

Final Project

Kelompok 7

"Hiduplah seolah engkau mati besok. Belajarlah seolah engkau hidup selamanya."

Dibimbing X Kampus Merdeka

MEET OUR TEAM



Animni Fiddaroini

UNIVERSITAS
INDRAPRASTA
PGRI



Alya Nabila

UNIVERSITAS
DIPONEGORO



Fathur Rahman S.

UNIVERSITAS
NEGERI
YOGYAKARTA



Gloria W.Z.

UNIVERSITAS
BINA SARANA
INFORMATIKA



Imas Siti M.

UNIVERSITAS
NAHDLATUL ULAMA
YOGYAKARTA

TABLE OF CONTENT

1

Introduction : Team

2

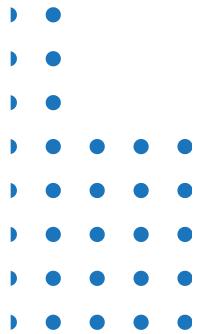
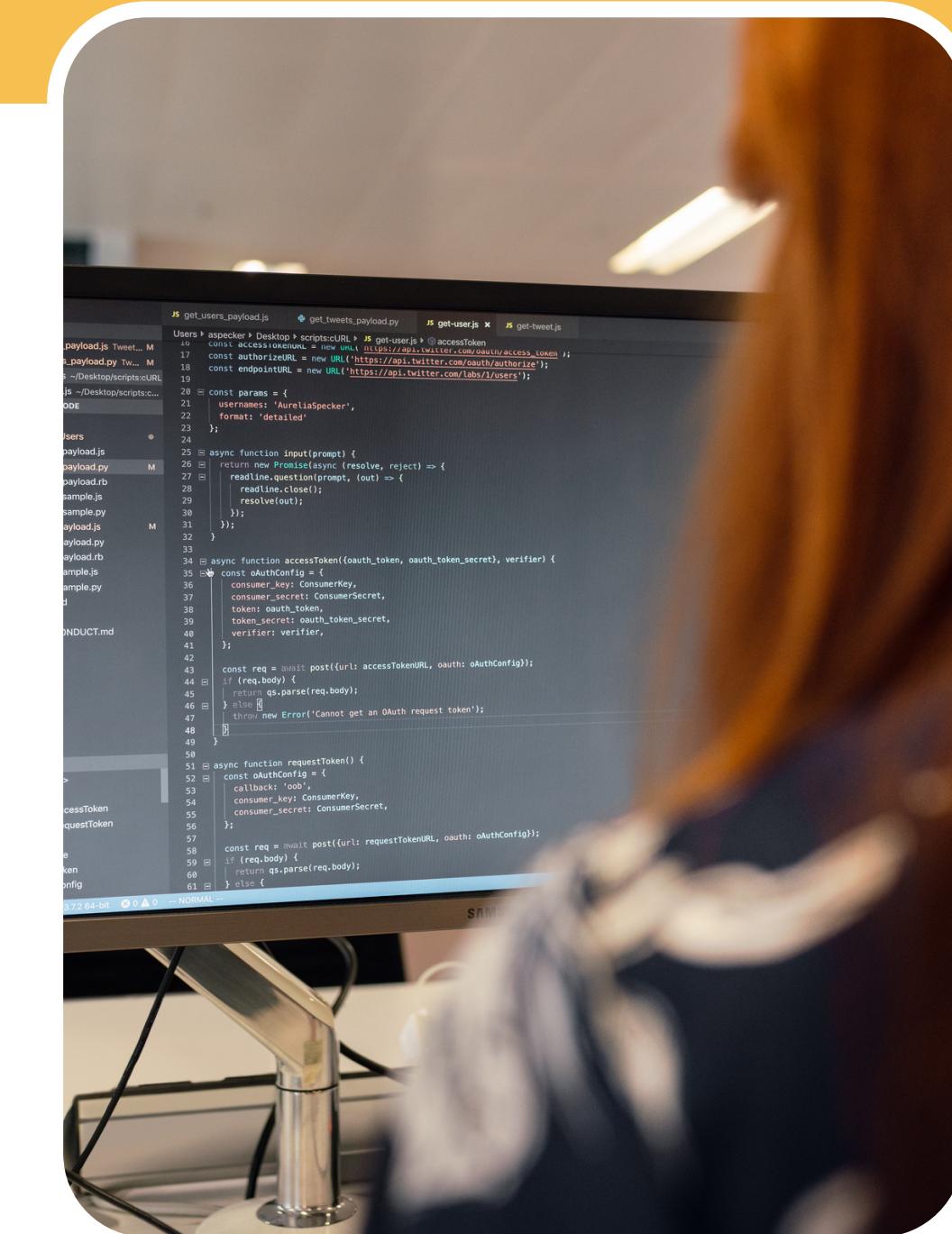
Latar Belakang

3

Tasks

4

Dokumentasi



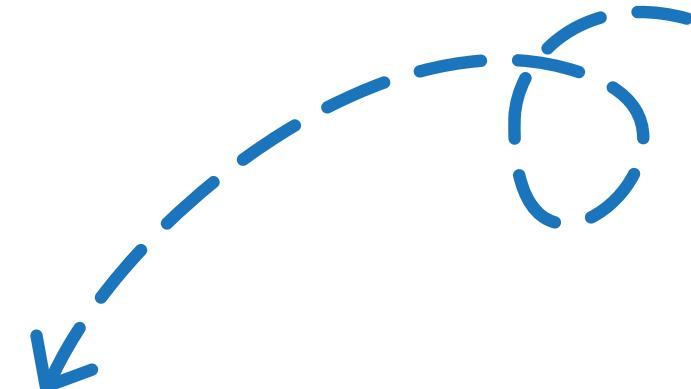
LATAR BELAKANG PROJECT

Data Engineer adalah orang yang ahli dalam bidang TI untuk mengolah hingga mengumpulkan berbagai macam data yang diperlukan oleh perusahaan dalam berbagai bidang. Data Engineer berperan besar dalam menyimpan, menganalisis dan mengolah data bahkan dalam jumlah besar di hampir seluruh bidang industri pekerjaan.

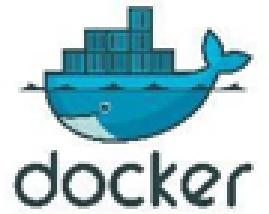
Sebagai seorang data engineer, kita dipekerjakan di perusahaan ritel online AS yang menjual produk pelanggan umum langsung ke pelanggan dari berbagai pemasok di seluruh dunia. Kita membangun infrastruktur data menggunakan data yang dihasilkan yang dibuat untuk mencerminkan data dunia nyata dari perusahaan teknologi terkemuka.



Tasks

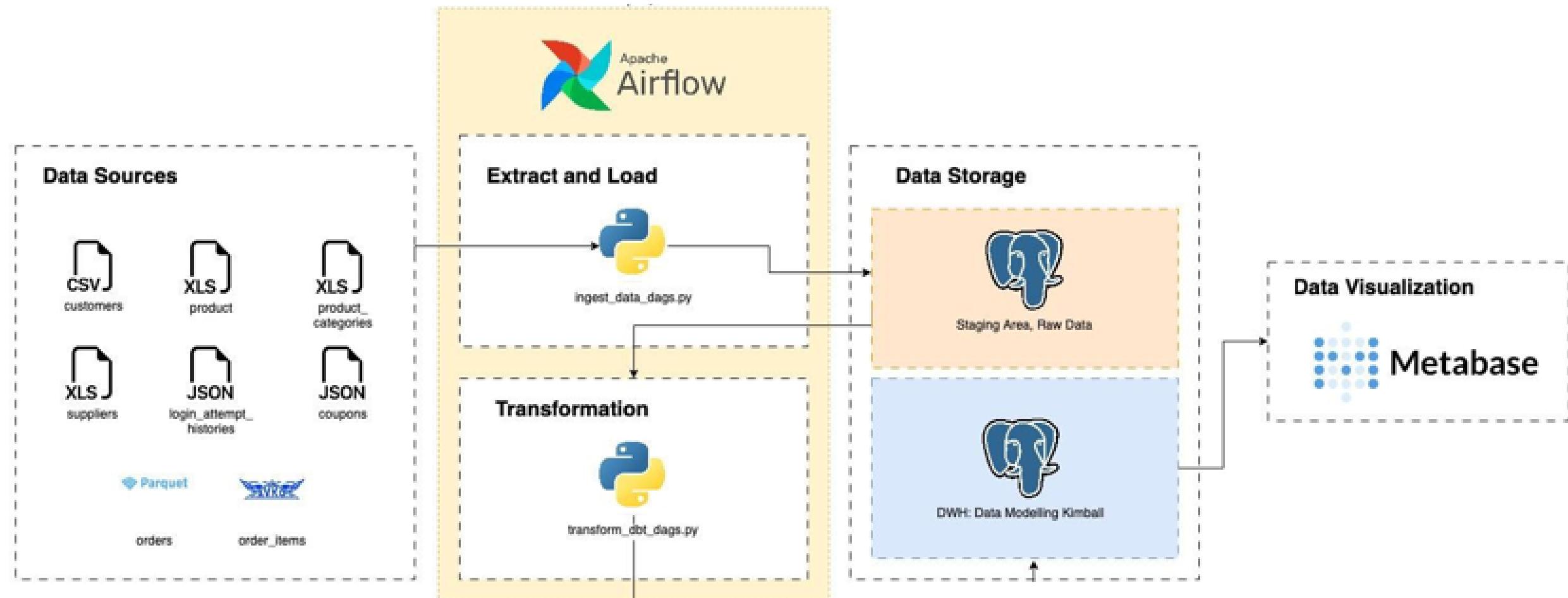
- 
- ✓ ETL/ELT Job Creation using Airflow
 - ✓ Data Modeling in Postgres
 - ✓ Dashboard Creation with Data Visualization
 - ✓ Craft a Presentation Based on Your Work

