LAPORAN TUGAS BESAR BIG DATA

Implementasi Pipeline Big Data ETL dan ELT pada Studi Kasus Transaksi Kartu Kredit Fraud dan Non Fraud

Mata Kuliah : Big Data
Program Studi : S1 Teknik Komputer
Fakultas : Fakultas Teknik Elektro

Disusun oleh

- Alfikri - 1103223015
- Raihan Abdul Majid – 1103223091

Dosen Pengampu
Angel Metanosa Afinda, S.Kom., M.Kom.

Tahun Akademik
Ganjil 2025/2026

ABSTRAK

Perkembangan sistem pembayaran digital, khususnya penggunaan kartu kredit, telah mendorong peningkatan volume transaksi keuangan secara signifikan. Di sisi lain, peningkatan ini juga diikuti oleh risiko fraud transaksi yang semakin kompleks dan sulit dideteksi. Data transaksi kartu kredit memiliki karakteristik volume yang sangat besar, variasi atribut yang beragam, serta kecepatan pertumbuhan data yang tinggi, sehingga memerlukan pendekatan Big Data Analytics untuk pengelolaannya.

Penelitian ini bertujuan untuk mengimplementasikan dan membandingkan pipeline Big Data menggunakan pendekatan Extract–Transform–Load (ETL) dan Extract–Load–Transform (ELT) pada studi kasus transaksi kartu kredit fraud dan non-fraud. Dataset utama yang digunakan berupa data transaksi kartu kredit dengan lebih dari satu juta baris data, serta data eksternal dari Federal Reserve Economic Data (FRED) sebagai data pendukung. Proses ETL diimplementasikan menggunakan Python untuk transformasi data sebelum dimuat ke dalam data warehouse PostgreSQL, sedangkan pendekatan ELT memanfaatkan transformasi langsung di dalam data warehouse berbasis query SQL.

Hasil pengolahan data disajikan dalam bentuk data warehouse dengan skema star schema serta dashboard analitik interaktif menggunakan Microsoft Power BI. Dashboard tersebut menampilkan pola fraud berdasarkan kategori transaksi, waktu kejadian, nominal transaksi, dan lokasi geografis. Analisis komparatif menunjukkan bahwa ETL lebih unggul dalam kontrol kualitas data sejak awal, sementara ELT menawarkan fleksibilitas dan kecepatan eksplorasi data yang lebih tinggi.

Kata kunci: Big Data, ETL, ELT, Fraud Transaksi, Data Warehouse, Dashboard Analitik

DAFTAR ISI

DAFTAR GAMBAR & DAFTAR TABEL

BAB I. PENDAHULUAN

1.1 Latar Belakang

Perkembangan sistem pembayaran digital telah mengubah cara masyarakat melakukan transaksi keuangan, dengan kartu kredit menjadi salah satu instrumen pembayaran yang paling banyak digunakan. Setiap transaksi kartu kredit menghasilkan jejak data yang mencakup waktu transaksi, nilai pembelanjaan, kategori merchant, lokasi geografis, serta identitas transaksi. Dalam skala operasional perbankan dan lembaga keuangan, akumulasi transaksi tersebut membentuk data dalam jumlah sangat besar dan terus bertambah setiap waktu.

Di balik kemudahan yang ditawarkan, penggunaan kartu kredit juga menghadirkan risiko fraud transaksi yang signifikan. Fraud transaksi kartu kredit dapat berupa transaksi tidak sah, penyalahgunaan data kartu, maupun aktivitas penipuan terorganisir yang dilakukan melalui berbagai saluran, baik offline maupun online. Karakteristik fraud yang jarang terjadi namun berdampak besar menyebabkan fraud sering tersembunyi di antara jutaan transaksi non-fraud, sehingga sulit dideteksi menggunakan pendekatan konvensional.

Data transaksi kartu kredit memiliki karakteristik khas yang menjadikannya relevan sebagai studi kasus Big Data. Dari sisi volume, dataset transaksi dapat mencapai jutaan baris data. Dari sisi variasi, data mencakup atribut numerik, kategorikal, temporal, dan spasial. Selain itu, data transaksi juga bersifat dinamis dan terus bertambah, serta memiliki permasalahan kualitas data seperti missing value, duplikasi, outlier pada nominal transaksi, dan ketimpangan kelas antara transaksi fraud dan non-fraud (class imbalance). Kondisi ini membuat proses pengolahan dan analisis data menjadi kompleks dan tidak efisien apabila hanya menggunakan pendekatan pengolahan data tradisional.

Untuk memahami pola fraud dan membedakannya dari transaksi non-fraud, diperlukan pipeline pengolahan data yang mampu menangani data berskala besar secara sistematis, konsisten, dan terotomasi. Pendekatan Extract–Transform–Load (ETL) dan Extract–Load–Transform (ELT) merupakan dua paradigma utama dalam pengolahan Big Data yang banyak digunakan dalam sistem analitik modern. ETL menekankan transformasi data sebelum dimuat ke data warehouse untuk memastikan kualitas data sejak awal, sedangkan ELT memanfaatkan kemampuan komputasi data warehouse untuk melakukan transformasi secara fleksibel setelah data dimuat.

Dalam konteks analisis transaksi fraud dan non-fraud, pemilihan pendekatan ETL atau ELT dapat memengaruhi kualitas data, fleksibilitas eksplorasi analitik, serta kecepatan pengambilan insight. Oleh karena itu, studi kasus transaksi kartu kredit menjadi konteks yang relevan untuk mengimplementasikan sekaligus membandingkan kedua pendekatan tersebut. Hasil pengolahan data kemudian disajikan dalam bentuk data warehouse dan dashboard analitik untuk membantu mengidentifikasi pola fraud berdasarkan kategori transaksi, waktu kejadian, nominal transaksi, dan lokasi geografis.

1.2 Rumusan Masalah

Berdasarkan konteks studi kasus transaksi kartu kredit fraud dan non-fraud, serta kebutuhan analisis yang ingin dicapai, maka rumusan masalah dalam tugas besar ini dirumuskan sebagai berikut:

1. Bagaimana membangun pipeline Big Data yang mampu mengolah data transaksi kartu kredit berskala besar untuk membedakan karakteristik transaksi fraud dan non-fraud secara sistematis?
2. Bagaimana mengidentifikasi pola dan karakteristik fraud transaksi kartu kredit berdasarkan kategori transaksi, waktu kejadian, nominal transaksi, dan lokasi geografis menggunakan pendekatan analitik Big Data?
3. Bagaimana menyajikan hasil analisis transaksi fraud dan non-fraud dalam bentuk data warehouse dan dashboard analitik yang mendukung eksplorasi data secara multidimensi?

1.3 Tujuan

Tujuan umum dari proyek ini adalah membangun pipeline Big Data untuk mengolah dan menganalisis data transaksi kartu kredit guna mengidentifikasi pola dan karakteristik transaksi fraud, serta menyajikan hasil analisis tersebut dalam bentuk data warehouse dan dashboard analitik.

Secara khusus, tujuan analitik dari proyek ini meliputi:

1. Menganalisis proporsi transaksi fraud dibandingkan dengan transaksi non-fraud untuk memahami tingkat ketimpangan kelas dalam dataset.
2. Mengidentifikasi kategori transaksi yang memiliki tingkat risiko fraud tertinggi berdasarkan jumlah dan persentase transaksi fraud.
3. Menganalisis pola waktu terjadinya fraud transaksi kartu kredit berdasarkan jam transaksi untuk mengidentifikasi periode waktu dengan risiko fraud yang lebih tinggi.
4. Mengkaji distribusi transaksi fraud berdasarkan nominal transaksi guna memahami rentang nilai transaksi yang paling sering terkait dengan fraud.
5. Menganalisis persebaran transaksi fraud secara geografis berdasarkan lokasi transaksi untuk mengidentifikasi wilayah dengan konsentrasi fraud yang tinggi.

Tujuan-tujuan tersebut menjadi dasar dalam perancangan pipeline ETL dan ELT, penyusunan query analitik, serta pembangunan dashboard analitik yang digunakan dalam proyek ini.

1.4 Ruang Lingkup

Ruang lingkup tugas besar ini dibatasi pada pengolahan dan analisis data transaksi kartu kredit dengan label fraud dan non-fraud. Dataset utama yang digunakan berupa data transaksi kartu kredit berskala besar, sedangkan data pendukung diperoleh dari API Federal Reserve Economic Data (FRED) sebagai konteks ekonomi makro. Implementasi pipeline dilakukan menggunakan Python untuk ETL, PostgreSQL sebagai data warehouse, dan Microsoft Power BI sebagai alat visualisasi dashboard.

Analisis difokuskan pada data historis transaksi dan tidak mencakup pemrosesan data real-time maupun streaming. Selain itu, proyek ini tidak membahas pengembangan model prediktif machine learning secara mendalam, melainkan berfokus pada pengolahan data, analisis deskriptif, serta perbandingan pendekatan ETL dan ELT dalam konteks fraud dan non-fraud.

BAB II. DESKRIPSI DATA DAN STUDI KASUS

2.1 Deskripsi Studi Kasus

Studi kasus dalam tugas besar ini berfokus pada analisis transaksi kartu kredit dengan tujuan utama membedakan karakteristik transaksi fraud dan non-fraud. Dalam sistem pembayaran modern, setiap transaksi kartu kredit dicatat secara detail dan menghasilkan data yang mencerminkan perilaku pengguna, jenis pembelanjaan, waktu transaksi, serta lokasi geografis tempat transaksi dilakukan. Data tersebut menjadi sumber informasi penting bagi institusi keuangan dalam memantau aktivitas transaksi dan mengelola risiko fraud.

Fraud transaksi kartu kredit merupakan kejadian yang relatif jarang dibandingkan transaksi non-fraud, namun memiliki dampak finansial dan reputasi yang signifikan. Karakteristik fraud yang tersembunyi di antara jutaan transaksi normal menyebabkan analisis fraud menjadi permasalahan yang kompleks. Oleh karena itu, studi kasus ini diarahkan untuk memahami pola-pola fraud berdasarkan berbagai dimensi transaksi, seperti kategori pembelanjaan, waktu kejadian, nominal transaksi, dan lokasi geografis.

Tujuan analitik dari studi kasus ini adalah untuk mengidentifikasi perbedaan karakteristik antara transaksi fraud dan non-fraud secara deskriptif dan multidimensi. Hasil analisis diharapkan dapat memberikan insight mengenai kategori transaksi dengan risiko fraud yang lebih tinggi, waktu transaksi yang cenderung rawan fraud, rentang nominal transaksi yang sering terkait dengan fraud, serta wilayah geografis dengan konsentrasi fraud yang lebih besar. Analisis ini menjadi dasar dalam perancangan pipeline Big Data serta penyusunan dashboard analitik pada tahap selanjutnya.

2.2 Sumber Data

Untuk mendukung analisis fraud dan non-fraud, proyek ini menggunakan dua sumber data yang berbeda, yaitu data transaksi utama dan data eksternal pendukung. Penggunaan data multisumber bertujuan untuk memperkaya konteks analisis serta memenuhi kebutuhan implementasi pipeline Big Data.

Sumber data disajikan dalam bentuk tabel berikut.

Tabel 2.1 Penggunaan Sumber Data

No	Sumber Data	Jenis	Format	Peran
1	Dataset Transaksi Kartu Kredit	File	CSV	Data utama transaksi fraud dan non-fraud

2	Federal Reserve Economic Data (FRED)	API	JSON	Data pendukung ekonomi makro

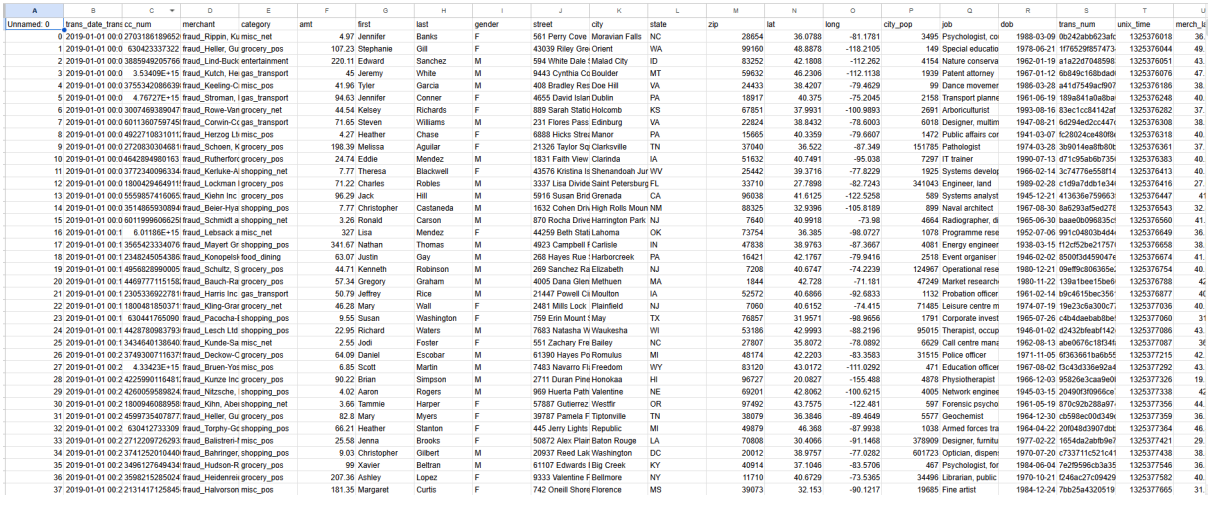
Dataset transaksi kartu kredit berperan sebagai data utama yang dianalisis untuk mengidentifikasi pola fraud. Sementara itu, data dari API Federal Reserve Economic Data (FRED) digunakan sebagai data pendukung untuk memberikan konteks ekonomi makro yang dapat dikaitkan dengan periode waktu transaksi..

