGER	NULLABLE	-	-	-	-	Relative humidity percentage
<input type="checkbox"/>	MinTemperature	FLOAT	NULLABLE	-	-	-	-	Minimum temperature recorded in Kelvin
<input type="checkbox"/>	MaxTemperature	FLOAT	NULLABLE	-	-	-	-	Maximum temperature recorded in Kelvin
<input type="checkbox"/>	WindSpeed	FLOAT	NULLABLE	-	-	-	-	Wind speed in meters per second
<input type="checkbox"/>	WindDirection	INTEGER	NULLABLE	-	-	-	-	Wind direction in degrees
<input type="checkbox"/>	WindGust	FLOAT	NULLABLE	-	-	-	-	Wind gust speed in meters per second
<input type="checkbox"/>	CloudCoverage	INTEGER	NULLABLE	-	-	-	-	Percentage of cloud cover
<input type="checkbox"/>	WeatherID	INTEGER	NULLABLE	-	-	-	-	Identifier for the weather type
<input type="checkbox"/>	WeatherMain	STRING	NULLABLE	-	-	-	-	Main category of the weather (e.g., Clouds, Rain)
<input type="checkbox"/>	WeatherDescription	STRING	NULLABLE	-	-	-	-	Detailed description of the weather conditions
<input type="checkbox"/>	WeatherIcon	STRING	NULLABLE	-	-	-	-	Code representation the weather icon

2. Implantação de um banco de dados NoSQL. O grupo deve escolher um banco de dados NoSQL e adicionar na sua solução em qualquer fase do pipeline de dados. Pesquise, questione o professor da disciplina e veja o que mais se adequa ao seu cenário e faça a implantação.

Todas as imagens anexadas no zip

```
mongosh mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
... {userId: 9, productId: 6, nota:9, review : "muito bom , ganhou um cliente"},
... {userId: 10, productId: 8, nota:0, review : "Produto defeituoso, em contato com a loja recebi um atendimento que deixou muito a desejar"}
... })
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66be9302e15c498dad228fd1'),
    '1': ObjectId('66be9302e15c498dad228fd2'),
    '2': ObjectId('66be9302e15c498dad228fd3'),
    '3': ObjectId('66be9302e15c498dad228fd4'),
    '4': ObjectId('66be9302e15c498dad228fd5'),
    '5': ObjectId('66be9302e15c498dad228fd6'),
    '6': ObjectId('66be9302e15c498dad228fd7'),
    '7': ObjectId('66be9302e15c498dad228fd8'),
    '8': ObjectId('66be9302e15c498dad228fd9'),
    '9': ObjectId('66be9302e15c498dad228fda')
  }
}
livesmongodb>
```

```
mongosh mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
...
{
  "productId": 10,
  "quantity": 4,
  "userId": 10
},
...
}

acknowledged: true,
insertedIds: {
  '0': ObjectId('60be92f5e15c498dad228fcb'),
  '1': ObjectId('60be92f5e15c498dad228fcb'),
  '2': ObjectId('60be92f5e15c498dad228fcb'),
  '3': ObjectId('60be92f5e15c498dad228fcb'),
  '4': ObjectId('60be92f5e15c498dad228fcb'),
  '5': ObjectId('60be92f5e15c498dad228fcb'),
  '6': ObjectId('60be92f5e15c498dad228fcb'),
  '7': ObjectId('60be92f5e15c498dad228fcb'),
  '8': ObjectId('60be92f5e15c498dad228fcb'),
  '9': ObjectId('60be92f5e15c498dad228fcb')
}

...
{
  "productId": 1,
  "note": 10,
  "review": "excelente produto",
  "userId": 1
},
{
  "productId": 1,
  "note": 10,
  "review": "Produto de qualidade, atendimento excepcional, custo benefício muito bom, adorei o resultado no carro",
  "userId": 3
},
{
  "productId": 1,
  "note": 9,
  "review": "custo benefício bom",
  "userId": 4
},
{
  "productId": 2,
  "note": 4,
  "review": "atraso na entrega",
  "userId": 5
},
{
  "productId": 10,
  "note": 9,
  "review": "Entrega express é excelente",
  "userId": 6
},
{
  "productId": 3,
  "note": 7,
  "review": "nada mal",
  "userId": 7
},
{
  "productId": 3,
  "note": 5,
  "review": "-",
  "userId": 8
},
{
  "productId": 6,
  "note": 9,
  "review": "muito bom, ganhou um cliente",
  "userId": 9
},
{
  "productId": 8,
  "note": 0,
  "review": "Produto defeituoso, em contato com a loja recebi um atendimento que deixou muito a desejar",
  "userId": 10
}
}

acknowledged: true,
insertedIds: {
  '0': ObjectId('60be9302e15c498dad228fd1'),
  '1': ObjectId('60be9302e15c498dad228fd2'),
  '2': ObjectId('60be9302e15c498dad228fd3'),
  '3': ObjectId('60be9302e15c498dad228fd4'),
  '4': ObjectId('60be9302e15c498dad228fd5')
}

20:48
15/08/2024
```

```
mongosh mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
...
{
  "email": "fernando@gmail.com",
  "name": "Fernando"
},
{
  "email": "sorocaba@gmail.com",
  "name": "Sorocaba"
}
}

acknowledged: true,
insertedIds: {
  '0': ObjectId('60be926ae15c498dad228fb5'),
  '1': ObjectId('60be926ae15c498dad228fb6'),
  '2': ObjectId('60be926ae15c498dad228fb7'),
  '3': ObjectId('60be926ae15c498dad228fb8'),
  '4': ObjectId('60be926ae15c498dad228fb9'),
  '5': ObjectId('60be926ae15c498dad228fba'),
  '6': ObjectId('60be926ae15c498dad228fbb'),
  '7': ObjectId('60be926ae15c498dad228fbc'),
  '8': ObjectId('60be926ae15c498dad228fbd')
}

...
{
  "email": "leandro@hotmail.com",
  "name": "Leandro"
},
{
  "email": "leonardo@terra.com",
  "name": "Leonardo"
},
{
  "email": "henrique@hotmail.com",
  "name": "Henrique"
},
{
  "email": "juliano@uol.com",
  "name": "Juliano"
},
{
  "email": "marilia@gmail.com",
  "name": "Marilia"
},
{
  "email": "bruno@gmail.com",
  "name": "Bruno"
},
{
  "email": "marrone@gmail.com",
  "name": "Marrone"
},
{
  "email": "fernando@gmail.com",
  "name": "Fernando"
},
{
  "email": "sorocaba@gmail.com",
  "name": "Sorocaba"
}
}

acknowledged: true,
insertedIds: {
  '0': ObjectId('60be92dce15c498dad228fbc'),
  '1': ObjectId('60be92dce15c498dad228fbc'),
  '2': ObjectId('60be92dce15c498dad228fbc'),
  '3': ObjectId('60be92dce15c498dad228fbc'),
  '4': ObjectId('60be92dce15c498dad228fbc'),
  '5': ObjectId('60be92dce15c498dad228fbc'),
  '6': ObjectId('60be92dce15c498dad228fbc'),
  '7': ObjectId('60be92dce15c498dad228fbc'),
  '8': ObjectId('60be92dce15c498dad228fbc')
}

...
{
  "productId": 1,
  "quantity": 1,
  "userId": 1
},
{
  "productId": 2,
  "quantity": 2,
  "userId": 2
},
{
  "productId": 10,
  "quantity": 4,
  "userId": 3
}

20:48
15/08/2024
```

```
mongosh mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Microsoft Windows [Version 10.0.19045.4651]
(c) Microsoft Corporation. All rights reserved.

C:\Users\labsfiap>mongosh "mongodb://localhost:27017"
Current Mongosh Log ID: 66be8e56e15c498dad228fb4
Connecting to:  mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.2.15
Using MongoDB:  7.0.12
Using Mongosh:  2.2.15

For mongosh info see: https://docs.mongodb.com/mongosh-shell/

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

-----
The server generated these startup warnings when booting
  2024-08-15T23:22:35.362+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
  2024-08-15T23:22:36.455+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
  2024-08-15T23:22:36.455+00:00: /sys/kernel/mm/transparent_hugepage/enabled is 'always'. We suggest setting it to 'never' in this binary version
  2024-08-15T23:22:36.455+00:00: vm.max_map_count is too low
-----

test> use livemongodb
switched to db livemongodb
livemongodb> db.createCollection("users")
{ ok: 1 }
livemongodb> db.createCollection("product")
{ ok: 1 }
livemongodb> db.createCollection("orders")
{ ok: 1 }
livemongodb> db.createCollection("reviews")
{ ok: 1 }
livemongodb> db.users.insertMany([
...   { name: "Leandro", email: "leandro@hotmail.com" },
...   { name: "Leonardo", email: "leonardo@terra.com" },
...   { name: "Henrique", email: "henrique@hotmail.com" },
...   { name: "Juliano", email: "juliano@uol.com" },
...   { name: "Marilyn", email: "marilia@gmail.com" },
...   { name: "Bruno", email: "bruno@gmail.com" },
...   { name: "Marrone", email: "marrone@gmail.com" },
...   { name: "Fernando", email: "fernando@gmail.com" },
...   { name: "Sorocaba", email: "sorocaba@gmail.com" }
])
```