Data Platform Architecture

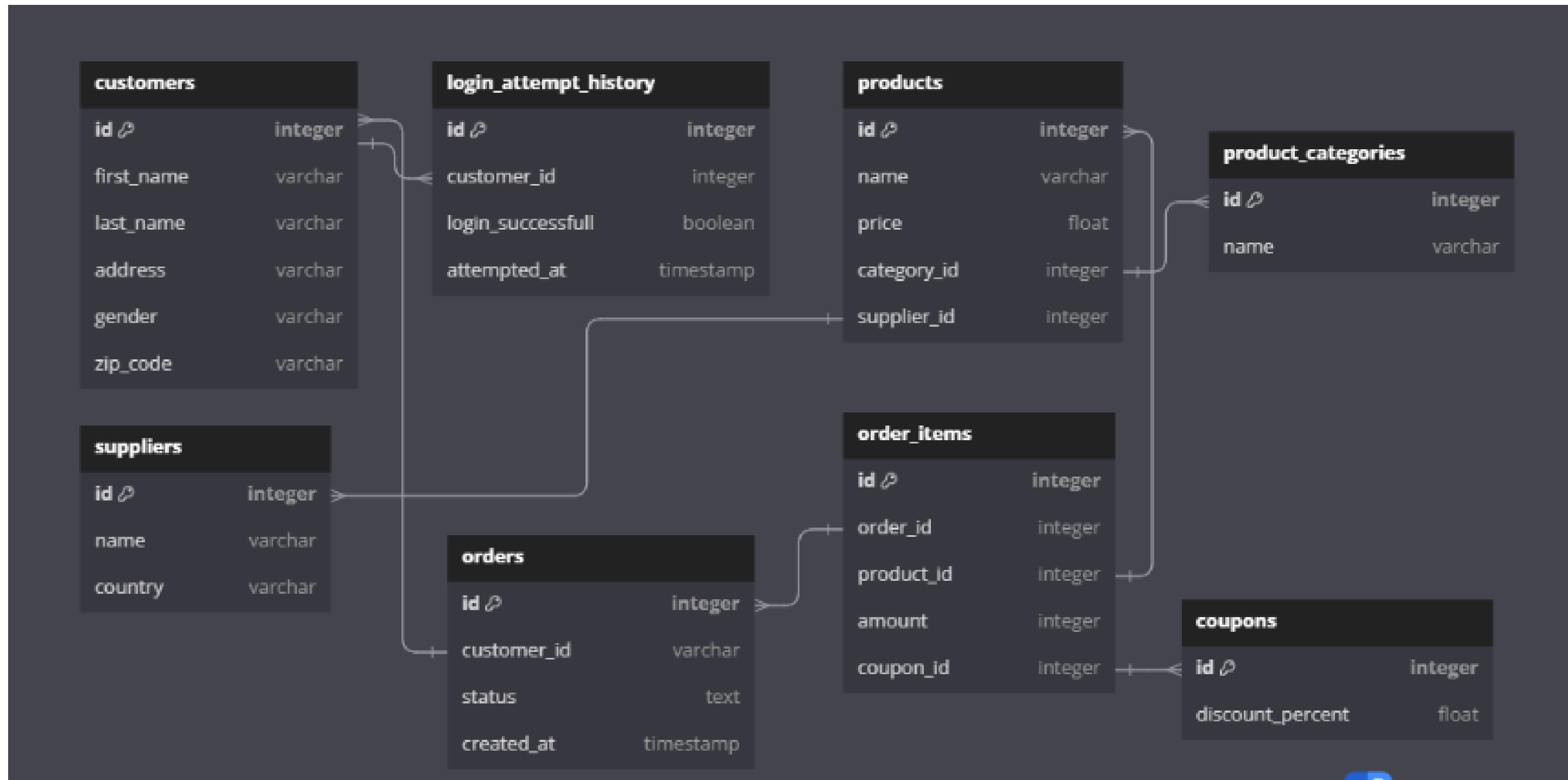


Tech Stack

- Docker
- Python
- Airflow
- Postgresql
- Metabase

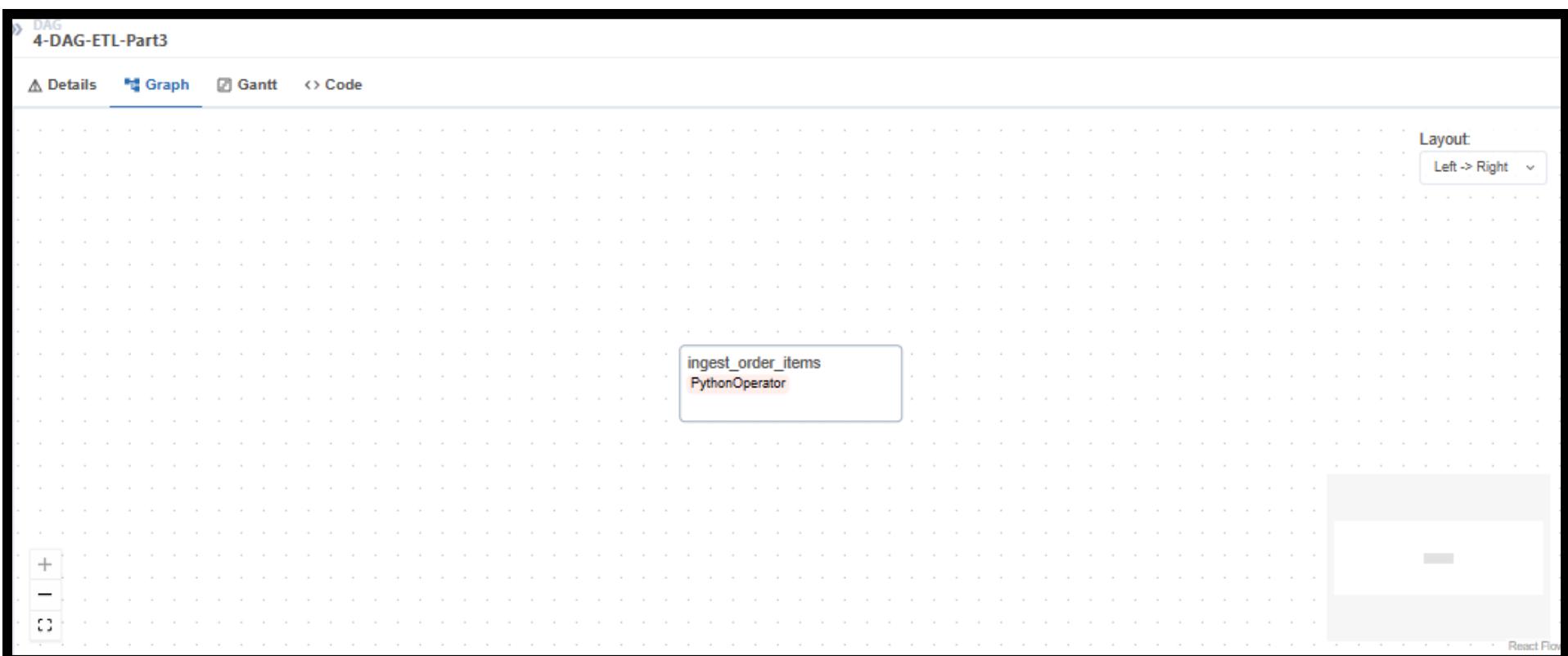
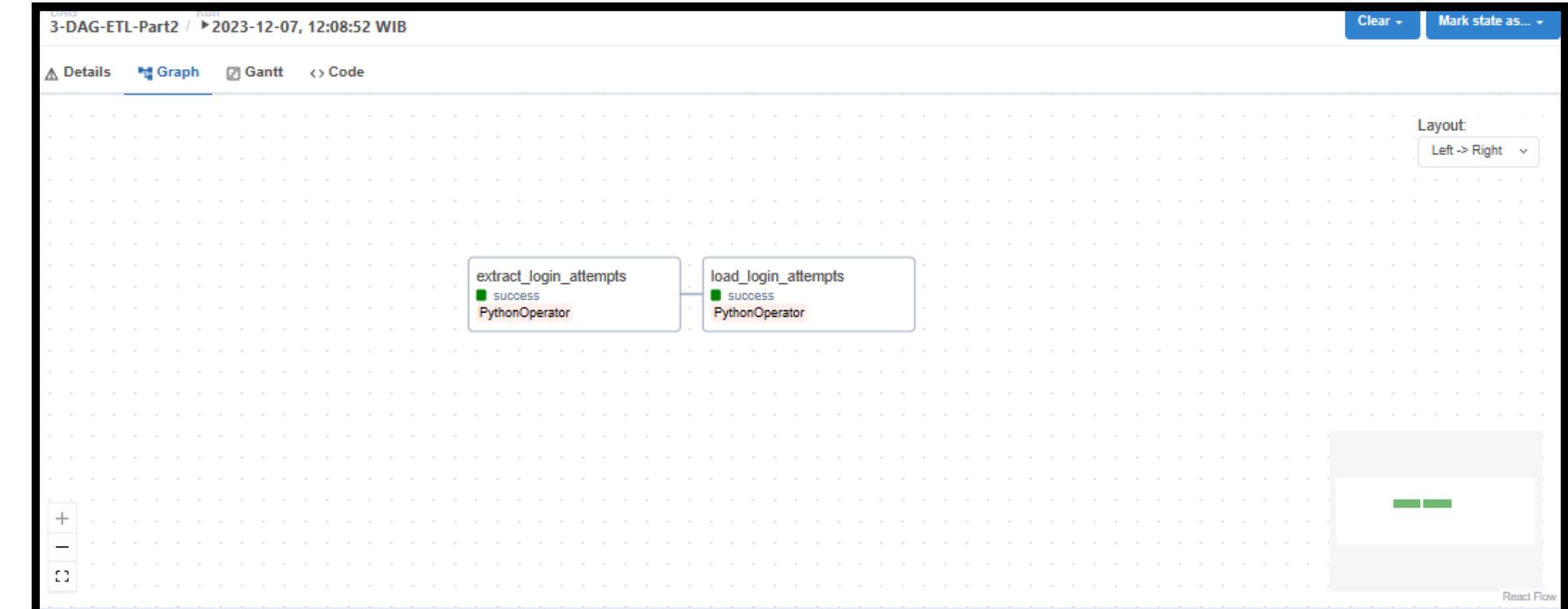
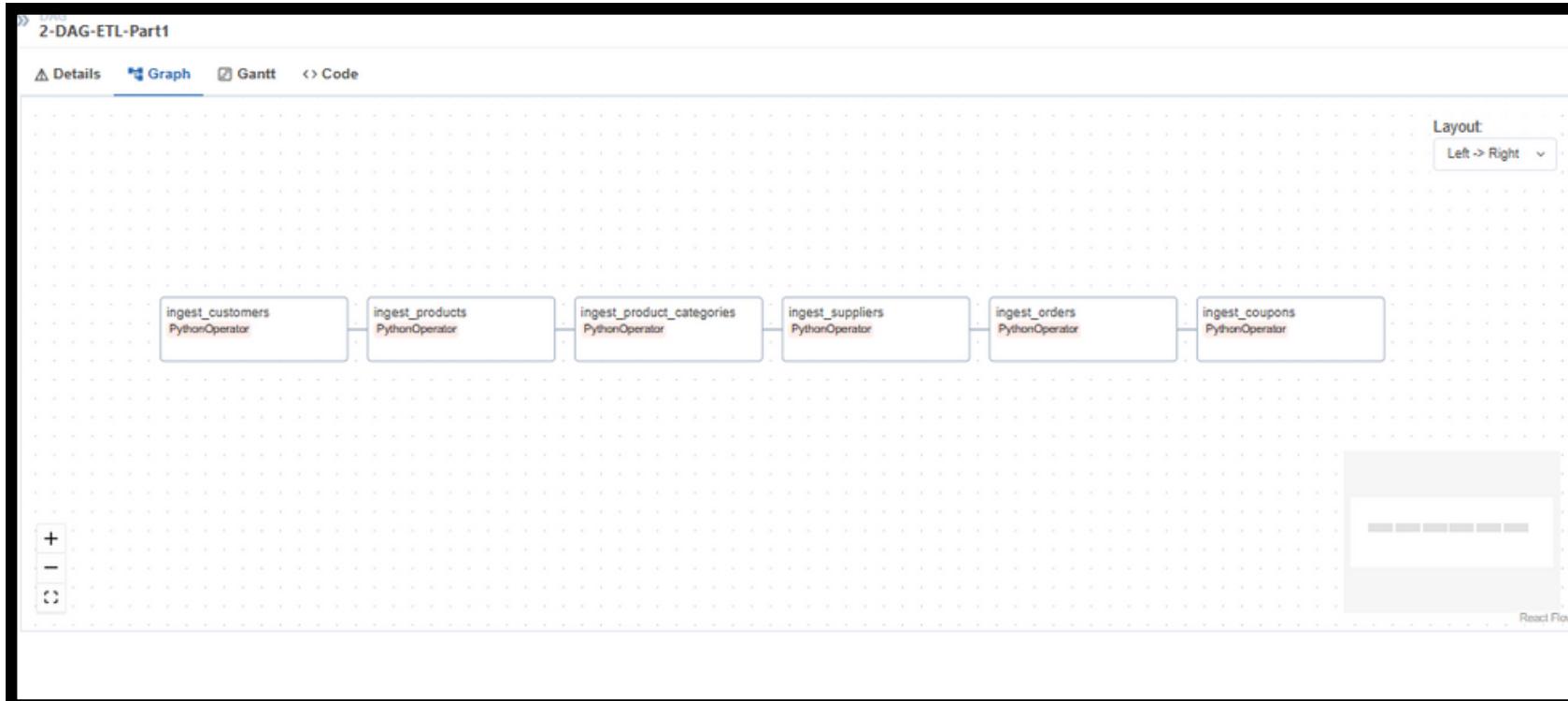


Data source



DOKUMENTASI

ETL Job Creation using Airflow



DOKUMENTASI

Ingest Data

The image shows two side-by-side code editors, likely from the VS Code IDE, displaying Python code for an Airflow DAG named '2-dag-ETL.py'. Both editors have identical layouts and content.

Left Editor Content:

```
1  from airflow import DAG
2  from datetime import datetime, timedelta
3  from airflow.operators.python_operator import PythonOperator
4  from airflow.hooks.postgres_hook import PostgresHook
5  import pandas as pd
6  from sqlalchemy import create_engine, types
7  from sqlalchemy import Integer, String, Float, DateTime
8  import json
9  import fastavro
10
11
12 default_args = {
13     'owner': 'airflow',
14     'start_date': datetime(2022, 11, 12),
15 }
16
17 # Step 3: Creating DAG Object
18 dag = DAG(dag_id='DAG-ingest5',
19            default_args=default_args,
20            schedule_interval='@once',
21            catchup=False
22            )
```

Right Editor Content:

```
171 ingest_products = PythonOperator(
172     task_id='ingest_products',
173     python_callable=ingest_products,
174     dag=dag)
175
176 ingest_product_categories = PythonOperator(
177     task_id='ingest_product_categories',
178     python_callable=ingest_product_categories,
179     dag=dag)
180
181 ingest_suppliers = PythonOperator(
182     task_id='ingest_suppliers',
183     python_callable=ingest_suppliers,
184     dag=dag)
185
186 ingest_orders = PythonOperator(
187     task_id='ingest_orders',
188     python_callable=ingest_orders,
189     dag=dag)
190
191 ingest_order_items = PythonOperator(
192     task_id='ingest_order_items',
193     python_callable=ingest_order_items,
```

Both editors include a dark sidebar with various icons for file operations, search, and other development tools. The status bar at the bottom of each editor shows the following information: 'Ln 1, Col 1', 'Spaces: 4', 'UTF-8', 'LF', '{ Python }', and a bell icon.