2.3 Karakteristik Dataset

2.3.1 Dataset Transaksi Kartu Kredit (Data Utama)

Dataset utama yang digunakan dalam proyek ini berupa data transaksi kartu kredit yang tersimpan dalam format CSV. Dataset ini merepresentasikan transaksi individual yang dilakukan oleh pengguna kartu kredit dan mencakup informasi transaksi, pengguna, merchant, serta lokasi geografis.

Sebelum dilakukan proses preprocessing, dataset transaksi kartu kredit memiliki lebih dari 1.000.000 baris data dengan 24 kolom. Setiap baris merepresentasikan satu transaksi kartu kredit. Setelah melalui tahap preprocessing pada pipeline ETL, seperti penghapusan duplikasi, penanganan missing value, dan validasi data, jumlah baris data berkurang sesuai dengan transaksi yang tidak memenuhi kriteria kualitas data. Pengurangan ini menunjukkan dampak langsung dari proses pembersihan data terhadap kualitas dataset yang digunakan untuk analisis lanjutan.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
	Unnamed: 0	trans_date_trans	cc_num	merchant	category	amt	first	last	gender	street	city	state	zip	lat	long	city_pop	job	dob	trans_num	unix_time	merch_la
0	0	2019-01-01 00:0	2703186189652	fraud_Rippin, Ku	misc_net	4.97	Jennifer	Banks	F	561 Perry Cove	Moravian Falls	NC	28654	36.0788	-81.1781	3495	Psychologist, co	1988-03-09	0b242abb623afc	1325376018	36.
1	1	2019-01-01 00:0	630423337322	fraud_Heller, Gu	grocery_pos	107.23	Stephanie	Gill	F	43039 Riley Gre	Orient	WA	99160	48.8878	-118.2105	149	Special educatio	1978-06-21	1f76529f857473	1325376044	49.
2	2	2019-01-01 00:0	3885949205766	fraud_Lind-Buck	entertainment	220.11	Edward	Sanchez	M	594 White Dale	Malad City	ID	83252	42.1808	-112.262	4154	Nature conserva	1962-01-19	a1a22d70485983	1325376051	43.
3	3	2019-01-01 00:0	3.53408E+15	fraud_Kutch, Her	gas_transport	45	Jeremy	White	M	9443 Cynthia Ct	Boulder	MT	59632	46.2306	-112.1138	1939	Patent attorney	1967-01-12	6b849c168bdad6	1325376076	47.
4	4	2019-01-01 00:0	3755342086539	fraud_Keeling-C	misc_pos	41.96	Tyler	Garcia	M	408 Bradley Rest	Doe Hill	VA	24433	38.4207	-79.4629	99	Dance movemer	1986-03-28	a41d7549ac907	1325376186	38.
5	5	2019-01-01 00:0	4.76727E+15	fraud_Stroman, J	gas_transport	94.63	Jennifer	Conner	F	4655 David Islan	Dublin	PA	18917	40.375	-75.2045	2158	Transport planne	1961-06-19	189a841a0a8ba	1325376248	40.
6	6	2019-01-01 00:0	3007469389047	fraud_Rowe-Van	grocery_net	44.54	Kelsey	Richards	F	889 Sarah Statio	Holcomb	KS	67851	37.9931	-100.9893	2691	Arboriculturist	1993-08-16	83ec1cc84142af	1325376282	37.
7	7	2019-01-01 00:0	6011360759745	fraud_Corwin-Cr	gas_transport	71.65	Steven	Williams	M	231 Flores Pass	Edinburg	VA	22824	38.8432	-78.6003	6018	Designer, multim	1947-08-21	6d294ed2cc447	1325376308	38.
8	8	2019-01-01 00:0	4922710831011	fraud_Herzog Lt	misc_pos	4.27	Heather	Chase	F	6888 Hicks Stree	Manor	PA	15665	40.3359	-79.6607	1472	Public affairs cor	1941-03-07	fc28024ce480f8	1325376318	40.
9	9	2019-01-01 00:0	2720830304681	fraud_Schoen, K	grocery_pos	198.39	Melissa	Aguilar	F	21326 Taylor Sq	Clarksville	TN	37040	36.522	-87.349	151785	Pathologist	1974-03-28	3b9014ea8fb80	1325376361	37.
10	10	2019-01-01 00:0	4642894860163	fraud_Rutherford	grocery_pos	24.74	Eddie	Mendez	M	1831 Faith View	Clarinda	IA	51632	40.7491	-95.038	7297	IT trainer	1990-07-13	d71c95ab6b735	1325376383	40.
11	11	2019-01-01 00:0	3772340986334	fraud_Kerluke-Ab	shopping_net	7.77	Theresa	Blackwell	F	43576 Kristina Is	Shenandoah Jur	WV	25442	39.3716	-77.8229	1925	Systems develop	1966-02-14	3c74776e558f14	1325376413	40.
12	12	2019-01-01 00:0	180042946491	fraud_Lockman I	grocery_pos	71.22	Charles	Robles	M	3337 Lisa Divide	Saint Petersburg	FL	33710	27.7898	-82.7243	341043	Engineer, land	1989-02-28	c1d9a7ddb1e34	1325376416	27.
13	13	2019-01-01 00:0	5559857416065	fraud_Kiehn Inc	grocery_pos	96.29	Jack	Hill	M	5916 Susan Brid	Grenada	CA	96038	41.6125	-122.5258	589	Systems analyst	1945-12-21	413636e759663	1325376447	41
14	14	2019-01-01 00:0	3514865930894	fraud_Beier-Hya	shopping_pos	7.77	Christopher	Castaneda	M	1632 Cohen Driv	High Rolls Moun	NM	88325	32.9396	-105.8189	899	Naval architect	1967-08-30	8a6293af5ed278	1325376543	32.
15	15	2019-01-01 00:0	6011999606625	fraud_Schmidt a	shopping_net	3.26	Ronald	Carson	M	870 Rocha Drive	Harrington Park	NJ	7640	40.9918	-73.98	4664	Radiographer, di	1965-06-30	baae0b096835c	1325376560	41.
16	16	2019-01-01 00:1	6.01186E+15	fraud_Lebsack a	misc_net	327	Lisa	Mendez	F	44259 Beth Stati	Lahoma	OK	73754	36.385	-98.0727	1078	Programme rese	1952-07-06	991c04803b4d4	1325376649	36.
17	17	2019-01-01 00:1	3565423334076	fraud_Mayert Gr	shopping_pos	341.67	Nathan	Thomas	M	4923 Campbell F	Carlisle	IN	47838	38.9763	-87.3667	4081	Energy engineer	1938-03-15	f12cf52be21757	1325376658	38.
18	18	2019-01-01 00:1	2348245054386	fraud_Konopelski	food_dining	63.07	Justin	Gay	M	268 Hayes Rue	Harborcreek	PA	16421	42.1767	-79.9416	2518	Event organiser	1946-02-02	8500f3d459047e	1325376674	41.
19	19	2019-01-01 00:1	4958828990065	fraud_Schultz, S	grocery_pos	44.71	Kenneth	Robinson	M	269 Sanchez Ra	Elizabeth	NJ	7208	40.6747	-74.2239	124967	Operational rese	1980-12-21	09eff9c806365e	1325376754	40.
20	20	2019-01-01 00:1	4469777115158	fraud_Bauch-Ra	grocery_pos	57.34	Gregory	Graham	M	4905 Dana Glen	Methuen	MA	1844	42.728	-71.181	47249	Market research	1980-11-22	139a1bee15be6	1325376788	42.
21	21	2019-01-01 00:2	2305336922781	fraud_Harris Inc	gas_transport	50.79	Jeffrey	Rice	M	21447 Powell Cir	Moulton	IA	52572	40.6866	-92.6833	1132	Probation officer	1961-02-14	b9c4615bec356	1325376877	40
22	22	2019-01-01 00:1	1800481850371	fraud_Kling-Grar	grocery_net	46.28	Mary	Wall	F	2481 Mills Lock	Plainfield	NJ	7060	40.6152	-74.415	71485	Leisure centre m	1974-07-19	19e23c6a300c7	1325377036	40.
23	23	2019-01-01 00:1	630441765090	fraud_Pacocha-I	shopping_pos	9.55	Susan	Washington	F	759 Erin Mount	May	TX	76857	31.9571	-98.9656	1791	Corporate invest	1965-07-26	c4b4daeba88be	1325377060	31
24	24	2019-01-01 00:1	4428780983793	fraud_Lesch Ltd	shopping_pos	22.95	Richard	Waters	M	7683 Natasha W	Waukesha	WI	53186	42.9993	-88.2196	95015	Therapist, occup	1946-01-02	d2432bfeabf142	1325377086	43.
25	25	2019-01-01 00:1	3434640138640	fraud_Kunde-Sa	misc_net	2.55	Jodi	Foster	F	551 Zachary Fre	Bailey	NC	27807	35.8072	-78.0892	6629	Call centre mana	1962-08-13	abe0676c18f34f	1325377087	36
26	26	2019-01-01 00:2	3749300711637	fraud_Deckow-C	grocery_pos	64.09	Daniel	Escobar	M	61390 Hayes Po	Romulus	MI	48174	42.2203	-83.3583	31515	Police officer	1971-11-05	6f363661ba6b55	1325377215	42.
27	27	2019-01-01 00:2	4.33423E+15	fraud_Bruen-Yos	misc_pos	6.85	Scott	Martin	M	7483 Navarro Fi	Freedom	WY	83120	43.0172	-111.0292	471	Education officer	1967-08-02	f3c430336e92a4	1325377292	43.
28	28	2019-01-01 00:2	4225990116481	fraud_Kunze Inc	grocery_pos	90.22	Brian	Simpson	M	2711 Duran Pine	Honokaa	HI	96727	20.0827	-155.488	4878	Physiotherapist	1966-12-03	95826a3caa9e0	1325377326	19.
29	29	2019-01-01 00:2	4260059589824	fraud_Nitzsche,	shopping_pos	4.02	Aaron	Rogers	M	969 Huerta Path	Valentine	NE	69201	42.8062	-100.6215	4005	Network enginee	1945-03-15	20490f3f0966ce	1325377338	42
30	30	2019-01-01 00:2	1800946088958	fraud_Klhn, Abe	shopping_net	3.66	Tammie	Harper	F	57887 Gutierrez	Westfir	OR	97492	43.7575	-122.481	597	Forensic psycho	1961-05-19	870c92b288a97	1325377356	44.
31	31	2019-01-01 00:2	4599735407877	fraud_Heller, Gu	grocery_pos	82.8	Mary	Myers	F	39787 Pamela F	Tiptonville	TN	38079	36.3846	-89.4649	5577	Geochemist	1964-12-30	cb598ec0d349c	1325377359	36.
32	32	2019-01-01 00:2	630412733309	fraud_Torphy-Gc	shopping_pos	66.21	Heather	Stanton	F	445 Jerry Lights	Republic	MI	49879	46.368	-87.9938	1038	Armed forces tra	1964-04-22	20f048d3907dbb	1325377364	46.
33	33	2019-01-01 00:2	2712209726293	fraud_Balistreri-I	misc_pos	25.58	Jenna	Brooks	F	50672 Alex Plair	Baton Rouge	LA	70808	30.4066	-91.1468	378909	Designer, furnitu	1977-02-22	1654da2abfb9e7	1325377421	29.
34	34	2019-01-01 00:2	3741252010440	fraud_Bahringer,	shopping_pos	9.03	Christopher	Gilbert	M	20937 Reed Lak	Washington	DC	20012	38.9757	-77.0282	601723	Optician, dispen	1970-07-20	c73711c521c41	1325377438	38.
35	35	2019-01-01 00:2	3496127648434	fraud_Hudson-R	grocery_pos	99	Xavier	Beltran	M	61107 Edwards I	Big Creek	KY	40914	37.1046	-83.5706	467	Psychologist, for	1984-06-04	7e2f9596cb3a35	1325377546	36.
36	36	2019-01-01 00:2	3598215285024	fraud_Heidenrei	grocery_pos	207.36	Ashley	Lopez	F	9333 Valentine F	Bellmore	NY	11710	40.6729	-73.5365	34496	Librarian, public	1970-10-21	f24ac27c09429	1325377582	40.
37	37	2019-01-01 00:2	2131417125845	fraud_Halvorson	misc_pos	181.35	Margaret	Curtis	F	742 Oneill Shore	Florence	MS	39073	32.153	-90.1217	19685	Fine artist	1984-12-24	7bb25a4320519	1325377665	31.

