Final Project DE

This final project is about Dockerize ETL Pipeline using ETL tools Airflow that extract Public API data from PIKOBAR, then load into MySQL (Staging Area) and finally aggregate the data and save into PostgreSQL.

A. ETL Architecture Diagram and Integration Design Diagram

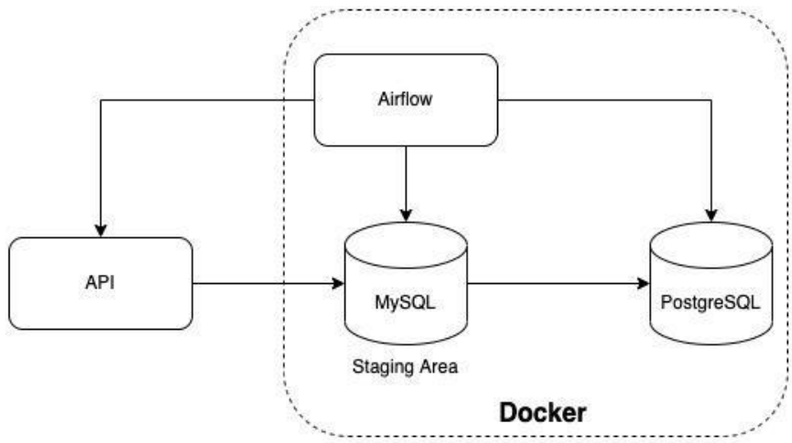

Figure 1. ETL Architecture Diagram

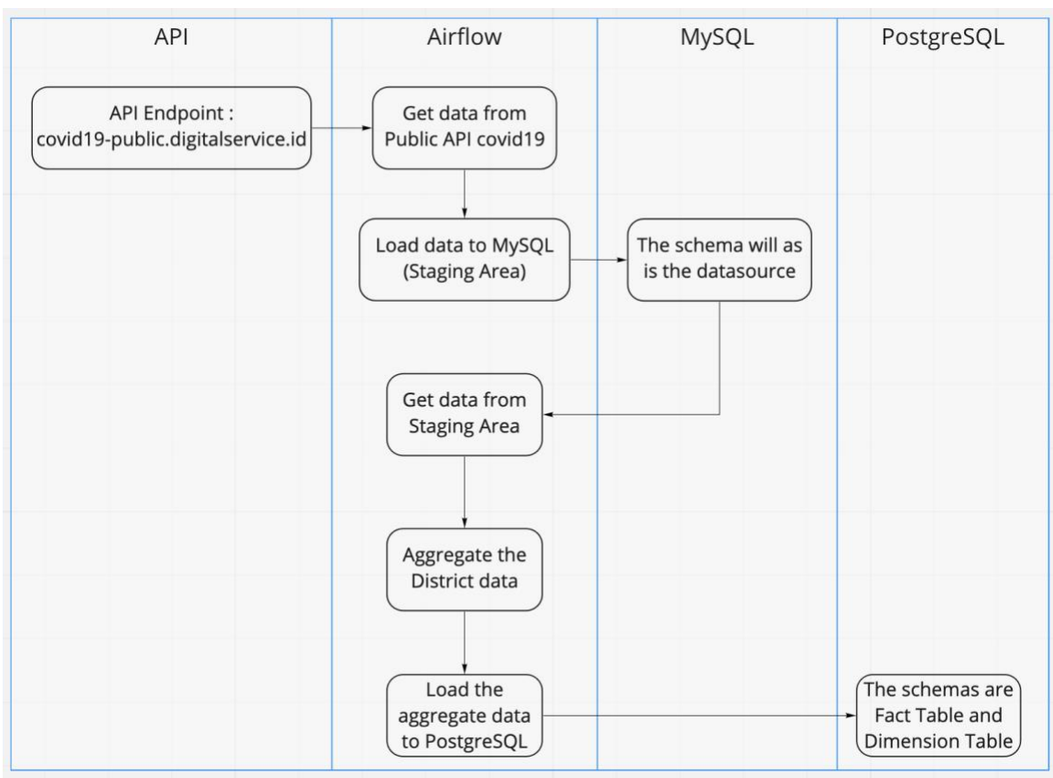

Figure 2. Integration Design Diagram

B. API Specifications

API Endpoint	covid19-public.digitalservice.id
Path	/api/v1/rekapitulasi_v2/jabar/harian
Content-Type	application/json
Request Parameter	level
Request method	GET

C. API Request Example

```
curl -X GET "https://covid19-public.digitalservice.id/api/v1/rekapitulasi_v2/jabar/harian?level=kab"
-H "accept: application/json"
```

D. API Response Example


```
{
  "status_code": 200,
  "data": {
    "metadata": {
      "last_update": null
    },
    "content": [
      {
        "tanggal": "2020-08-05",
        "kode_prov": "32",
        "nama_prov": "Jawa Barat",
        "kode_kab": "3204",
        "nama_kab": "Kabupaten Bandung",
        "SUSPECT": 2210,
        "CLOSECONTACT": 274,
        "PROBABLE": 26,
        "suspect_diisolasi": 31,
        "suspect_discarded": 2179,
        "closecontact_dikarantina": 0,
        "closecontact_discarded": 274,
        "probable_diisolasi": 0,
        "probable_discarded": 0,
        "CONFIRMATION": 0,
        "confirmation_sembuh": 0,
        "confirmation_meninggal": 0,
        "suspect_meninggal": 0,
        "closecontact_meninggal": 0,
        "probable_meninggal": 26
      }
    ]
  }
}
```


E. Project Steps

1. Create Docker (MySQL, Airflow and PostgreSQL) in your local computer
2. Create Database in MySQL and PostgreSQL
3. Create Connection from Airflow to MySQL and PostgreSQL
4. Create DAG
5. Create DDL in MySQL
6. Get data from Public API covid19 and load data to MySQL
7. Create DDL in PostgreSQL for Fact table and Dimension table
8. Create aggregate Province Daily save to Province Daily Table
9. Create aggregate Province Monthly save to Province Monthly Table
10. Create aggregate Province Yearly save to Province Yearly
11. Create aggregate District Monthly save to District Monthly
12. Create aggregate District Yearly save to District Yearly
13. Schedule the DAG daily save to Province Monthly

F. Table Specification

Dimension table

1. Province table
    a. province_id
    b. province_name

2. District table
    a. district_id
    b. province_id
    c. district_name

3. Case table
    a. Id
    b. Status name (suspect, closecontact, probable, confirmation)
    c. Status detail

Fact table

1. Province Daily Table
    a. Id (auto generate)
    b. province_id
    c. case_id
    d. date
    e. total

2. Province Monthly Table
    a. Id (auto generate)
    b. province_id
    c. case_id
    d. month
    e. total

3. Province Yearly Table
    a. Id (auto generate)
    b. province_id
    c. case_id
    d. year
    e. total

4. District Monthly Table
    a. Id (auto generate)
    b. district_id
    c. case_id
    d. month
    e. total

5. District Yearly Table
    a. Id (auto generate)
    b. district_id
    c. case_id
    d. year
    e. total