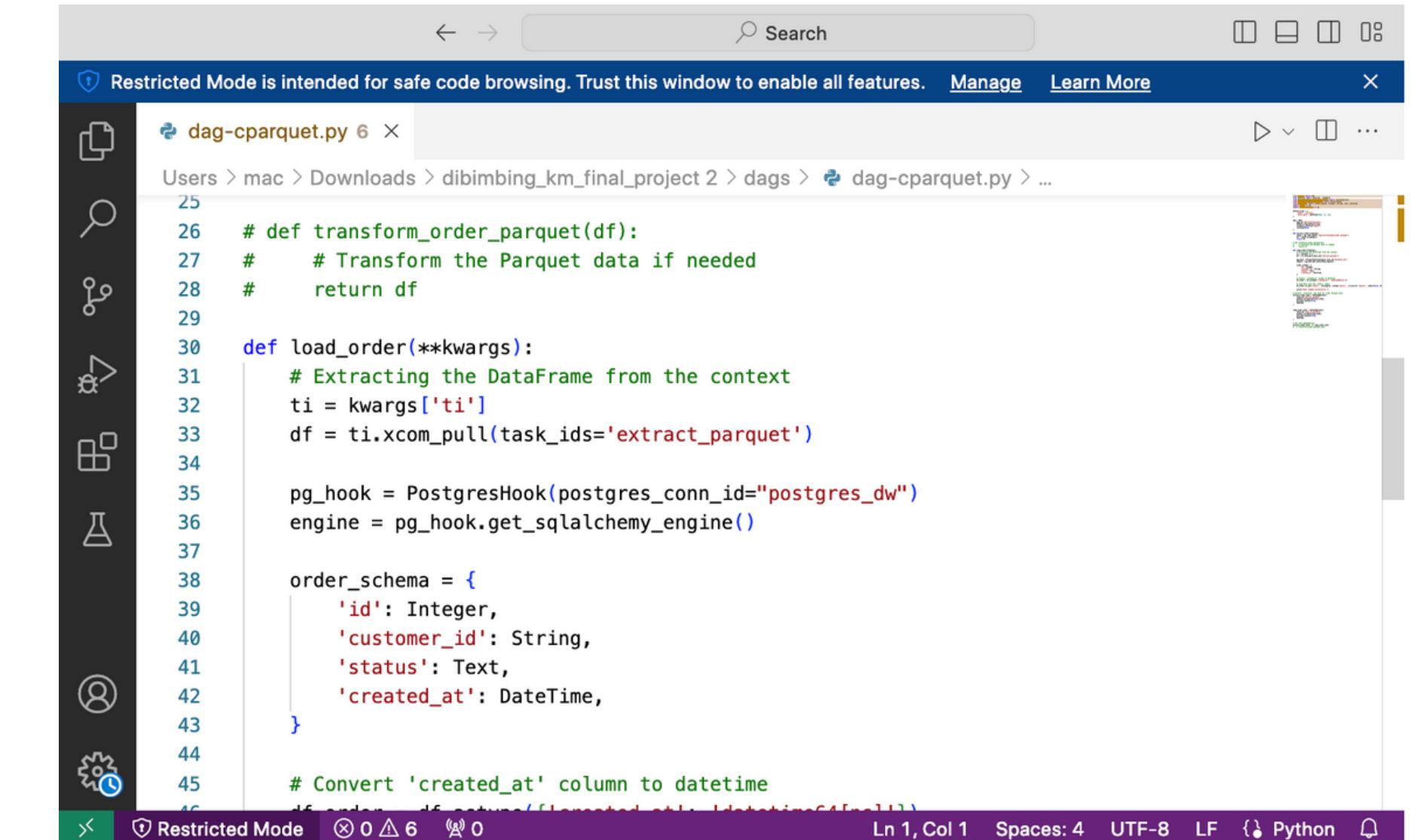
DOKUMENTASI

Transform

- Menghapus Kolom berlebih di tabel suppliers, customers, dan product_categoeris
- Menggabungkan file customers yang terpisah
- Menyesuaikan bentuk dari data json dan avro menjadi dataframe
- Mengubah tipe data sesuai dengan rancangan ERD

Load

- Memuat data yang diolah kedalam data warehouse



The screenshot shows a code editor window with a dark theme. On the left is a vertical toolbar with icons for file operations like Open, Save, Find, and Refresh. The main area displays a Python script titled 'dag-cparquet.py'. The script contains several functions and imports:

```
25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46
```

```
# def transform_order_parquet(df):
#     # Transform the Parquet data if needed
#     return df

def load_order(**kwargs):
    # Extracting the DataFrame from the context
    ti = kwargs['ti']
    df = ti.xcom_pull(task_ids='extract_parquet')

    pg_hook = PostgresHook(postgres_conn_id="postgres_dw")
    engine = pg_hook.get_sqlalchemy_engine()

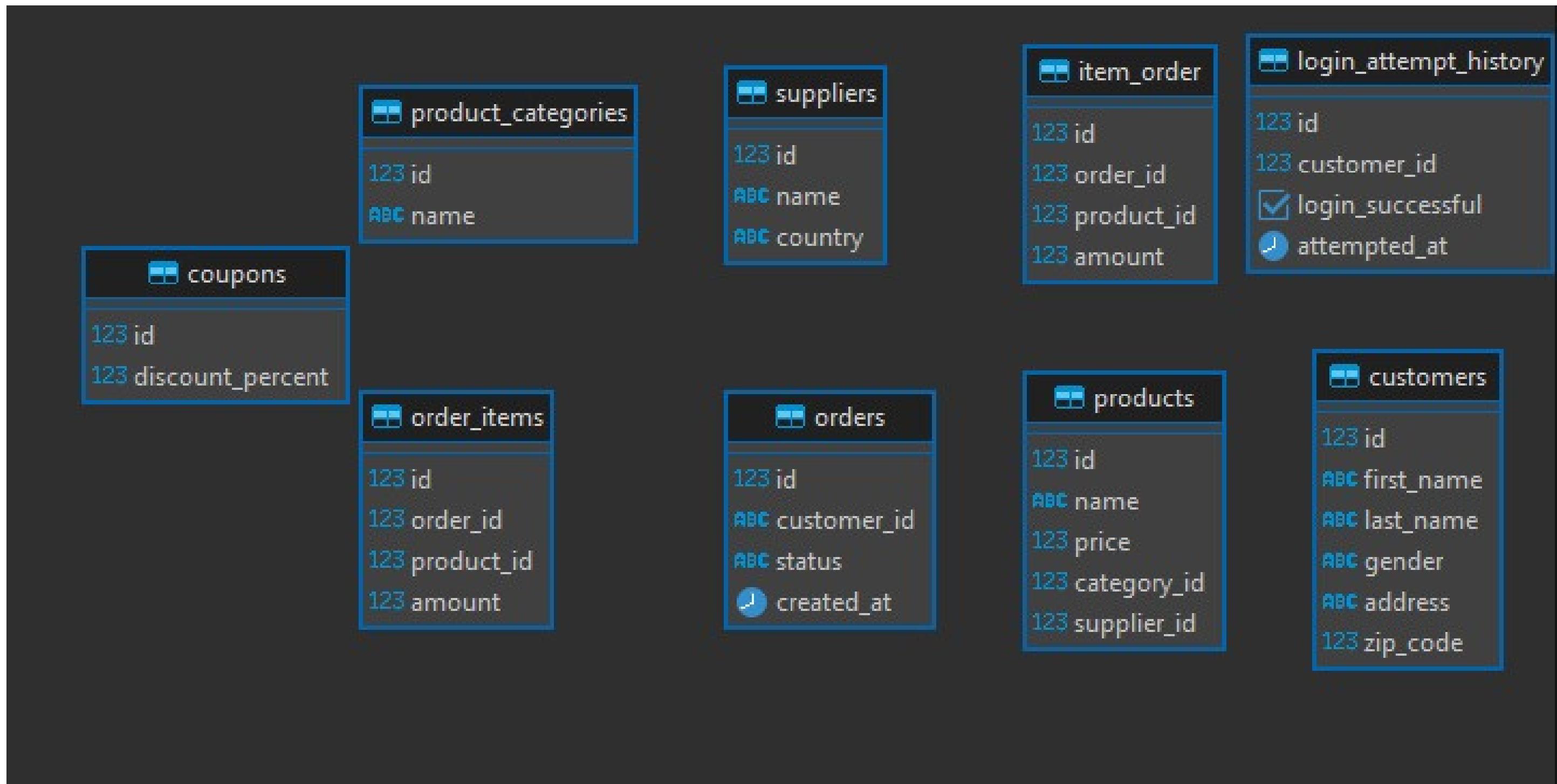
    order_schema = {
        'id': Integer,
        'customer_id': String,
        'status': Text,
        'created_at': DateTime,
    }

    # Convert 'created_at' column to datetime
    df['created_at'] = df['created_at'].dt.datetime.strptime(df['created_at'], '%Y-%m-%d %H:%M:%S').dt.tz_localize(None)
```

The status bar at the bottom indicates 'Restricted Mode' is active, along with other standard status information.