Gambar 2.1 Dataset Utama (CSV)

Gambar (Gambar 2.1) menampilkan contoh data dari dataset transaksi kartu kredit sebelum preprocessing. Dari gambar tersebut terlihat struktur dataset yang mencakup atribut waktu transaksi, identitas transaksi, kategori pembelanjaan, nominal transaksi, serta atribut demografis dan geografis pengguna.

Tipe kolom dalam dataset transaksi kartu kredit dapat diklasifikasikan sebagai berikut:

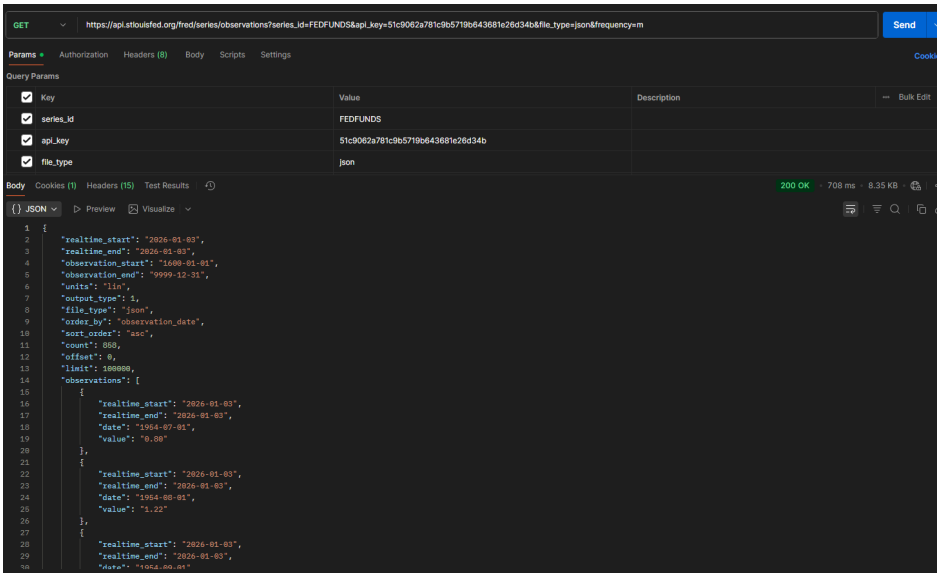
- **Kolom numerik**
  Contoh: amt, city_pop, lat, long, unix_time
  Kolom ini digunakan untuk analisis kuantitatif, seperti distribusi nominal transaksi dan analisis spasial.
- **Kolom kategorikal**
  Contoh: category, gender, state, job, merchant
  Kolom ini berperan penting dalam analisis perbandingan risiko fraud berdasarkan kategori transaksi dan karakteristik pengguna.
- **Kolom bertipe waktu (temporal)**
  Contoh: trans_date_trans_time, dob
  Kolom ini digunakan untuk analisis pola waktu terjadinya transaksi fraud dan non-fraud.
- **Kolom identitas (ID)**
  Kolom trans_num berfungsi sebagai primary key yang merepresentasikan identitas unik setiap transaksi. Kolom ini digunakan sebagai acuan utama dalam proses penghapusan data duplikat serta relasi dengan tabel dimensi pada data warehouse.

Dari sisi kualitas data, dataset transaksi kartu kredit memiliki beberapa permasalahan yang perlu ditangani. Ditemukan missing value pada beberapa atribut tertentu, yang berpotensi mengganggu proses analisis jika tidak ditangani dengan tepat. Selain itu, terdapat kemungkinan data duplikat berdasarkan identitas transaksi, sehingga diperlukan validasi berbasis primary key. Dataset juga menunjukkan keberadaan outlier pada nominal transaksi, yaitu transaksi dengan nilai ekstrem yang dapat mendistorsi hasil analisis distribusi jika tidak dilakukan penanganan, seperti transformasi atau pembatasan nilai.

Karakteristik penting lainnya adalah ketimpangan kelas (class imbalance) antara transaksi fraud dan non-fraud, di mana transaksi non-fraud jauh lebih dominan. Kondisi ini memiliki implikasi signifikan terhadap analisis fraud, karena interpretasi hasil harus mempertimbangkan bahwa jumlah fraud relatif kecil namun berdampak besar.

2.3.2 Dataset API Federal Reserve Economic Data (FRED)

Dataset pendukung diperoleh melalui API Federal Reserve Economic Data (FRED) menggunakan endpoint fred/series/observations dengan seri data FEDFUNDS. Dataset ini menyajikan data time-series suku bunga acuan Federal Funds Rate dalam format JSON.



Gambar 2.2 Dataset Pendukung (API)

Berdasarkan hasil ekstraksi, dataset FRED memiliki ratusan baris data observasi dengan beberapa kolom utama, seperti date dan value. Setiap baris merepresentasikan satu observasi suku bunga pada periode waktu tertentu dengan frekuensi bulanan. Gambar (Gambar 2.2) menampilkan contoh response JSON dari API FRED yang digunakan dalam proyek ini.

Tipe kolom pada dataset FRED meliputi:

- **Kolom bertipe waktu (temporal)**
  date, yang merepresentasikan periode observasi data ekonomi.
- **Kolom numerik**
  value, yang menunjukkan nilai suku bunga acuan pada periode tertentu.

Dataset FRED tidak memiliki primary key eksplisit, namun kolom date digunakan sebagai kunci temporal untuk proses penggabungan (join) dengan dataset transaksi kartu kredit pada level periode waktu bulanan. Dataset ini relatif bersih dan tidak menunjukkan permasalahan missing value yang signifikan, namun tetap dilakukan validasi tipe data untuk memastikan kolom nilai bersifat numerik.

2.3.3 Implikasi Karakteristik Data terhadap Pipeline Big Data

Karakteristik kedua dataset menegaskan perlunya penerapan pipeline Big Data yang terstruktur. Dataset transaksi kartu kredit memerlukan proses pembersihan dan validasi yang ketat untuk menangani missing value, duplikasi, dan outlier. Sementara itu, dataset FRED digunakan sebagai data pendukung yang memperkaya konteks analisis dan memerlukan penyesuaian temporal agar dapat diintegrasikan dengan data transaksi.

Perbedaan karakteristik antara data utama dan data API juga menjadi dasar pemilihan pendekatan ETL dan ELT. Pada pendekatan ETL, transformasi data dilakukan sebelum pemuatan ke data warehouse untuk memastikan kualitas data sejak awal. Sebaliknya, pendekatan ELT memanfaatkan kemampuan data warehouse untuk melakukan transformasi dan eksplorasi data secara fleksibel setelah data dimuat.

BAB III. ARSITEKTUR SISTEM

3.1 Desain Arsitektur

Arsitektur sistem Big Data pada proyek ini dirancang untuk mendukung pengolahan dan analisis data transaksi kartu kredit fraud dan non-fraud secara end-to-end, mulai dari pengambilan data mentah hingga penyajian hasil analisis dalam bentuk dashboard analitik. Arsitektur ini mengadopsi dua pendekatan pengolahan data, yaitu Extract–Transform–Load (ETL) dan Extract–Load–Transform (ELT), yang diimplementasikan secara paralel untuk tujuan analisis komparatif.

Sistem menerima data dari dua sumber utama. Sumber pertama adalah dataset transaksi kartu kredit dalam format CSV yang berisi lebih dari satu juta baris transaksi. Sumber kedua adalah data eksternal berupa data suku bunga acuan dari API Federal Reserve Economic Data (FRED) dengan seri FEDFUNDS. Proses pengambilan data dari kedua sumber dilakukan menggunakan bahasa pemrograman Python.

Pada tahap awal, seluruh data hasil ekstraksi disimpan dalam bentuk data mentah (raw data) tanpa transformasi. Penyimpanan ini dilakukan pada Google Drive dan berfungsi sebagai data lake, yang menjadi lapisan penyimpanan awal dalam arsitektur sistem. Data lake digunakan untuk menjaga reprodusibilitas proses, memungkinkan audit data, serta memastikan bahwa proses transformasi dapat diulang tanpa perlu melakukan ekstraksi ulang dari sumber data.

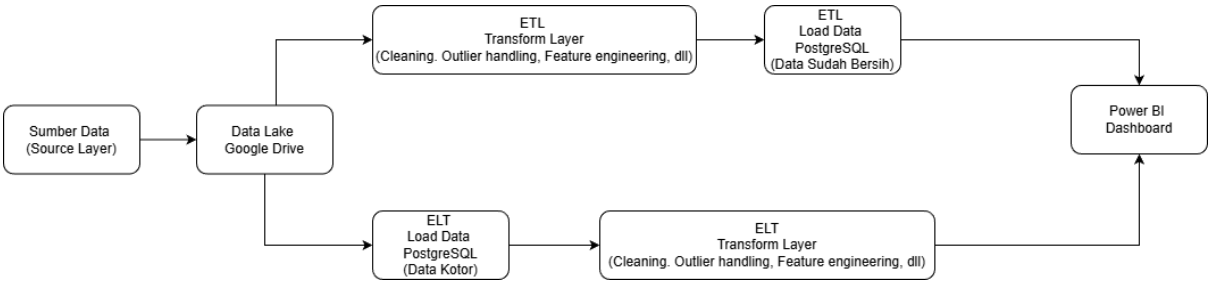
Setelah tahap extract, alur data dipisahkan menjadi dua jalur pengolahan yang berbeda. Pada jalur ETL, data mentah dari data lake diproses terlebih dahulu melalui tahap transformasi menggunakan Python. Transformasi mencakup pembersihan data, penanganan missing value, penghapusan duplikasi berdasarkan identitas transaksi, standardisasi format waktu, penanganan outlier pada nominal transaksi, feature engineering, serta pengayaan data dengan data suku bunga dari FRED. Proses validasi kualitas data juga diterapkan untuk memastikan integritas dan konsistensi data sebelum dimuat ke dalam data warehouse.

Pada jalur ELT, data mentah dari data lake dimuat langsung ke dalam data warehouse tanpa melalui transformasi awal. Transformasi data dilakukan di dalam data warehouse menggunakan query SQL, termasuk proses pembersihan, penggabungan data, serta pembentukan fitur analitik. Pendekatan ELT ini memanfaatkan kemampuan komputasi data warehouse untuk mendukung eksplorasi data yang lebih fleksibel dan cepat ketika kebutuhan analitik berubah.

Data warehouse berperan sebagai pusat penyimpanan data terstruktur yang digunakan untuk analisis fraud dan non-fraud. Data warehouse dibangun menggunakan PostgreSQL dan dirancang dengan skema dimensional (star schema), yang terdiri dari satu tabel fakta transaksi dan beberapa tabel dimensi, seperti dimensi waktu, kategori transaksi, lokasi, dan karakteristik pelanggan. Desain ini memungkinkan query analitik dan agregasi data dilakukan secara efisien untuk mendukung analisis multidimensi.

Lapisan terakhir dari arsitektur sistem adalah dashboard analitik, yang dibangun menggunakan Microsoft Power BI. Dashboard ini terhubung langsung ke data warehouse PostgreSQL dan digunakan untuk memvisualisasikan hasil analisis transaksi fraud dan non-fraud. Visualisasi yang disajikan mencakup metrik utama, perbandingan fraud dan non-fraud, pola waktu terjadinya fraud, distribusi nominal transaksi, serta persebaran fraud secara geografis. Dengan demikian, arsitektur sistem ini mendukung alur data yang utuh dan terintegrasi dari hulu ke hilir.

