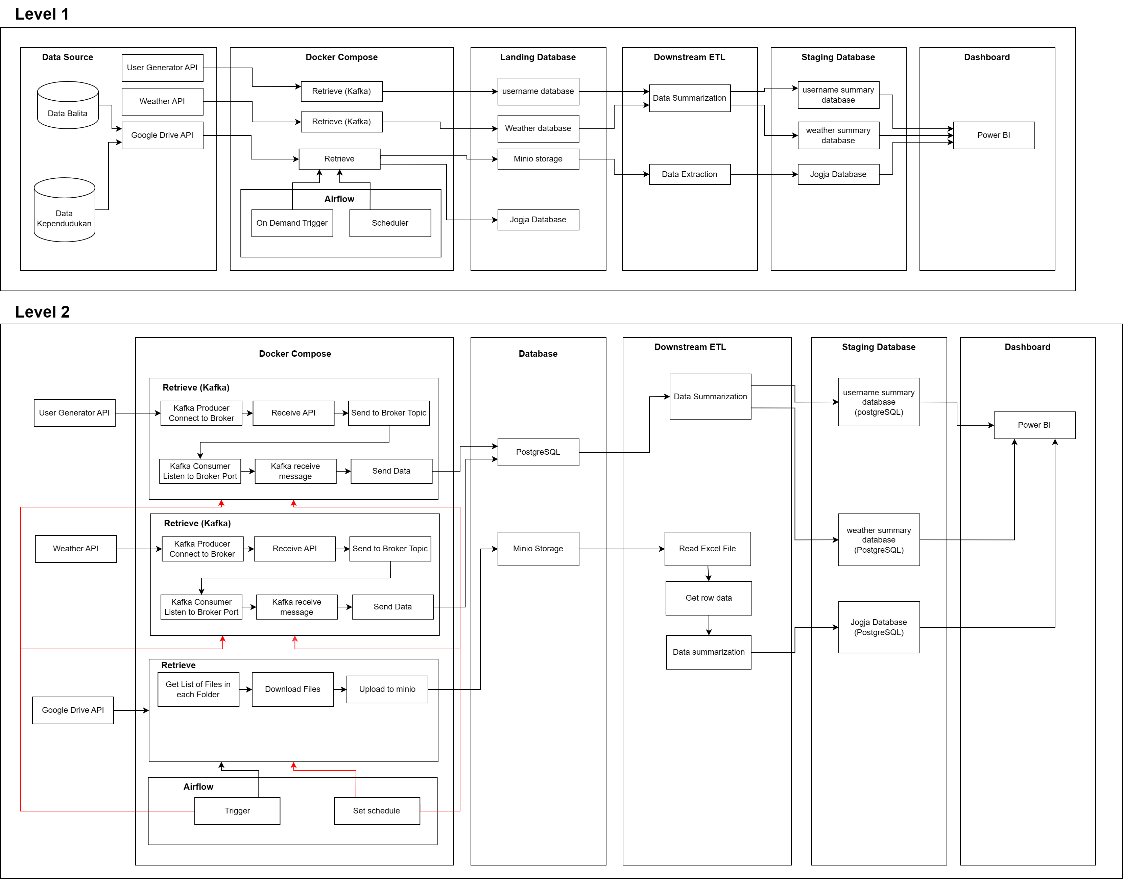
# Big Data Architecture Data Engineering

### Arsitektur Diagram

#### Blok Diagram Arsitektur

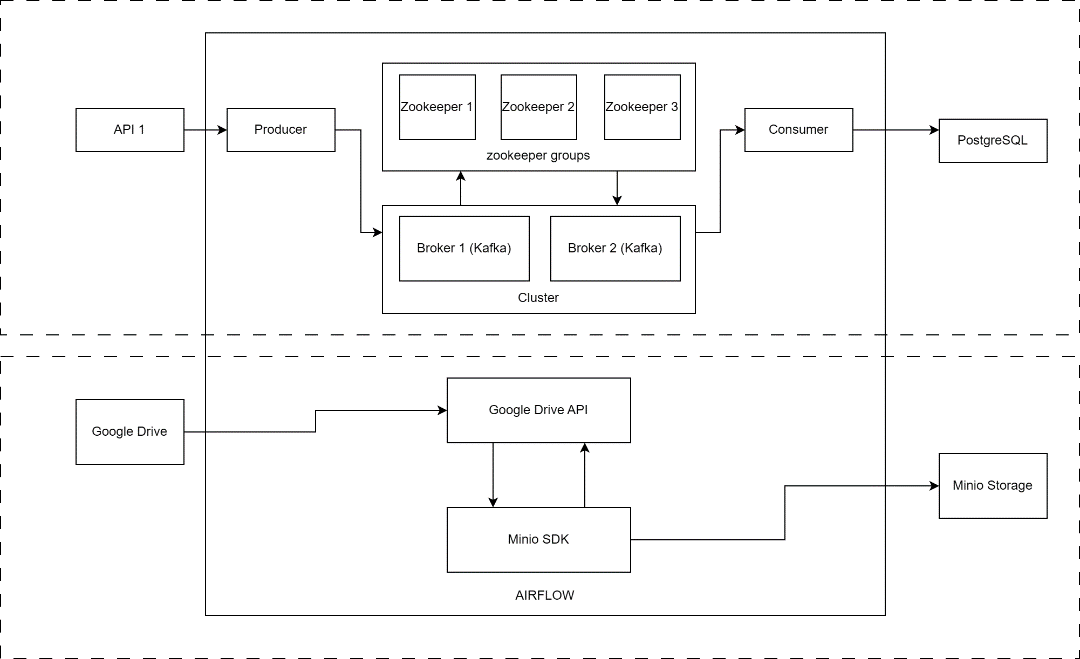


Alur pengolahan data dari awal hingga akhir:

1. API: API yang digunakan adalah Random User generator dan Weather API. Dua API ini digunakan untuk mensimulasikan data streaming processing menggunakan Kafka dan Airflow.
2. Google Drive: Drive digunakan sebagai storage untuk menyimpan data berupa files (csv dan excel) yang ditransfer secara berkala oleh Airflow menuju minio storage. File-file tersebut kemudian diekstrak dan disimpan dalam PostgreSQL.
3. Kafka: Kafka digunakan sebagai tools data streaming processing untuk menarik data dari API weather dan random user generator. Hasil penarikan dari kedua API disimpan dalam database staging dalam PostgreSQL. Kafka digunakan bersama Zookeeper untuk menjaga availability dari para broker untuk mengirim data dari producer menuju consumer.
4. Airflow: Airflow digunakan sebagai alat scheduler dan orchestration dari task-task yang telah didefinisikan pada batch processing dan stream processing.
5. Power BI: Dashboard Power BI digunakan untuk menampilkan analisis data hasil pengolahan dari API dan file excel dan csv.

#### Skema Pengolahan Data

Skema pengiriman dari data source menuju staging database ditampilkan sebagai berikut.



Hasil dari staging database kemudian diolah sebelum ditampilkan pada dashboard Power BI.

### Dokumentasi Kode Program

#### Direktori Repositori

**-Dags**

**--batch\_unstructured**

**---creds**

----Credentials.json

**---py**

----airflow.py

----google\_api.py

----minio\_api.py

----models.py

----process\_csv\_files.py

----queries.py

--- .env

**--jogja\_api**

--- .env

--- apis.py

--- kafka\_admin.py

--- kafka\_consume\_stream.py

--- kafka\_produce\_stream.py

--- models.py

--- transform.py

-.env

-.gitignore

-docker-compose.yml

-Jogja Big Data.pbix

-requirements.txt

#### Penjelasan File dalam Direktori

1. airflow.py: digunakan untuk orkestrasi batch processing menggunakan airflow, drive, dan minio.
2. google\_api.py: Segala kebutuhan i.e koneksi, download file, upload file, dan menampilkan file-file dalam Google Drive.
3. minio\_api.py: Segala kebutuhan i.e koneksi, download file, upload file, dan menampilkan file-file dalam Minio.
4. models.py: Digunakan untuk mendefinisikan schema tabular dari beberapa tabel yang digunakan dalam processing.
5. process\_csv\_files.py: Digunakan untuk memproses konten file-file (csv dan Excel) dan menyimpan data tsb di dalam PostgreSQL.
6. queries.py: Digunakan untuk melakukan processing penyimpanan properties dari file-file logging ketika mengirim data dari Google Drive menuju Minio.
7. apis.py: Segala kebutuhan terkait konektivitas, pendefinisian schema tabular, dan sejenisnya dari API dengan menggunakan SQLAlchemy, requests, dan Python.
8. Kafka\_admin.py: Digunakan untuk orkestrasi stream processing menggunakan Kafka dan Airflow.
9. Kafka\_consume\_stream.py: Digunakan untuk *consume* data dari Broker menuju PostgreSQL.
10. Kafka\_produce\_stream.py: Digunakan untuk menarik data dari API pada apis.py menuju producer Kafka melalui Broker.
11. Transform.py: Digunakan sebagai downstream ETL dari database postgreSQL sebelum ditampilkan pada Dashboard Power BI.

### User Manual

#### Instalasi

##### Konfigurasi Credentials pada Google Cloud Platform

1. URL: [Create access credentials  |  Google Workspace  |  Google for Developers](https://developers.google.com/workspace/guides/create-credentials).
2. Observe bagian Service Account Credentials.

A screenshot of a computer screen

Description automatically generated

1. Klik “Go to Service Accounts”.

A screenshot of a computer

Description automatically generated

1. Klik Create Service Account dengan konektivitas menuju Google Drive.
2. Setelah Service Account dibuat, share email service account dengan Drive pemiliki akun. Pada kasus ini, akun yang digunakan adalah [jogjabigdata@gmail.com](mailto:jogjabigdata@gmail.com).

A screenshot of a computer

Description automatically generated

##### Konfigurasi Docker Compose dan Clone dari Repositori

1. Clone repositori github.
2. Download Docker Desktop agar dapat memonitor docker container yang di-run.
3. Redirect menuju lokasi project.
4. Ketik “docker compose up -d” pada command prompt di direktori project.
5. Tunggu hingga selesai 😊.
6. Lanjutkan ke tahap penggunaan.
7. Note: dalam beberapa kali biasanya ada notif image webserver error, itu biarkan saja karena docker sedang dalam proses instalasi modul-modul.

#### Penggunaan

##### Console Airflow

URL: <http://localhost:8080/>

1. Pada console, terdapat list DAG yang telah terinstall dan ter-import.
2. Trigger untuk melihat data yang dihasilkan.

##### Console Kafka

<http://localhost:9021/>

##### Console Minio

<http://172.25.157.89:9001/>