DOKUMENTASI

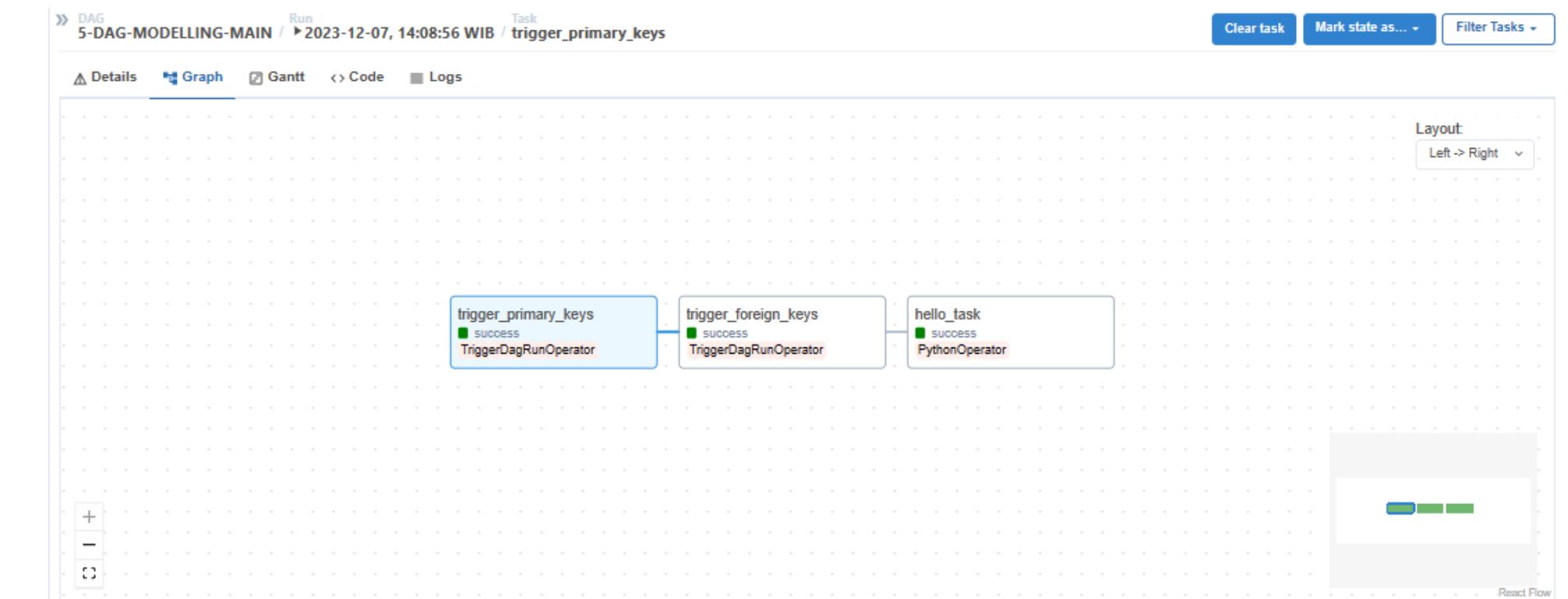
Default ETL Sebelum Modeling



DOKUMENTASI

Data Modelling

- Menentukan Primary keys dari masing-masing tabel
- Mengatur relasi tabel sesuai dengan ERD yang direncanakan



DOKUMENTASI

Data Modelling

Membuat dua tabel baru:

- sales_per_category yang didapat dengan menghitung total penjualan per kategori dengan cara: amount dikalikan price. Data diambil dari tabel order_items dan products.
- return_rate_per_category yang didapat dengan merata-rata total jumlah dari setiap kategori. Tabel ini dibuat dengan tujuan untuk mengetahui kategori mana yang memiliki pengembalian tinggi, dan mengetahui faktor apa yang menyebabkannya.

```
19 # Task untuk membuat tabel sales_per_category
20 create_sales_per_category_table = PostgresOperator(
21     task_id='create_sales_per_category_table',
22     sql="""
23     CREATE TABLE IF NOT EXISTS sales_per_category AS
24     SELECT
25         p.category_id,
26         c.name AS category,
27         SUM(oi.amount * p.price) AS total_sales
28     FROM order_items oi
29     INNER JOIN products p ON oi.product_id = p.id
30     INNER JOIN product_categories c ON c.id = p.category_id
31     GROUP BY p.category_id, c.name;
32     """,
33     postgres_conn_id='postgres_dw',
34     autocommit=True,
35     dag=dag,
36 )
```

DOKUMENTASI

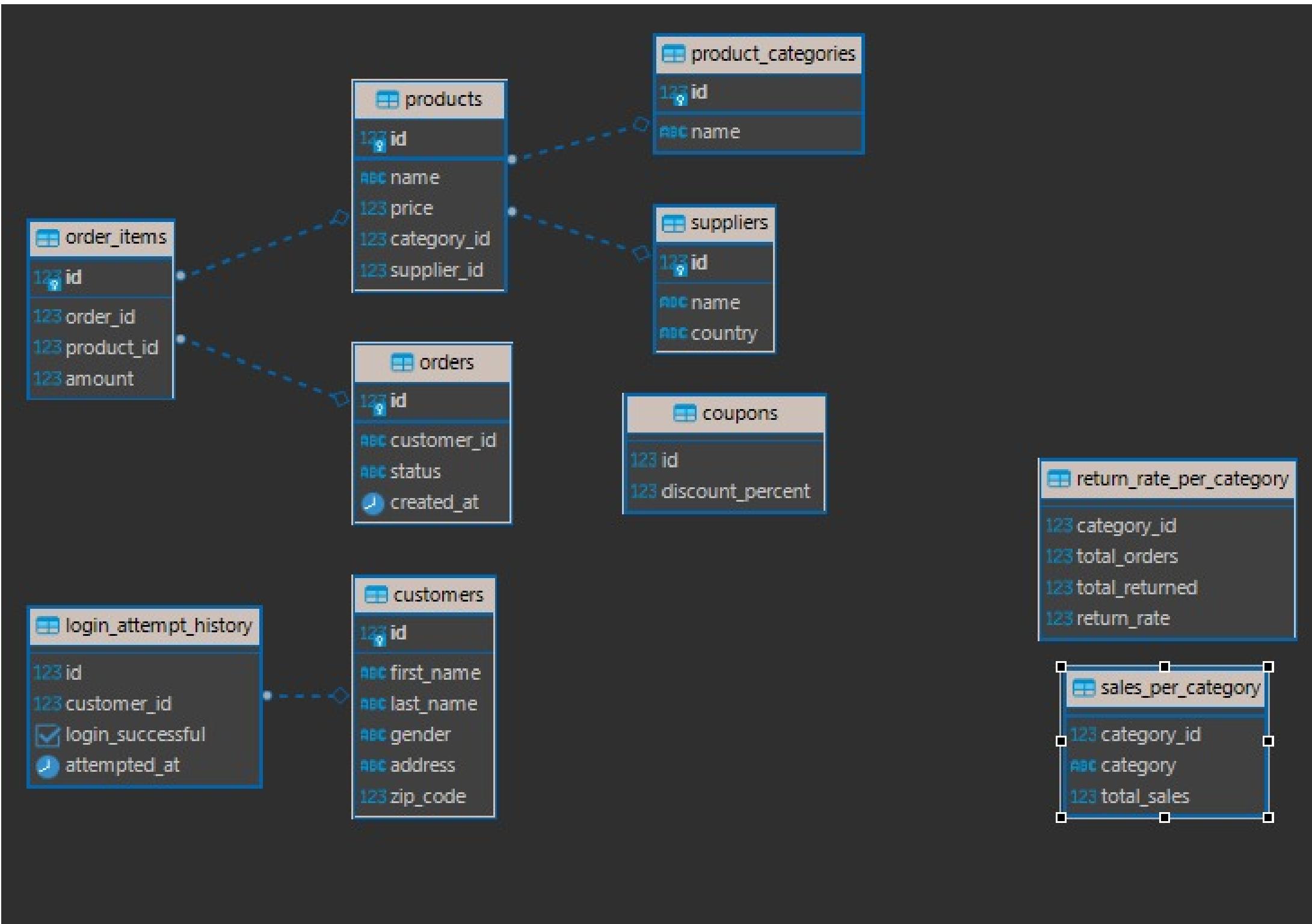
Perubahan yang dilakukan

```
requirements.txt
1  pandas
2  fastavro
3  pyarrow
4  xlrd
5  cramjam
6  avro
7  python-snappy
8  copy
```