3.2 Diagram Arsitektur



Gambar 3.1 Diagram Pipeline

Diagram arsitektur sistem Big Data pada Gambar 3.1 menggambarkan alur pengolahan data transaksi kartu kredit fraud dan non-fraud menggunakan dua pendekatan yang berbeda, yaitu pipeline ETL (Extract–Transform–Load) dan pipeline ELT (Extract–Load–Transform). Diagram tersebut disusun untuk menunjukkan secara jelas perbedaan urutan proses, tujuan desain, serta peran masing-masing komponen dalam sistem yang dibangun.

Alur data dimulai dari Sumber Data (Source Layer) yang terdiri dari dataset transaksi kartu kredit dalam format CSV dan data ekonomi makro dari API Federal Reserve Economic Data (FRED). Seluruh data dari sumber tersebut diekstraksi menggunakan Python dan disimpan terlebih dahulu ke dalam Data Lake berbasis Google Drive. Pada tahap ini, data disimpan dalam kondisi mentah tanpa transformasi untuk menjaga reprodusibilitas proses dan memungkinkan audit data.

Dari data lake, alur pengolahan data kemudian bercabang menjadi dua jalur yang berbeda. Pada jalur ETL, data mentah terlebih dahulu masuk ke ETL Transform Layer yang diimplementasikan menggunakan Python. Pada tahap ini dilakukan proses transformasi data secara menyeluruh, meliputi pembersihan data (cleaning), penanganan outlier, feature engineering, serta proses pendukung lainnya untuk memastikan kualitas data. Setelah proses transformasi selesai, data yang telah berada dalam kondisi bersih dan terstruktur dimuat ke dalam PostgreSQL sebagai data warehouse, sebagaimana ditunjukkan pada komponen ETL Load Data PostgreSQL (Data Sudah Bersih). Data hasil ETL ini kemudian digunakan secara langsung sebagai sumber data untuk dashboard analitik.

Sebaliknya, pada jalur ELT, data mentah dari data lake langsung dimuat ke dalam PostgreSQL melalui komponen ELT Load Data PostgreSQL (Data Kotor) tanpa melalui

proses transformasi awal. Transformasi data dilakukan setelah proses load, yaitu pada ELT Transform Layer yang dijalankan di dalam data warehouse menggunakan query SQL. Proses ini mencakup pembersihan data, penanganan outlier, serta feature engineering untuk menghasilkan data yang siap digunakan dalam analisis. Pendekatan ini memanfaatkan kemampuan komputasi data warehouse untuk mendukung eksplorasi data yang lebih fleksibel.

Kedua jalur pengolahan data, baik ETL maupun ELT, bermuara pada sistem analitik yang sama, yaitu Power BI Dashboard. Dashboard ini terhubung langsung dengan PostgreSQL dan digunakan untuk memvisualisasikan hasil analisis transaksi fraud dan non-fraud, termasuk perbandingan fraud dan non-fraud, pola waktu transaksi, distribusi nominal transaksi, serta analisis berdasarkan kategori dan lokasi.

Diagram arsitektur ini secara jelas menunjukkan perbedaan mendasar antara pendekatan ETL dan ELT, baik dari sisi urutan proses transformasi maupun kondisi data saat dimuat ke data warehouse. Diagram tersebut juga konsisten dengan implementasi pipeline ETL dan ELT yang dijelaskan pada Bab IV dan Bab V, serta merepresentasikan sistem yang benar-benar dibangun dalam proyek ini, bukan sekadar desain konseptual.

BAB IV. PIPELINE ETL

4.1 Extract

Tahap extract bertujuan untuk mengambil data dari masing-masing sumber tanpa melakukan proses pembersihan atau transformasi apa pun. Pada proyek ini, data diekstraksi dari dua sumber yang berbeda, yaitu dataset transaksi kartu kredit dan API Federal Reserve Economic Data (FRED).

4.1.1 Ekstraksi Dataset Transaksi Kartu Kredit

Dataset transaksi kartu kredit diekstraksi dari file CSV menggunakan pustaka pandas. Proses ekstraksi dilakukan dengan membaca file CSV secara langsung dan menyimpannya kembali ke dalam direktori raw pada Google Drive sebagai data mentah. Pada tahap ini, tidak dilakukan pembersihan, transformasi, maupun modifikasi data dalam bentuk apa pun.

Lokasi penyimpanan raw data:
/dataset/bigdata/uas/raw/raw_credit_card.csv

4.1.2 Ekstraksi Data API Federal Reserve Economic Data (FRED)

Data eksternal diekstraksi dari API Federal Reserve Economic Data menggunakan endpoint fred/series/observations dengan seri data FEDFUNDS. Proses ekstraksi dilakukan menggunakan metode HTTP GET dan menghasilkan response dalam format JSON. Data hasil ekstraksi kemudian dikonversi menjadi DataFrame dan disimpan sebagai file CSV di direktori raw tanpa transformasi tambahan.

Lokasi penyimpanan raw data:
/dataset/bigdata/uas/raw/raw_fred_fedfunds.csv

4.1.3 Log Extract

Tabel berikut menunjukkan log proses extract dari masing-masing sumber data:

Tabel 4.1 Log Extract

No	Sumber Data	Metode Extract	Format	Jumlah Baris	Jumlah Kolom	Lokasi Raw Data	Waktu Eksekusi
1	Transaksi Kartu Kredit	Read CSV (Pandas)	CSV	±1.000.000	24	/raw/raw_credit_card.csv	±3 detik

2	FRED FEDFUNDS	HTTP GET (API)	JSON	±850	5	/raw/raw_fred_fedfunds.csv	±1 detik

Tahap extract ini memenuhi ketentuan bahwa tidak ada proses cleaning atau transformasi data yang dilakukan sebelum data disimpan sebagai raw data.

4.2 Transform

Tahap transform merupakan inti dari pipeline ETL, di mana data mentah diproses agar siap digunakan untuk analisis fraud dan non-fraud. Seluruh proses transformasi dilakukan menggunakan Python sebelum data dimuat ke dalam data warehouse.

4.2.1 Data Cleaning

- **Penghapusan duplikasi**
  Data duplikat dihapus berdasarkan kolom trans_num yang berfungsi sebagai identitas unik setiap transaksi. Pendekatan ini memastikan bahwa setiap transaksi hanya direpresentasikan satu kali dalam dataset.
- **Penanganan missing value**
  Strategi penanganan missing value dilakukan dengan membedakan kolom kritis dan non-kritis. Kolom kritis seperti cc_num, amt, category, dan gender harus memiliki nilai valid sehingga baris dengan nilai kosong pada kolom tersebut dihapus. Sementara itu, kolom non-kritis seperti merch_zipcode diisi dengan nilai default untuk menjaga jumlah data.
- **Standardisasi datetime**
  Kolom trans_date_trans_time dan dob dikonversi ke format datetime standar untuk mendukung analisis temporal dan perhitungan usia pengguna secara konsisten.
- **Penanganan outlier**
  Outlier pada kolom nominal transaksi (amt) ditangani menggunakan metode Interquartile Range (IQR) dengan teknik capping. Metode ini dipilih karena mampu membatasi nilai ekstrem tanpa menghilangkan transaksi secara langsung, sehingga sinyal fraud dengan nilai transaksi tinggi tetap dipertahankan untuk analisis.

4.2.2 Standardisasi Data

- **Penamaan kolom**
  Seluruh nama kolom distandardisasi ke format snake_case untuk menjaga konsistensi dan memudahkan penulisan query analitik.
- **Normalisasi data numerik**
  Normalisasi dilakukan menggunakan metode Min-Max Scaling pada minimal dua kolom numerik, yaitu amt dan age, sehingga nilai berada pada rentang 0–1 dan dapat dibandingkan secara proporsional.
- **Encoding kategorikal**
  Kolom gender diencoding menjadi bentuk numerik menggunakan pendekatan label encoding. Selain itu, kolom category diencoding menggunakan one-hot encoding untuk mendukung analisis per kategori transaksi.
- **Konsistensi tipe data**

Seluruh kolom numerik, kategorikal, dan temporal divalidasi untuk memastikan kesesuaian tipe data sebelum tahap pemuatan ke data warehouse.

4.2.3 Data Enrichment & Feature Engineering

Data transaksi kartu kredit digabungkan dengan data FRED berdasarkan periode waktu bulanan (year_month). Join ini bertujuan untuk memperkaya dataset transaksi dengan konteks ekonomi makro.

Pipeline ETL menghasilkan 8 fitur baru, antara lain:

1. year: tahun transaksi, untuk analisis tren tahunan
2. month: bulan transaksi
3. hour: jam transaksi, untuk analisis pola waktu fraud
4. age: usia pengguna saat transaksi
5. amt_log: transformasi log dari nominal transaksi
6. amt_norm: nominal transaksi ternormalisasi
7. age_norm: usia pengguna ternormalisasi
8. interest_rate: suku bunga acuan dari FRED

Setiap fitur dirancang untuk mendukung analisis fraud berdasarkan waktu, nilai transaksi, dan karakteristik pengguna.

4.2.4 Validasi Kualitas Data

Validasi kualitas data dilakukan menggunakan beberapa aturan berikut:

1. Uniqueness: memastikan trans_num bersifat unik
2. Null check: memastikan kolom kritis tidak memiliki nilai kosong
3. Range: memastikan nilai amt tidak bernilai negatif
4. Datatype: memastikan kolom interest_rate bertipe numerik
5. Range temporal: memastikan nilai tahun transaksi berada dalam rentang wajar
6. Referential consistency: memastikan relasi waktu (year_month) valid untuk proses join

Apabila data tidak memenuhi aturan validasi, dilakukan perbaikan pada tahap transformasi sebelum data dimuat ke data warehouse.

4.3 Load

Tahap load merupakan tahap akhir dalam pipeline ETL, di mana data yang telah melalui seluruh proses transformasi dan validasi kualitas dimuat ke dalam sistem penyimpanan terstruktur untuk keperluan analisis. Pada proyek ini, proses load dilakukan setelah data dipastikan bersih, konsisten, dan memenuhi seluruh aturan kualitas data yang telah ditetapkan pada tahap sebelumnya.

Data hasil transformasi dimuat ke dalam PostgreSQL yang berperan sebagai data warehouse. PostgreSQL dipilih karena mendukung penyimpanan data dalam skala besar, memiliki kemampuan query analitik yang kuat, serta mudah diintegrasikan dengan bahasa pemrograman Python dan alat visualisasi seperti Microsoft Power BI. Proses pemuatan data

dilakukan menggunakan pustaka SQLAlchemy, yang memungkinkan pemuatan data secara efisien dalam bentuk batch untuk menangani dataset dengan jumlah baris yang besar.

Struktur data warehouse dirancang menggunakan pendekatan skema dimensional (star schema). Pada skema ini, tabel utama yang digunakan adalah tabel fakta fact_credit_card_transactions, yang merepresentasikan transaksi kartu kredit sebagai unit analisis utama. Setiap baris pada tabel fakta merepresentasikan satu transaksi unik dan diidentifikasi oleh atribut trans_num sebagai primary key. Tabel fakta ini memuat atribut numerik dan hasil feature engineering yang digunakan untuk analisis fraud dan non-fraud.

Selain tabel fakta, data warehouse juga dirancang untuk mendukung penggunaan tabel dimensi, seperti dimensi waktu, kategori transaksi, lokasi, dan karakteristik pengguna. Relasi antara tabel fakta dan tabel dimensi direpresentasikan melalui foreign key, sehingga analisis multidimensi dapat dilakukan secara efisien. Meskipun pada implementasi awal beberapa atribut dimensi masih disimpan dalam satu tabel fakta, struktur ini tetap mengikuti prinsip star schema dan dapat dikembangkan lebih lanjut menjadi tabel dimensi terpisah apabila diperlukan.

Proses load dilakukan dengan pendekatan replace pada tabel tujuan untuk memastikan konsistensi data antara hasil transformasi terbaru dan data yang tersimpan di data warehouse. Selama proses pemuatan, data dimuat dalam ukuran chunk tertentu untuk mengoptimalkan performa dan mencegah kegagalan proses akibat keterbatasan memori.

Setelah proses load selesai, dilakukan verifikasi sederhana untuk memastikan bahwa jumlah baris data yang dimuat ke dalam data warehouse sesuai dengan jumlah baris pada data hasil transformasi. Data yang telah berhasil dimuat kemudian digunakan sebagai sumber utama untuk proses query analitik dan visualisasi pada dashboard Power BI. Dengan demikian, tahap load ini memastikan bahwa data hasil ETL telah tersedia dalam bentuk terstruktur dan siap digunakan untuk mendukung analisis transaksi fraud dan non-fraud secara efektif.