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\labsfiap> docker --version
Docker version 25.0.4-rc, build c4cd0a9
PS C:\Users\labsfiap> docker pull mongo:latest
latest: Pulling from library/mongo
171382180277: Pull complete
89bdaccc097: Pull complete
d6b691142508: Pull complete
bdc1924dee6d: Pull complete
a91a7990873d: Pull complete
77e5254f6ae8: Pull complete
403f753f5920: Pull complete
08cd53ea307c: Pull complete
Digest: sha256:d67a09266a48faee269e6b5c6b1c7b9d9de3dd8ald5097a0881e15576bbb4
Status: Downloaded newer image for mongo:latest
docker.io/library/mongo:latest
PS C:\Users\labsfiap> docker run --name mongodb-container -d -p 27017:27017 mongo:latest
d1be64091603cc28036c4a36db2c40f0fe90ba1ce0b1b4a91a727303ebe860
PS C:\Users\labsfiap>
```

### 3. Documentação e Evidências

Todas as etapas foram capturadas e documentadas para comprovar o funcionamento do pipeline.

### Conclusão

O pipeline foi implementado utilizando Google Cloud Composer com Apache Airflow para orquestrar a coleta, transformação e carga de dados climáticos. As três DAGs (weather\_data\_pipeline\_dag, etl\_raw\_to\_trusted\_weather, e

etl\_trusted\_to\_refined\_weather) garantem que os dados sejam coletados, transformados e enriquecidos de forma escalável e automatizada. As descrições dos campos na camada REFINED ajudam a documentar e entender o conjunto de dados final para consumo analítico.

Este documento serve como uma referência completa das ferramentas e processos utilizados, garantindo uma visão clara do fluxo de dados desde a coleta até o refinamento final para uso analítico.