Menginstall library yang di pakai

DOKUMENTASI

Data Modeling in Postgres



DOKUMENTASI

Data Preview at PostgreSQL

Grid	123 id	ABC first_name	ABC last_name	ABC address	ABC gender	ABC zip_code
Text	1	Katelyn	Hernandez	354 Brian Turnpike	F	3981
Text	2	Timothy	Reynolds	854 James Dale Apt. 608	M	83703
Text	3	Michelle	Mcintyre	0595 Pierce Orchard	F	73023
Text	4	Gary	Hoover	49129 Ward Track Suite 126	M	57131
Text	5	Laura	Mclaughlin	7825 Barker Fall	F	5371
Text	6	Christopher	Pittman	396 Henson Bypass Suite 507	M	84525
Text	7	Katelyn	Hernandez	354 Brian Turnpike	F	3981
Text	8	Christopher	Hunt	897 Michael Bypass Apt. 361	M	66432
Text	9	Ann	Hobbs	97867 Kathryn Terrace Suite 048	F	83615
Text	10	Andrew	Davis	61843 Brennan Corner	M	29258
Text	11	Melanie	Moore	4351 Valentine Port	F	5079
Text	12	Timothy	Reynolds	854 James Dale Apt. 608	M	83703
Text	13	Michelle	Mcintyre	0595 Pierce Orchard	F	73023
Text	14	Gary	Hoover	49129 Ward Track Suite 126	M	57131
Text	15	Laura	Mclaughlin	7825 Barker Fall	F	5371
Text	16	Timothy	Harrell	184 Green Cliffs Suite 167	M	97711
Text	17	Amy	Logan	164 Carla Freeway	F	51744
Text	18	Luke	Gilbert	76203 Emily Manors	M	82387
Text	19	Melissa	Wright	030 Johnson Cliff Suite 144	F	37289
Text	20	Phillip	Lopez	60130 Mary Freeway Apt. 467	M	80912
Text	21	Carol	Pittman	396 Henson Bypass Suite 507	F	84525
Text	22	Christopher	Hunt	897 Michael Bypass Apt. 361	M	66432
Text	23	Ann	Hobbs	97867 Kathryn Terrace Suite 048	F	83615
Text	24	Andrew	Davis	61843 Brennan Corner	M	29258

customers

Grid	123 id	ABC name	123 price	123 category_id	123 supplier_id
Text	1	3,961 Steel Chair Licensed	5	1	75
Text	2	5,616 Concrete Chips Used	25	6	30
Text	3	893 Cotton Mouse Ergonomic	25	3	42
Text	4	3,409 Frozen Chair Generic	10	1	101
Text	5	5,917 Cotton Hat For repair	5	4	119
Text	6	2,157 Wooden Pizza Fantastic	75	6	154
Text	7	5,453 Metal Mouse Gently Used	5	3	111
Text	8	5,710 Rubber Salad Used	50	6	55
Text	9	3,912 Soft Chips Handmade	15	6	64
Text	10	3,217 Concrete Chair Generic	10	1	29
Text	11	2,154 Wooden Fish Fantastic	15	6	133
Text	12	922 Granite Shirt Ergonomic	10	4	15
Text	13	1,037 Frozen Mouse Ergonomic	50	3	68
Text	14	5,487 Soft Soap Gently Used	25	4	114
Text	15	2,082 Fresh Fish Incredible	5	6	7
Text	16	1,943 Plastic Sausages Incredible	15	6	150
Text	17	3,411 Frozen Computer Generic	15	3	49
Text	18	2,485 Cotton Hat Practical	25	4	111
Text	19	3,953 Frozen Chicken Handmade	1,000	6	49
Text	20	5,141 Granite Mouse New	10	3	136
Text	21	4,326 Cotton Bike Refined	15	5	150
Text	22	5,743 Soft Ball Used	50	5	5
Text	23	142 Granite Salad Small	5	6	70

products

DOKUMENTASI

Struktur Folder

```
└── dags
    ├── 1-dag-createtable.py
    ├── 2-dag-ETL.py
    ├── 3-dag-ETL-part2.py
    ├── 4-dag-ETL-part3.py
    ├── 5-main_dag.py
    ├── 6-dag-modelling.py
    ├── 7-dag-modelling.py
    ├── 8-dag-modelling.py
    ├── foreign_keys.py
    └── primary_keys.py

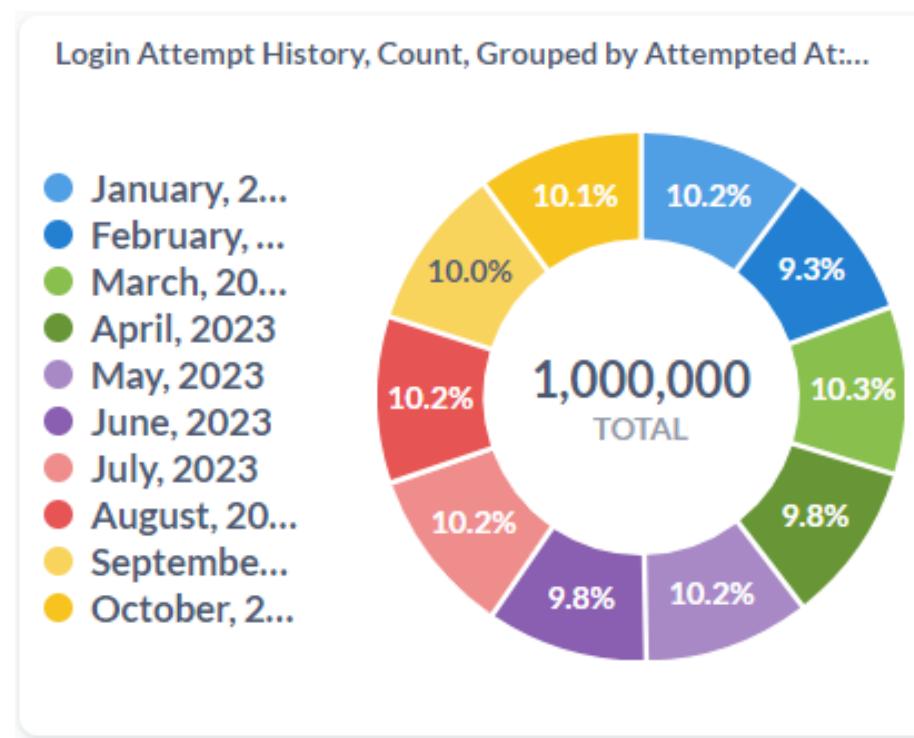
    ├── data
    ├── docker
    ├── scripts
    ├── .env
    ├── makefile
    ├── README.md
    └── requirements.txt
```

Dapat diakses di Github

 fathurrsyah	Update README.md	fa
└── dags	Go	
└── data	Go	
└── docker	Go	
└── scripts	Go	
└── .env	Go	
└── README.md	Update README.md	
└── makefile	Go	
└── requirements.txt	Go	

DOKUMENTASI

Dashboard Creation with Data Visualization



Menghitung jumlah Customer selama satu Tahun belakangan



Penjualan Per Kategori dan Jumlah Total Penjualan, dikelompokkan serta diurutkan berdasarkan Jumlah Total Penjualan secara menurun