4.3.1 Query Analitik pada Tahap Load

Setelah data hasil transformasi dimuat ke dalam data warehouse PostgreSQL, dilakukan serangkaian query SQL analitik untuk mendukung analisis transaksi fraud dan non-fraud. Query-query ini dijalankan pada tabel fakta fact_credit_card_transactions serta beberapa view pendukung, dan digunakan sebagai dasar penyusunan dashboard analitik.

1. **Total Transaksi Keseluruhan**

Query | Query History

```
1  SELECT COUNT(*) AS total_transaksi
2  FROM fact_credit_card_transactions;
3
```

Gambar 4.1 Query Total Transaksi Keseluruhan

Hasil:

	total_transaksi bigint
1	1296675

Gambar 4.2 Hasil Query Total Transaksi Keseluruhan

Query ini digunakan untuk menghitung jumlah total transaksi yang tersimpan dalam tabel fakta. Tujuan dari query ini adalah untuk memperoleh gambaran umum skala data yang dianalisis setelah melalui proses ETL. Hasil query menunjukkan bahwa dataset terdiri dari 1.296.675 transaksi, yang menegaskan bahwa data yang digunakan berskala besar dan relevan untuk analisis Big Data.

2. **Total Transaksi Fraud**

Query | Query History

```
1  SELECT COUNT(*) AS total_fraud
2  FROM fact_credit_card_transactions
3  WHERE is_fraud = 1;
4
```

Gambar 4.3 Query Total Transaksi Fraud

Hasil:

	total_fraud bigint
1	7506

Gambar 4.4 Hasil Query Total Transaksi Fraud

Query ini bertujuan untuk menghitung jumlah transaksi yang teridentifikasi sebagai fraud berdasarkan atribut is_fraud. Dengan memfilter transaksi yang bernilai fraud, query ini memberikan informasi mengenai jumlah absolut kejadian fraud dalam dataset. Hasil yang diperoleh menunjukkan terdapat 7.506 transaksi fraud, yang menandakan bahwa kasus fraud merupakan kejadian minoritas dibandingkan total transaksi.

3. **Persentase Fraud vs Non-Fraud**

```sql
SELECT
    is_fraud,
    COUNT(*) * 100.0 / SUM(COUNT(*)) OVER () AS percentage
FROM fact_credit_card_transactions
GROUP BY is_fraud;
```

Gambar 4.5 Query Persentase Fraud vs Non-Fraud

Hasil:

	is_fraud bigint	percentage numeric
1	0	99.4211348256116606
2	1	0.57886517438833940656

Gambar 4.6 Hasil Query Persentase Fraud vs Non-Fraud

Query ini digunakan untuk menghitung proporsi transaksi fraud dan non-fraud dalam bentuk persentase. Perhitungan dilakukan dengan membandingkan jumlah transaksi pada masing-masing kelas terhadap total transaksi. Hasil analisis menunjukkan bahwa transaksi non-fraud mendominasi dataset dengan persentase lebih dari 99%, sedangkan transaksi fraud berada di bawah 1%. Temuan ini mengonfirmasi adanya ketimpangan kelas (class imbalance) yang signifikan, yang merupakan karakteristik umum pada data fraud transaksi.

4. **Jumlah Fraud Berdasarkan Kategori Transaksi**

```sql
SELECT
    category_name,
    COUNT(*) AS fraud_count
FROM dw.v_fact_with_category
WHERE is_fraud = 1
GROUP BY category_name
ORDER BY fraud_count DESC;
```

Gambar 4.7 Query Jumlah Fraud Berdasarkan Kategori Transaksi

Hasil:

	category_name text	fraud_count bigint
1	grocery_pos	1743
2	shopping_net	1713
3	misc_net	915
4	shopping_pos	843
5	gas_transport	618
6	misc_pos	250
7	kids_pets	239
8	entertainment	233
9	personal_care	220
10	home	198
11	food_dining	151
12	grocery_net	134
13	health_fitness	133
14	travel	116

Gambar 4.8 Hasil Query Jumlah Fraud Berdasarkan Kategori Transaksi

Query ini bertujuan untuk mengidentifikasi kategori transaksi dengan jumlah fraud tertinggi. Dengan mengelompokkan transaksi fraud berdasarkan kategori dan menghitung jumlahnya, query ini membantu mengungkap jenis pembelanjaan yang paling sering dikaitkan dengan fraud. Hasil analisis menunjukkan bahwa kategori seperti grocery_pos, shopping_net, dan misc_net memiliki jumlah fraud yang relatif tinggi dibandingkan kategori lainnya. Informasi ini penting untuk analisis risiko berbasis kategori transaksi.

5. **Fraud Rate per Kategori Transaksi**

Query | Query History

```
1  SELECT
2      category_name,
3      SUM(is_fraud) * 1.0 / COUNT(*) AS fraud_rate
4  FROM dw.v_fact_with_category
5  GROUP BY category_name
6  ORDER BY fraud_rate DESC;
7
```

Gambar 4.9 Query Fraud Rate per Kategori Transaksi

Hasil:

	category_name text	fraud_rate numeric
1	shopping_net	0.01756148570374091426
2	misc_net	0.01445794554963894639
3	grocery_pos	0.01409760753166502208
4	shopping_pos	0.00722538398244651673
5	gas_transport	0.00469394420434607585
6	misc_pos	0.00313853493189379198
7	grocery_net	0.00294816509724544574
8	travel	0.00286370256992618560
9	entertainment	0.00247835428765928479
10	personal_care	0.00242402873575883118
11	kids_pets	0.00211438934843190162
12	food_dining	0.00165097691912399821
13	home	0.00160825244689924055
14	health_fitness	0.00154869059956450355

Gambar 4.10 Hasil Query Fraud Rate per Kategori Transaksi

Berbeda dengan query sebelumnya yang menghitung jumlah absolut fraud, query ini digunakan untuk menghitung fraud rate, yaitu rasio transaksi fraud terhadap total transaksi pada setiap kategori. Pendekatan ini memberikan perspektif yang lebih adil dalam membandingkan tingkat risiko antar kategori. Hasil analisis menunjukkan bahwa beberapa kategori dengan jumlah transaksi relatif kecil justru memiliki fraud rate yang lebih tinggi, sehingga perlu mendapatkan perhatian khusus dalam pemantauan risiko.

6. **Jam Terjadinya Fraud**

Query | Query History

```
1  SELECT
2      hour,
3      COUNT(*) AS fraud_count
4  FROM fact_credit_card_transactions
5  WHERE is_fraud = 1
6  GROUP BY hour
7  ORDER BY hour;
8
```

Gambar 4.11 Query Jam Terjadinya Fraud

Hasil:

	hour integer	fraud_count bigint
1	0	635
2	1	658
3	2	625
4	3	609
5	4	46
6	5	60
7	6	40
8	7	56
9	8	49
10	9	47
11	10	40
12	11	42
13	12	67
14	13	80
15	14	86
16	15	79
17	16	76
18	17	78
19	18	81
20	19	81
21	20	62
22	21	74
23	22	1931
24	23	1904

Gambar 4.12 Hasil Query Jam Terjadinya Fraud

Query ini digunakan untuk menganalisis distribusi jumlah transaksi fraud berdasarkan jam transaksi. Dengan mengelompokkan transaksi fraud berdasarkan atribut waktu (hour), query ini bertujuan untuk mengidentifikasi pola temporal kejadian fraud dalam satu hari. Hasil analisis menunjukkan adanya variasi jumlah fraud pada jam-jam tertentu, yang mengindikasikan bahwa waktu transaksi dapat menjadi faktor penting dalam analisis fraud.

7. **Fraud Rate Berdasarkan Jam Transaksi**

Query | Query History

```
1  SELECT
2      hour,
3      SUM(is_fraud) * 1.0 / COUNT(*) AS fraud_rate
4  FROM fact_credit_card_transactions
5  GROUP BY hour
6  ORDER BY hour;
7
```

Gambar 4.13 Query Fraud Rate Berdasarkan Jam Transaksi

Hasil:

	hour integer	fraud_rate numeric
1	0	0.01494047338948755353
2	1	0.01534908675266509599
3	2	0.01465210052513128282
4	3	0.01423928546377048797
5	4	0.00109882234909108282
6	5	0.00142277868677527211
7	6	0.00094562647754137116
8	7	0.00132691988721180959
9	8	0.00115280555228796612
10	9	0.00111414009719094465
11	10	0.00094627522414894372
12	11	0.00099805142341143482
13	12	0.00102670977826133595
14	13	0.00122485225219707873
15	14	0.00132542190028511983
16	15	0.00120811732501414568
17	16	0.00115631561330371542
18	17	0.00119174942704354469
19	18	0.00122632511241313530
20	19	0.00123649019967026928
21	20	0.00095241021229530861
22	21	0.00112920208139410679
23	22	0.02882864053029172016
24	23	0.02837386742966142108

Gambar 4.14 Hasil Query Fraud Rate Berdasarkan Jam Transaksi

Query ini menghitung fraud rate pada setiap jam transaksi dengan membandingkan jumlah transaksi fraud terhadap total transaksi pada jam yang sama. Tujuan dari query ini adalah untuk mengidentifikasi periode waktu dengan tingkat risiko fraud yang relatif lebih tinggi, terlepas dari volume transaksi. Hasil analisis menunjukkan bahwa fraud rate tidak selalu sejalan dengan jumlah transaksi, sehingga analisis berbasis rasio menjadi penting dalam memahami risiko fraud secara lebih akurat.

8. **Distribusi Fraud Berdasarkan Nominal Transaksi**

```sql
SELECT
    width_bucket(amt, 0, 200, 10) AS amount_bin,
    COUNT(*) AS fraud_count
FROM fact_credit_card_transactions
WHERE is_fraud = 1
GROUP BY amount_bin
ORDER BY amount_bin;
```

Gambar 4.15 Query Distribusi Fraud Berdasarkan Nominal Transaksi

Hasil:

	amount_bin integer	fraud_count bigint
1	1	1245
2	2	316
3	3	85
4	5	6
5	6	73
6	7	63
7	8	9
8	9	1
9	10	5708

Gambar 4.16 Hasil Query Distribusi Fraud Berdasarkan Nominal Transaksi

Query ini digunakan untuk menganalisis distribusi transaksi fraud berdasarkan rentang nominal transaksi. Dengan menggunakan teknik bucketization pada nilai transaksi, query ini mengelompokkan transaksi fraud ke dalam beberapa rentang nilai. Hasil analisis menunjukkan bahwa sebagian besar transaksi fraud terkonsentrasi pada rentang nominal tertentu, yang dapat menjadi indikasi pola nilai transaksi yang sering dimanfaatkan dalam aktivitas fraud.

9. **Persebaran Fraud Berdasarkan Lokasi**

```sql
SELECT
    state,
    city,
    COUNT(*) AS fraud_count
FROM fact_credit_card_transactions
WHERE is_fraud = 1
GROUP BY state, city
ORDER BY fraud_count DESC;
```

Gambar 4.17 Query Persebaran Fraud Berdasarkan Lokasi

Hasil:

	state text	city text	fraud_count bigint
1	TX	Houston	39
2	AL	Huntsville	29
3	FL	Naples	29
4	OK	Tulsa	27
5	TX	Dallas	27
6	KS	Topeka	27
7	MI	Detroit	26
8	TX	San Antonio	25
9	FL	Clearwater	24
10	NM	Albuquerque	24
11	PA	Beaver Falls	24
12	NY	New York City	23
13	CO	Aurora	23
14	MI	Warren	23
15	DC	Washington	21
16	WY	Fort Washakie	21
17	FL	Lakeland	21
18	PA	Fenelton	21
19	LA	Kenner	20
20	NY	Albany	20
21	MN	Hovland	19
22	MN	Minneapolis	19
23	NE	Hubbell	19
24	CA	San Jose	18
25	KS	Wichita	18
26	KY	Deane	18
27	CA	San Diego	18

Total rows: 722 Query complete 00:00:02.079

Gambar 4.18 Hasil Query Persebaran Fraud Berdasarkan Lokasi

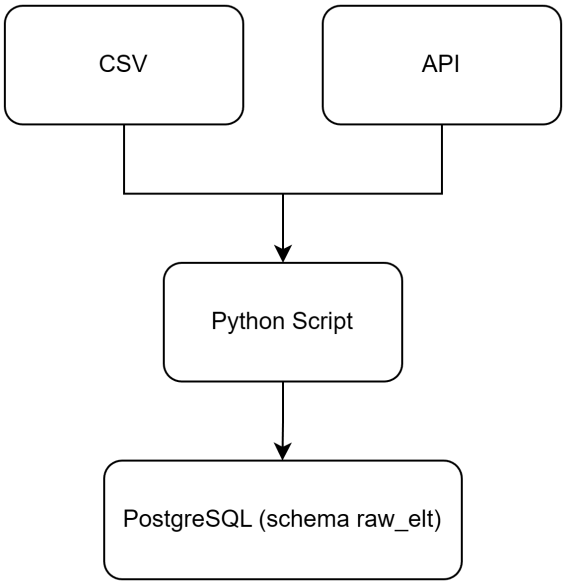
Query ini bertujuan untuk menganalisis persebaran transaksi fraud berdasarkan lokasi geografis, yaitu negara bagian dan kota. Dengan mengelompokkan transaksi fraud berdasarkan atribut lokasi, query ini membantu mengidentifikasi wilayah dengan konsentrasi fraud yang lebih tinggi. Hasil analisis menunjukkan bahwa beberapa kota dan negara bagian memiliki jumlah transaksi fraud yang relatif lebih besar, yang dapat menjadi dasar analisis risiko berbasis lokasi.

BAB V. PIPELINE ELT

5.1 Extract & Load

Tahap Extract & Load merupakan tahap awal dalam pipeline ELT yang bertujuan untuk memindahkan data dari sumber ke dalam data warehouse tanpa melakukan transformasi terlebih dahulu. Pada penelitian ini, data yang digunakan terdiri dari dua sumber utama. Dataset pertama adalah data transaksi kartu kredit yang diperoleh dalam format CSV. Dataset kedua adalah data suku bunga Federal Funds Rate yang diperoleh melalui API FRED. Kedua dataset tersebut diekstraksi menggunakan script Python.

Proses extract dilakukan dengan membaca file CSV dan memanggil API eksternal tanpa melakukan preprocessing, pembersihan, maupun perubahan struktur data. Selanjutnya, data yang telah diekstraksi langsung dimuat ke dalam database PostgreSQL pada schema raw_elt. Data transaksi disimpan pada tabel raw_elt.cc_tx, sedangkan data suku bunga disimpan pada tabel raw_elt.fred_fedfunds. Pendekatan ini mengikuti konsep ELT (Extract Load Transform), di mana data mentah disimpan terlebih dahulu di dalam data warehouse untuk menjaga keutuhan data sebelum dilakukan transformasi pada tahap selanjutnya. Untuk memastikan proses load berjalan dengan baik, dilakukan verifikasi dengan menghitung jumlah baris pada masing-masing tabel raw serta menampilkan sebagian isi data.



Gambar 5.1 Diagram Extract dan Load

```sql
SELECT 'cc_tx' AS table_name, COUNT(*) AS total_rows FROM raw_elt.cc_tx
UNION ALL
SELECT 'fred_fedfunds', COUNT(*) FROM raw_elt.fred_fedfunds;
```

Gambar 5.2 Query Verifikasi Jumlah Data pada Tabel Raw

	table_name text	total_rows bigint
1	cc_tx	1296675
2	fred_fedfunds	858

Gambar 5.3 Hasil verifikasi jumlah data

5.2 Transform di Warehouse

5.2.1 Tujuan Transformasi

Tahap transformasi dilakukan setelah data mentah berhasil dimuat ke dalam data warehouse pada skema raw_elt. Transformasi ini bertujuan untuk menyiapkan data agar siap digunakan pada tahap analisis dan visualisasi dengan cara melakukan penyesuaian tipe data, pembersihan nilai tidak valid, serta penyeragaman format waktu. Pendekatan ini menggunkan konsep ELT (Extract Load Transform), di mana seluruh proses transformasi dilakukan langsung di dalam database PostgreSQL menggunakan SQL.

5.2.2 Transformasi Data Transaksi

Transformasi data transaksi dilakukan dengan membuat tabel staging.cc_tx_casted. Pada tahap ini, data transaksi yang masih bertipe teks pada tabel raw_elt.cc_tx diubah ke tipe data yang sesuai. Transformasi yang dilakukan yaitu:

- Casting tipe data numerik (amt)
- Casting label fraud (is_fraud)
- Konversi waktu transaksi dari format Unix Time ke timestamp
- Casting koordinat geografis (latitude dan longitude)
- Pembersihan data dengan menghilangkan nilai kosong (NULL)

```sql
CREATE SCHEMA IF NOT EXISTS staging;

DROP TABLE IF EXISTS staging.cc_tx_casted;

CREATE TABLE staging.cc_tx_casted AS
SELECT
    trans_num,
    cc_num,
    merchant,
    category,

    amt::NUMERIC(12,2) AS amount,
    is_fraud::INT AS is_fraud,

    to_timestamp(unix_time::BIGINT) AT TIME ZONE 'UTC' AS transaction_ts,

    city,
    state,
    zip,
    merch_zipcode,

    lat::DOUBLE PRECISION AS cust_lat,
    long::DOUBLE PRECISION AS cust_long,
    merch_lat::DOUBLE PRECISION AS merch_lat,
    merch_long::DOUBLE PRECISION AS merch_long,

    city_pop::INT AS city_pop

FROM raw_elt.cc_tx
WHERE amt IS NOT NULL;
```

Gambar 5.4 Query Transformasi Data Transaksi ke Tabel Staging

Transformasi dilakukan sepenuhnya menggunakan perintah SQL seperti gambar 5.4.

5.2.3 Transformasi Data Suku Bunga

Tahap ini menyiapkan data suku bunga Federal Funds Rate (FEDFUNDS) yang sebelumnya sudah dimuat pada tabel raw_elt.fred_fedfunds. Karena data raw masih bertipe teks, dilakukan transformasi di dalam database PostgreSQL menggunakan SQL untuk:

- mengubah kolom tanggal observasi menjadi tipe DATE
- mengubah nilai suku bunga menjadi tipe NUMERIC
- memastikan data siap dipakai analisis / join (kalau nanti dibutuhkan)

Hasil transformasi disimpan pada tabel staging.fred_fedfunds_casted.

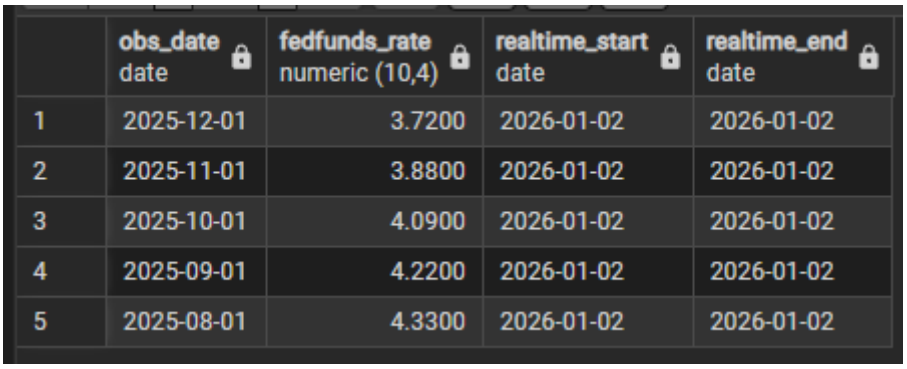
Query Query History

```sql
SELECT COUNT(*) FROM staging.fred_fedfunds_casted;

SELECT *
FROM staging.fred_fedfunds_casted
ORDER BY obs_date ASC
LIMIT 5;

SELECT *
FROM staging.fred_fedfunds_casted
ORDER BY obs_date DESC
LIMIT 5;
```

Gambar 5.5 Query Transformasi Data Suku Bunga (FRED) ke Tabel staging.fred_fedfunds_casted



	obs_date date	fedfunds_rate numeric (10,4)	realtime_start date	realtime_end date
1	2025-12-01	3.7200	2026-01-02	2026-01-02
2	2025-11-01	3.8800	2026-01-02	2026-01-02
3	2025-10-01	4.0900	2026-01-02	2026-01-02
4	2025-09-01	4.2200	2026-01-02	2026-01-02
5	2025-08-01	4.3300	2026-01-02	2026-01-02

Gambar 5.6 Hasil Verifikasi Transformasi Data Suku Bunga pada Tabel staging.fred_fedfunds_casted

Hasil transformasi memperlihatkan data suku bunga Federal Funds Rate berhasil dikonversi ke tipe data numerik dan tanggal yang sesuai serta tersimpan pada tabel staging. Data ini siap digunakan pada tahap analisis lanjutan atau integrasi dengan data transaksi apabila diperlukan.

5.2.4 Data Enrichment dan Feature Engineering

Tahap ini merupakan tahap lanjutan transformasi pada pipeline ELT yang bertujuan untuk memperkaya data transaksi dengan atribut tambahan dan membangun fitur baru yang siap digunakan pada tahap analisis. Pada tahap ini dilakukan proses data enrichment dengan menggabungkan data transaksi kartu kredit dari tabel staging.cc_tx_casted dengan data suku bunga Federal Funds Rate dari tabel staging.fred_fedfunds_casted. Proses penggabungan dilakukan menggunakan perintah SQL LEFT JOIN berdasarkan kesesuaian periode waktu (tahun dan bulan transaksi).

Dilakukan feature engineering yaitu transformasi logaritmik pada nilai transaksi (amount_log) untuk menormalkan distribusi data numerik dan mempermudah analisis lanjutan. Hasil dari proses disimpan pada tabel fakta dw.elt_fact_transactions sebagai data utama pada data warehouse untuk keperluan analisis dan visualisasi.

```sql
SELECT COUNT(*) FROM dw.elt_fact_transactions;

SELECT
  COUNT(*) AS total_rows,
  COUNT(interest_rate) AS rows_with_rate,
  COUNT(*) - COUNT(interest_rate) AS rows_rate_null
FROM dw.elt_fact_transactions;

SELECT *
FROM dw.elt_fact_transactions
ORDER BY transaction_ts DESC
LIMIT 5;
```

Gambar 5.7 Query Verifikasi Tabel Fakta

	trans_num text	category text	amount numeric	amount_log numeric	interest_ numeric	is_fraud integer	transaction_ts timestamp without time
1	8f7c8e4ab7f25875d753b422917c98c9	food_dining	4.30	1.6677068205580762	0.0900	0	2013-06-21 12:13:37
2	d667cdcbadaaed3da3f4020e83591c83	food_dining	74.90	4.3294166844015842	0.0900	0	2013-06-21 12:13:36
3	483f52fe67fabef353d552c1e662974c	food_dining	105.93	4.6721744147685652	0.0900	0	2013-06-21 12:12:32
4	278000d2e0d2277d1de2f890067dcc0a	food_dining	51.70	3.9646154555473166	0.0900	0	2013-06-21 12:12:19
5	440b587732da4dc1a6395aba5fb41669	entertainment	15.56	2.8069901489571136	0.0900	0	2013-06-21 12:12:08

Gambar 5.8 Hasil Verifikasi Sampel Data Tabel Fakta

BAB VI. DASHBOARD ANALITIK

6.1 Tools & Koneksi

Untuk kebutuhan visualisasi dan analisis, pengembangan dashboard analitik pada proyek ini menggunakan perangkat lunak dan konfigurasi koneksi sebagai berikut:

1. Tools Dashboard

Dashboard analitik dibangun menggunakan Microsoft Power BI Desktop. Perangkat lunak ini dipilih karena memiliki fitur visualisasi yang lengkap, antarmuka yang intuitif, dan kompatibilitas yang baik dengan berbagai sumber data, termasuk basis data relasional. Power BI memungkinkan pembuatan metrik kustom (DAX), filter interaktif, dan navigasi *drill-down* yang diperlukan untuk menganalisis pola fraud secara mendalam.

2. Koneksi ke Data Warehouse

Dashboard terhubung secara langsung dengan data warehouse yang berada di PostgreSQL (skema dw). Koneksi dilakukan menggunakan *connector* bawaan PostgreSQL pada Power BI dengan mode konektivitas Import Mode (atau *DirectQuery* jika kamu menggunakannya).

- Host: localhost
- Database: etl_warehouse
- Tabel Sumber: Tabel fakta dw.elt_fact_transactions dan view lainnya.

Mekanisme koneksi ini memastikan dashboard selalu menampilkan data yang bersumber dari hasil transformasi pipeline ETL atau ELT yang telah divalidasi, menjaga konsistensi antara data di *warehouse* dan visualisasi di dashboard.

6.2 Desain Dashboard

Dashboard analitik pada proyek ini dibangun menggunakan Microsoft Power BI sebagai alat visualisasi utama. Power BI dipilih karena kemampuannya dalam mengintegrasikan data warehouse PostgreSQL, mendukung visualisasi interaktif, serta menyediakan fitur filter dan drill-down yang memudahkan eksplorasi data fraud secara multidimensi.