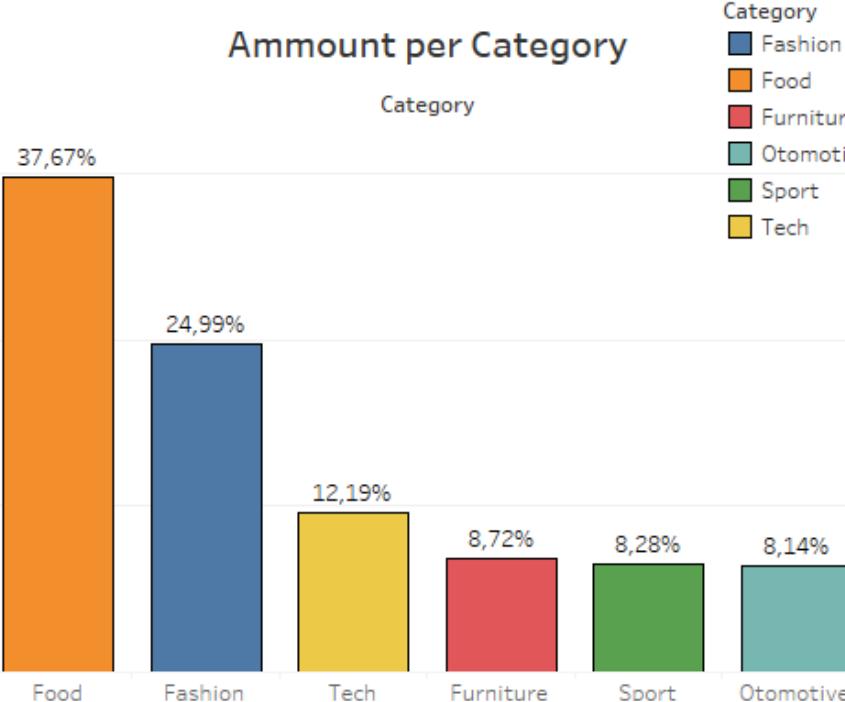
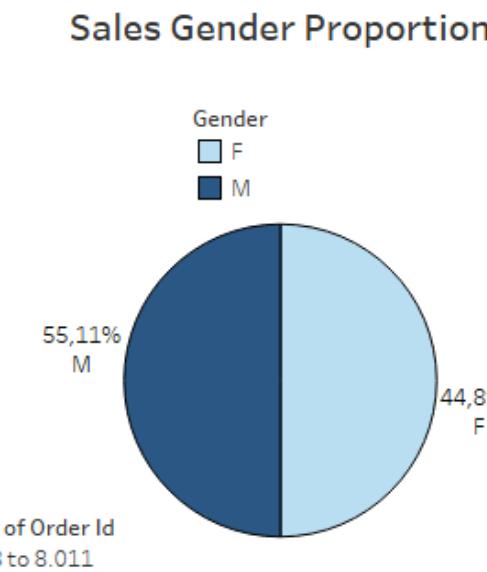
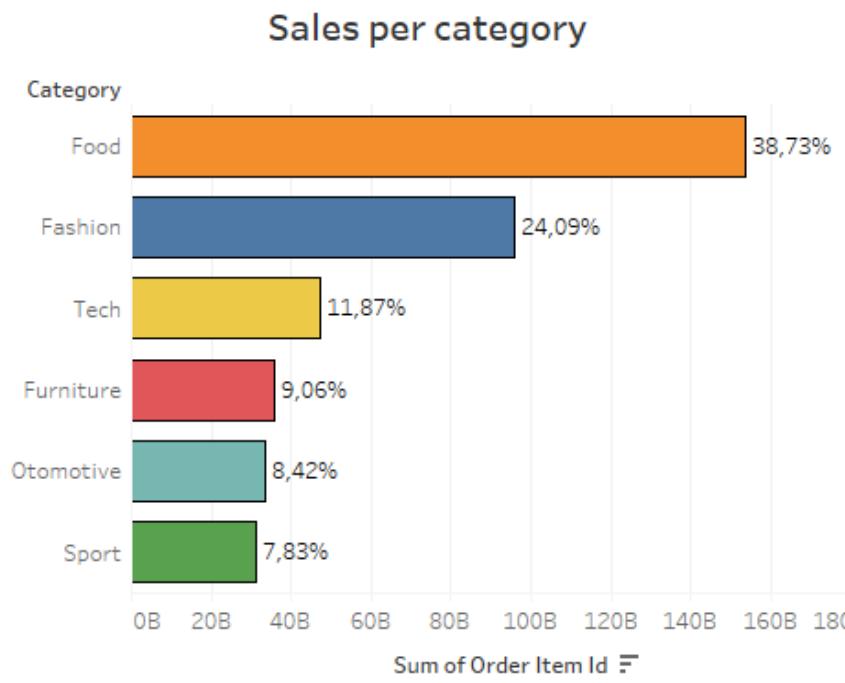
Top 10 Zip Code

Zip Code	Sum of Number Of Customers
20003	12
20031	10
20032	10
20017	8
20022	8
20029	8
20034	8
20035	8
19862	6
20004	6

menampilkan jumlah total pelanggan yang memiliki jumlah 10, 12, 8, atau 6 di setiap kode pos yang ada dalam tabel "zipcode_customer"