Struktur dashboard dirancang dengan pendekatan top-down analytics, dimulai dari ringkasan metrik utama (Key Performance Indicator/KPI), kemudian dilanjutkan dengan visualisasi analitik yang lebih detail. Secara umum, struktur dashboard terdiri dari:

1. Bagian KPI
   Menampilkan metrik ringkasan seperti total transaksi, total transaksi fraud, dan fraud rate secara keseluruhan.
2. Analisis Komparatif
   Visualisasi perbandingan fraud berdasarkan kategori transaksi dan nominal transaksi.
3. Analisis Temporal
   Visualisasi pola waktu terjadinya fraud berdasarkan jam transaksi.

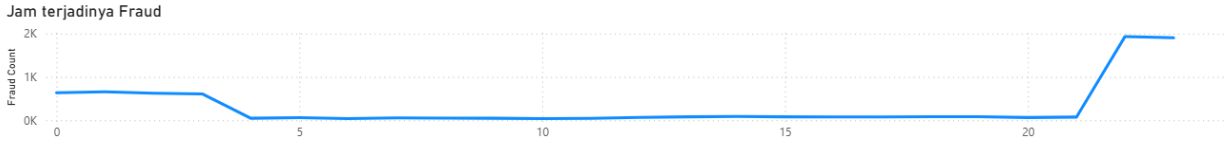
4. Analisis Risiko
   Scatter plot (risk matrix) yang menghubungkan jumlah transaksi dan tingkat fraud per kategori.
5. Analisis Spasial
   Peta interaktif untuk menunjukkan persebaran fraud berdasarkan lokasi geografis (kota dan negara bagian).

Struktur ini dirancang agar pengguna dapat dengan cepat memahami kondisi umum fraud, kemudian melakukan eksplorasi lebih lanjut terhadap faktor penyebabnya.

a. Grafik Tren

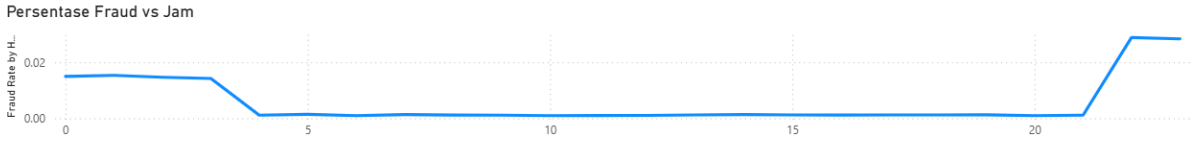
Grafik tren digunakan untuk menganalisis pola waktu terjadinya fraud, khususnya berdasarkan jam transaksi.
Visualisasi ini ditampilkan dalam bentuk line chart yang menunjukkan:

- Jumlah transaksi fraud per jam



Gambar 6.1 Tren Jumlah Fraud Berdasarkan Jam Transaksi

- Persentase fraud per jam



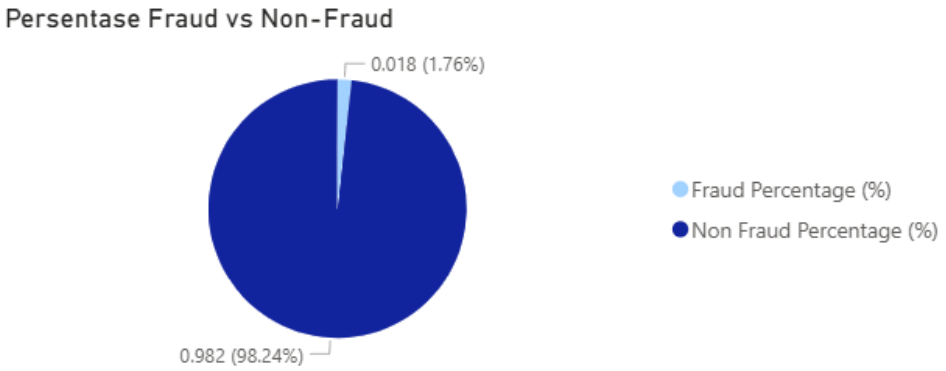
Gambar 6.2 Tren Persentase Fraud Berdasarkan Jam Transaksi

Grafik ini membantu mengidentifikasi jam-jam dengan risiko fraud tertinggi dalam satu hari.

b. Grafik Perbandingan

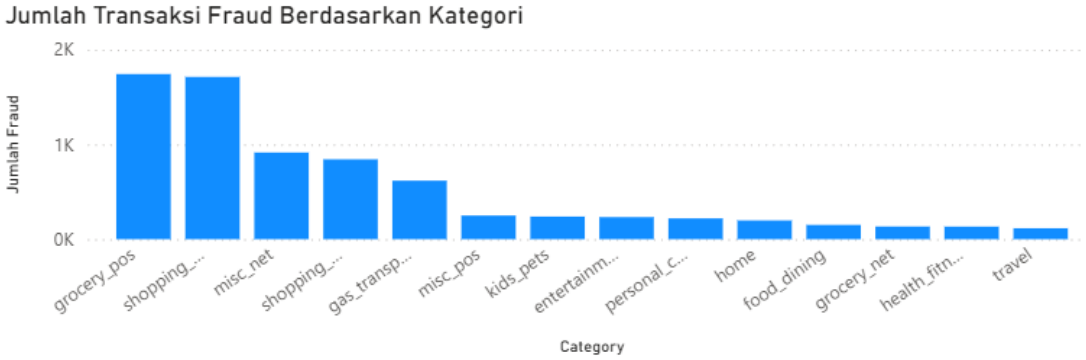
Beberapa grafik perbandingan digunakan untuk menganalisis perbedaan risiko fraud, antara lain:

- Pie chart persentase fraud vs non-fraud untuk melihat ketimpangan kelas transaksi.



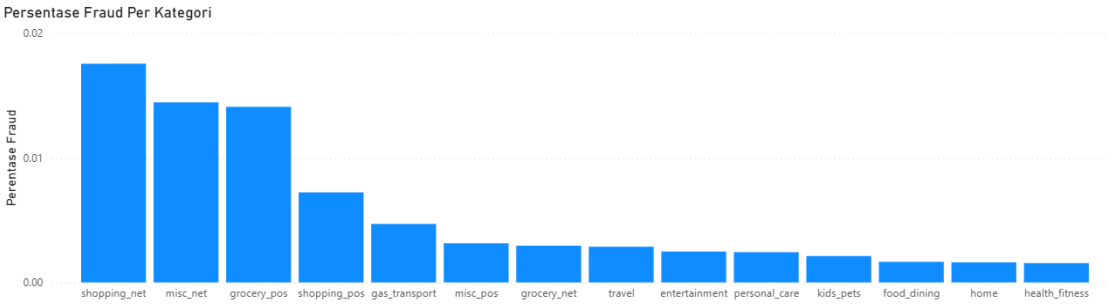
Gambar 6.3 Proporsi Transaksi Fraud vs Non-Fraud

- Bar chart jumlah transaksi fraud berdasarkan kategori transaksi.



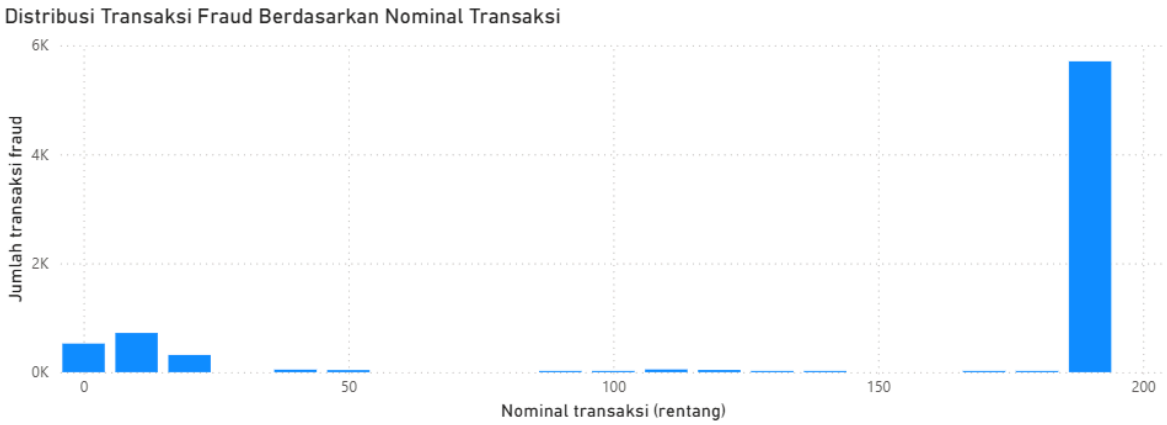
Gambar 6.4 Jumlah Transaksi Fraud per Kategori

- Bar chart persentase fraud per kategori untuk mengukur tingkat risiko relatif setiap kategori.



Gambar 6.5 Tingkat Risiko (Fraud Rate) per Kategori

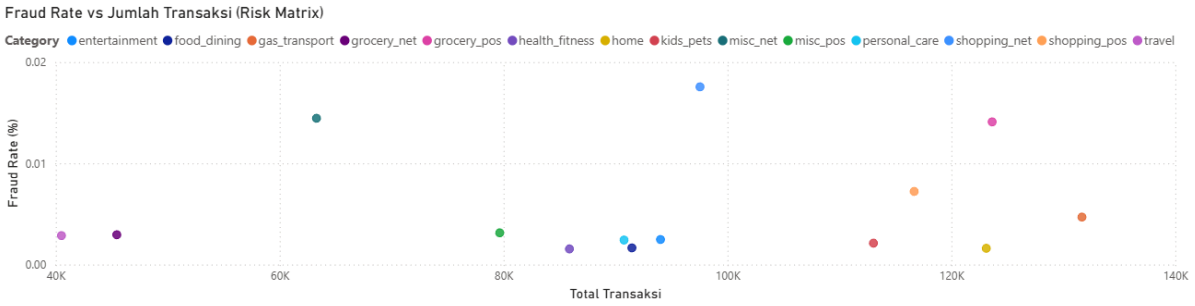
- Histogram/bin chart distribusi transaksi fraud berdasarkan rentang nominal transaksi.



Gambar 6.6 Distribusi Fraud Berdasarkan Rentang Nominal Transaksi

Grafik-grafik ini memudahkan identifikasi kategori dan nominal transaksi yang paling berisiko terhadap fraud.

c. Risk Matrix (Scatter Plot)
Scatter plot digunakan untuk memvisualisasikan hubungan antara:

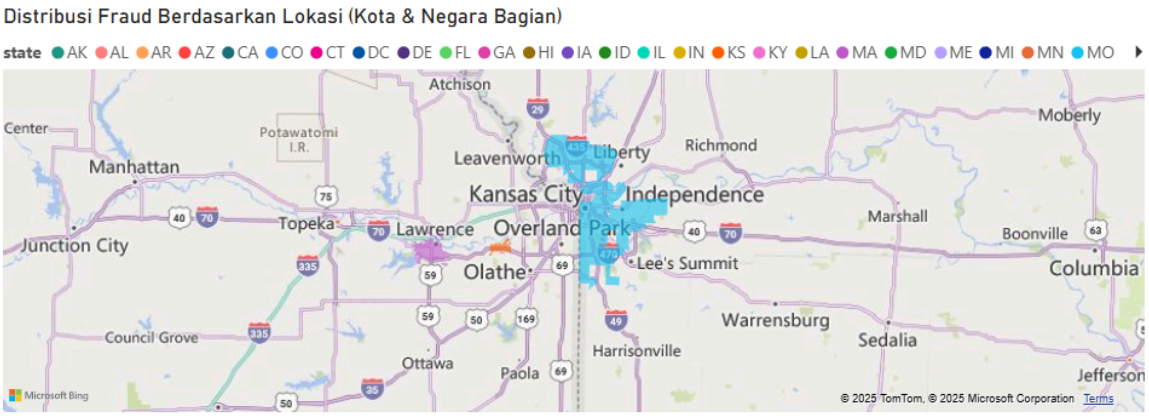


Gambar 6.7 Risk Matrix: Hubungan Jumlah Transaksi dan Fraud Rate

- Jumlah transaksi per kategori (sumbu X)
- Fraud rate per kategori (sumbu Y)

Setiap titik merepresentasikan satu kategori transaksi. Visualisasi ini berfungsi sebagai risk matrix, yang membantu mengelompokkan kategori ke dalam risiko rendah, sedang, dan tinggi berdasarkan kombinasi volume dan tingkat fraud.

d. Visualisasi Spasial
Peta geografis digunakan untuk menampilkan distribusi fraud berdasarkan lokasi (kota dan negara bagian).