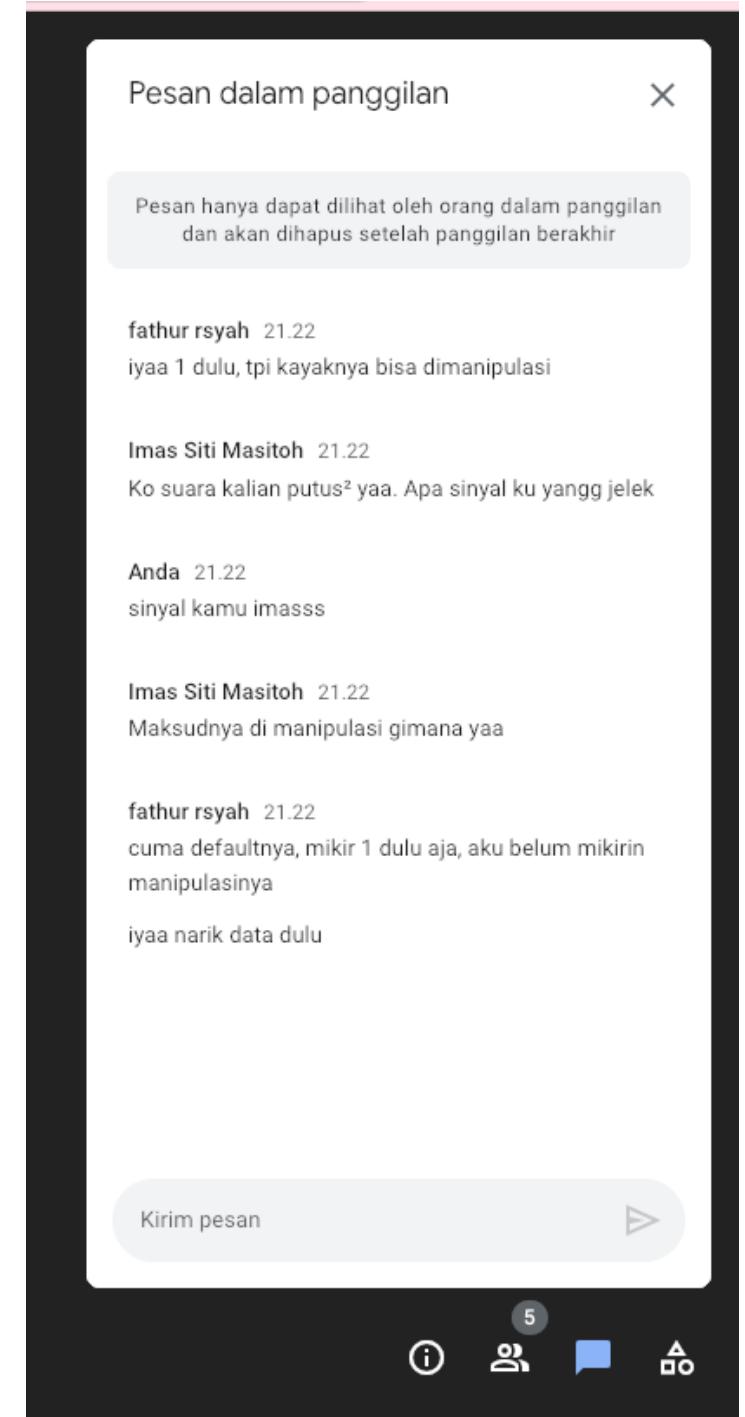
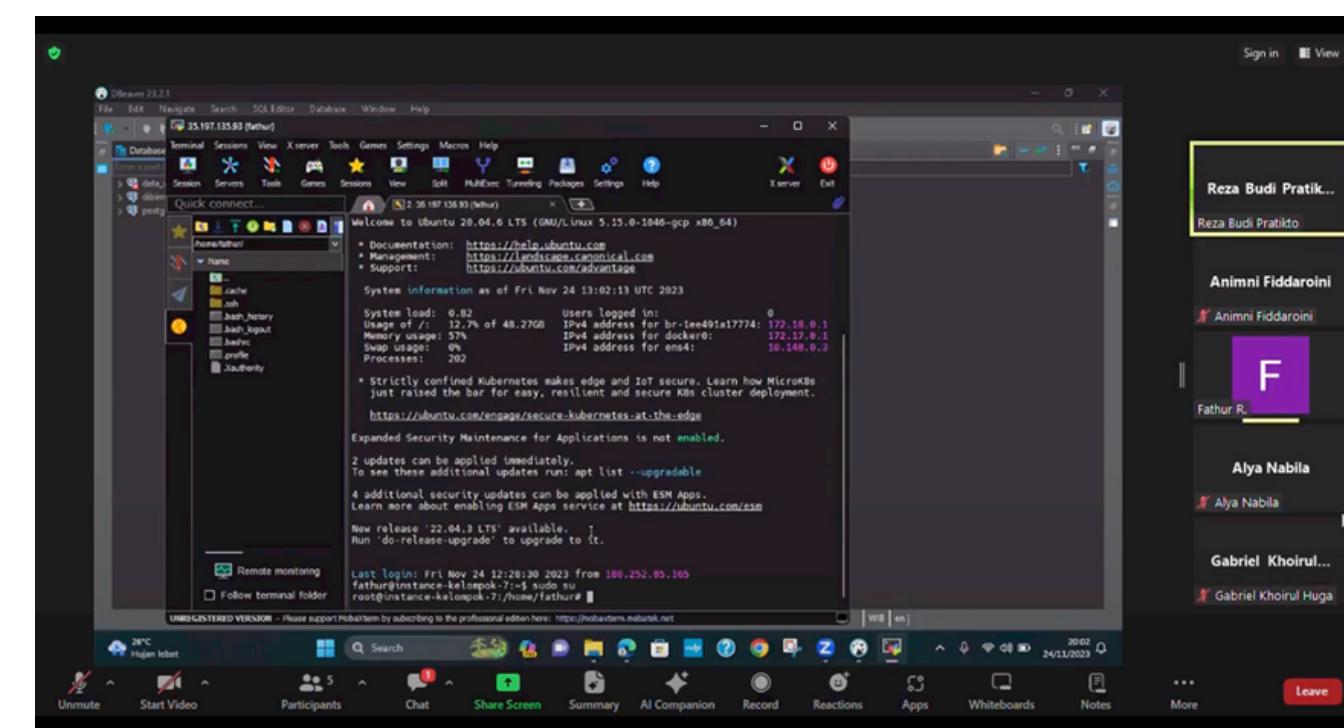
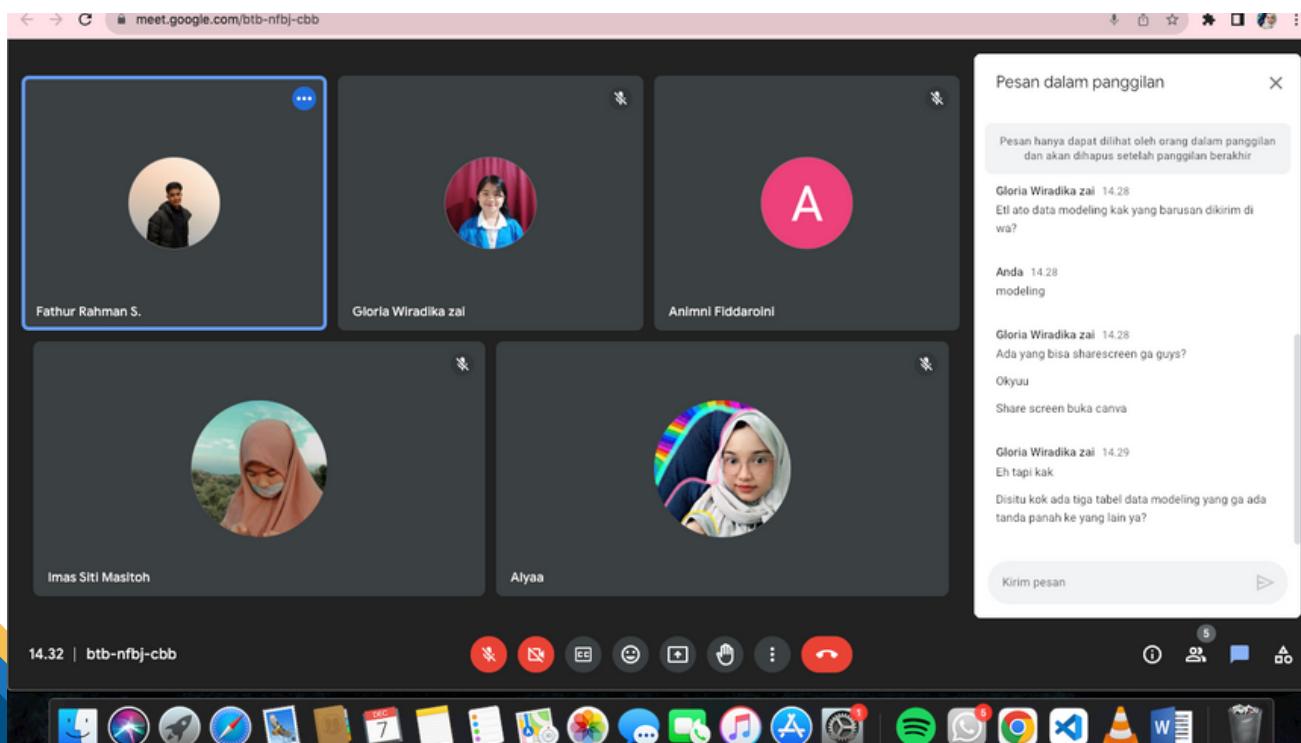
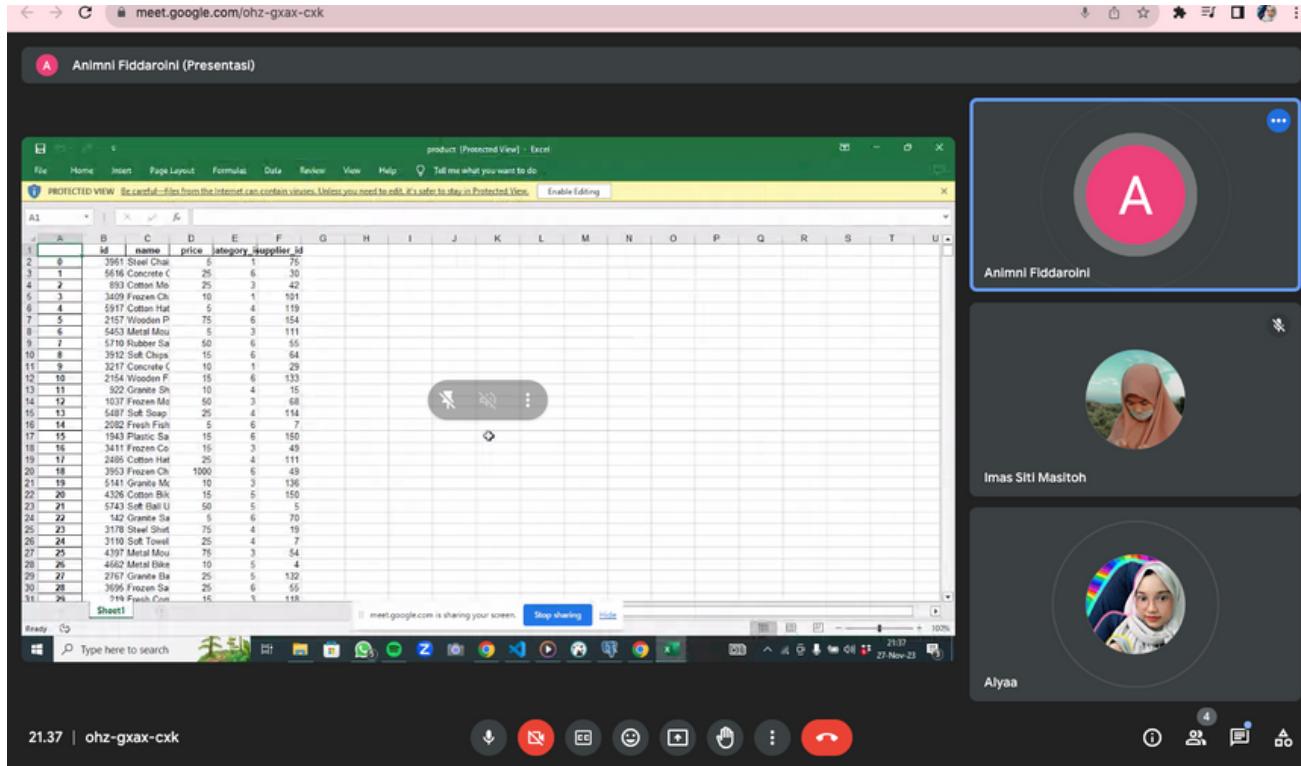
DOKUMENTASI

Pembuatan Dashboard menggunakan tableau



DOKUMENTASI

Aktivitas Diskusi Kelompok 7



THANK YOU

Dibimbing X Kampus Merdeka