Gambar 6.8 Peta Persebaran Fraud Berdasarkan Lokasi Geografis

e. Filter dan Interaktivitas

Dashboard dilengkapi dengan filter (slicer) yang memungkinkan pengguna melakukan eksplorasi data berdasarkan:

- Kategori transaksi
- Lokasi (kota dan state)
- Rentang waktu

Interaksi ini memungkinkan analisis fraud yang lebih fleksibel dan kontekstual.

BAB VII. ANALISIS KOMPARATIF ETL VS ELT

7.1 Tabel Perbandingan ETL dan ELT

Aspek	ETL	ELT
Kompleksitas Implementasi	Tinggi, memerlukan penulisan kode script yang panjang, setup environment Python, dan manajemen tipe data manual.	Sedang, menggunakan query SQL standar yaitu CREATE TABLE AS SELECT, kode ringkas dan mudah dibaca.
Fleksibilitas Perubahan	Rendah, llogika berubah, harus edit script python dan me load ulang seluruh data dari awal yang artinya berat.	Tinggi, saat logika berubah, DROP TABLE dan jalankan ulang query transformasi dalam hitungan detik.
Skalabilitas	Terbatas pada RAM. Pandas memuat seluruh data ke memori, jika data lebih dari RAM laptop, proses akan crash.	Tinggi, postgreSQL bisa mengolah data yang melebihi kapasitas RAM dengan manajemen disk based yang efisien.
Kemudahan Eksplorasi Data	Kurang, harus print dataframe (df.head()) berulang kali untuk melihat hasil perubahan.	Sangat baik, bisa langsung cek hasil transformasi parsial menggunakan query SELECT.
Beban Komputasi	Berat di python, memakan resource ram dan cpu laptop saat proses berjalan.	Berat di Server database, komputasi dipindahkan ke PostgreSQL, laptop akan ringan.
Konsistensi dan Kualitas Data	Lebih Aman, data kotor terdeteksi dan ditolak sebelum masuk ke database.	Lebih Berisiko, data raw masuk ke warehouse, perlu filter tambahan di query supaya hasilnya bersih.
Waktu Time-to-Analysis	Lambat, menunggu seluruh proses cleaning selesai baru data tersedia di warehouse.	Cepat, data mentah langsung tersedia di warehouse dan bisa langsung di query meskipun belum bersih.
Kesesuaian dengan Studi Kasus	Sangat bagus untuk feature engineering	Sangat efektif untuk analisis cepat sejuta data transaksi.

	kompleks di awal, dengan cara seperti pembersihan teks nama merchant yang kotor sebelum masuk database.	Memudahkan penggabungan (JOIN) data transaksi dengan data FRED secara instan di sisi database.

Tabel 7.1 Tabel Perbandingan ETL dan ELT

7.2 Analisis dan Refleksi

Berdasarkan implementasi pipeline ETL dan ELT pada dataset transaksi kartu kredit dengan data jumlah 1.296.675 baris, ditemukan perbedaan kinerja dan karakteristik yang terlihat antara kedua pendekatan. Analisis ini didasarkan pada metrik waktu eksekusi, penggunaan sumber daya komputasi, dan fleksibilitas dalam pengelolaan data multisumber (CSV dan API FRED).

1. Keunggulan Performa ELT

   Pendekatan ELT terlihat jauh lebih efisien dari sisi waktu eksekusi. Berdasarkan log sistem, proses loading data mentah ke data warehouse hanya memakan waktu sekitar 109,28 detik. Transformasi lanjutan seperti penggabungan data transaksi dengan data suku bunga FRED dan perhitungan amount_log) yang dilakukan menggunakan SQL berjalan cepat karena memanfaatkan optimasi query yang ada di PostgreSQL. Sebaliknya, pendekatan ETL membutuhkan waktu lebih lama karena proses pembacaan file ke dalam RAM dan penulisan ulang data oleh Python pakai library Pandas membuat overhead.

2. Keunggulan Kontrol ETL

   ETL memberikan kontrol yang lebih preventif. Pada tahap transformasi Python, pembersihan data teks yang kompleks seperti standardisasi nama merchant yang tidak konsisten bsia dilakukan dengan lebih presisi menggunakan library manipulasi string. Maka dari itu memastikan bahwa data yang masuk ke data warehouse sudah pada kondisi bersih, berbeda dengan ELT di mana data yang kotor masuk duluan ke lapisan staging.

Kesesuaian dengan Studi Kasus Dalam konteks studi kasus deteksi fraud ini, pendekatan ELT dinilai lebih sesuai untuk menangani arsitektur data secara keseluruhan, dengan catatan tertentu.

- Karakteristik data yang melibatkan penggabungan (join) antara tabel fakta transaksi yang besar dengan data dimensi eksternal pakai API FRED jauh lebih efisien dilakukan di sisi database pakai ELT. Melakukan operasi join sebesar ini di memori laptop menggunakan Python dapat menyebabkan memory overflow.

- Untuk penanganan fitur spesifik yang membutuhkan pemrosesan teks rumit seperti parsing detail lokasi dari string, logika prosedural Python pada ETL tetap lebih baik dibandingkan fungsi standar SQL.

Refleksi dan Implikasi masa mendatang, eksperimen ini juga menggaris bawahi keterbatasan skalabilitas dari pendekatan ETL berbasis in memory (Pandas). Jika volume data transaksi meningkat dari jutaan menjadi miliaran baris, pendekatan ETL konvensional yang dijalankan pada single node akan mengalami kegagalan memori. Sebagai implikasi untuk pengembangan sistem di masa mendatang, arsitektur ideal yang disarankan adalah Hybrid Pipeline, yaitu menggunakan pemrosesan terdistribusi seperti Apache Spark untuk ETL awal guna menangani pembersihan data yang kompleks, kemudian beralih ke ELT menggunakan Cloud Data Warehouse seperti Google BigQuery atau Snowflake untuk proses transformasi analitik, agregasi, dan penyajian data ke dashboard. Pendekatan ini akan menyeimbangkan kebutuhan akan kualitas data yang tinggi dan performa komputasi yang skalabel.

7.3 Ringkasan Temuan

Berdasarkan analisis komparatif yang dilakukan, dapat disimpulkan beberapa temuan:

1. **Efisiensi Waktu dan Komputasi**

   Pendekatan ELT menunjukkan performa yang lebih baik dalam menangani dataset berskala besar. Proses transformasi berbasis SQL di dalam data warehouse meminimalkan pergerakan data, menghasilkan waktu eksekusi yang lebih singkat dibandingkan ETL yang memproses seluruh data di RAM.

2. **Fleksibilitas Pengolahan**

   ELT memeberikan fleksibilitas yang lebih tinggi untuk eksplorasi data dan perubahan logika transformasi secara cepat tanpa perlu mengulang proses ekstraksi. Tapi, ETL memiliki keunggulan dalam penanganan kasus pembersihan data yang kompleks, khususnya manipulasi string dan regex yang sulit dilakukan dengan SQL standar.

3. **Skalabilitas**

   Arsitektur ELT lebih skalabel untuk jangka panjang karena memanfaatkan optimasi basis data (database engine) PostgreSQL, sedangkan pendekatan ETL berbasis Python (Pandas) memiliki keterbatasan pada kapasitas memori perangkat lokal.

BAB VIII. KESIMPULAN & SARAN

8.1 Kesimpulan

Implementasi pipeline Big Data pada proyek ini berhasil mengelola dataset transaksi kartu kredit sebanyak 1.296.675 baris, di mana pendekatan ELT (Extract-Load-Transform) terbukti memiliki kinerja yang lebih unggul dibandingkan ETL dengan waktu eksekusi pemuatan data mentah hanya 109,28 detik. Berdasarkan hasil analisis dashboard, ditemukan bahwa meskipun transaksi fraud hanya mencakup 0,58% dari total populasi, aktivitas ilegal ini memiliki pola spesifik yang terkonsentrasi pada kategori grocery_pos dengan volume tertinggi dan shopping_net dengan tingkat risiko terbesar, serta menunjukkan anomali lonjakan pada jam-jam larut malam antara pukul 22:00 hingga 23:00. Secara keseluruhan, pendekatan ELT disimpulkan sebagai metode yang lebih efektif untuk studi kasus ini karena kemampuannya melakukan integrasi data multisumber secara instan di dalam data warehouse tanpa membebani memori, sementara pendekatan ETL tetap relevan digunakan secara terbatas untuk kebutuhan pembersihan data tekstual yang kompleks di tahap awal.

8.2 Saran

Untuk pengembangan sistem di masa mendatang, disarankan agar pipeline yang saat ini berbasis historis ditingkatkan kemampuannya menjadi pemrosesan real-time menggunakan teknologi streaming serta diintegrasikan dengan model Machine Learning untuk memprediksi dan mendeteksi fraud secara otomatis saat transaksi baru terjadi. Selain itu, kinerja dan manajemen sistem dapat dioptimalkan lebih lanjut dengan menerapkan orkestrasi otomatis menggunakan Apache Airflow untuk menjadwalkan proses ETL/ELT secara berkala tanpa intervensi manual, serta mengimplementasikan teknik partisi tabel pada database PostgreSQL untuk menjaga stabilitas performa query dashboard seiring dengan pertumbuhan volume data yang terus meningkat.

BAB IX PEMBAGIAN TUGAS
DAN KONTRIBUSI INDIVIDU TIM

Mahasiswa 1

Komponen	Uraian
Nama Mahasiswa	Alfikri
NIM	1103223015
Persentase Kontribusi	50 %
Tanggung Jawab Utama	Perancangan ETL, menghubungkan dengan PostgreSQL dan analitikal dashboard menggunakan Power BI
Tanggung Jawab Tambahan	Membuat laporan tugas besar dari abstrak hingga bab 5
Kendala yang Dihadapi	Proses load data menuju PostgreSQL membutuhkan waktu yang lama, dikarenakan dataset berjumlah lebih dari 1 juta baris
Solusi yang Dilakukan	Proses dilakukan pada jauh hari, sehingga tidak mengganggu timeline pengerjaan

Mahasiswa 2

Komponen	Uraian
Nama Mahasiswa	Raihan Abdul Majid
NIM	1103223091
Persentase Kontribusi	50 %
Tanggung Jawab Utama	Perancangan dan implementasi pipeline ELT, pembuatan analisis dan kesimpulan pada dokumen
Tanggung Jawab Tambahan	Penyusunan query SQL untuk data cleaning, casting tipe data, data enrichment, dan feature engineering, Verifikasi hasil load dan transformasi data, penyusunan dokumentasi dan penulisan laporan pipeline ELT
Kendala yang Dihadapi	error pada power bi yang tidak mau konek dengan posgresqlnya, permasalahan autentikasi dan konfigurasi koneksi database.
Solusi yang Dilakukan	Melakukan pengecekan ulang konfigurasi PostgreSQL, memverifikasi koneksi database melalui pgAdmin dan Python.

DAFTAR PUSTAKA

[1] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani, and S. Ullah Khan, "The rise of 'big data' on cloud computing: Review and open research issues," *Inf Syst*, vol. 47, pp. 98–115, Jan. 2015, doi: 10.1016/J.IS.2014.07.006.

[2] C. Whitrow, D. J. Hand, P. Juszczak, D. Weston, and N. M. Adams, "Transaction aggregation as a strategy for credit card fraud detection," Data Min Knowl Discov, vol. 18, no. 1, pp. 30–55, Feb. 2009, doi: 10.1007/S10618-008-0116-Z.

[3] "Data Warehouse Design and Management: Theory and Practice." Accessed: Jan. 06, 2026. [Online]. Available: https://www.researchgate.net/publication/237138345_Data_Warehouse_Design_and_Management_Theory_and_Practice

[4] M. Golfarelli and S. Rizzi, *Data Warehouse Design: Modern Principles and Methodologies*. New York, NY, USA: McGraw-Hill, 2009.

[5] J. Vassiliadis, "A survey of Extract–Transform–Load technology," *International Journal of Data Warehousing and Mining*, vol. 5, no. 3, pp. 1–27, 2009.

[6] A. L. Labrinidis and H. V. Jagadish, "Challenges and opportunities with big data," *Proceedings of the VLDB Endowment*, vol. 5, no. 12, pp. 2032–2033, Aug. 2012.

[7] Federal Reserve Bank of St. Louis, "Federal Reserve Economic Data (FRED) API Documentation." [Online]. Available: https://api.stlouisfed.org/fred/series/observations. Accessed: Jan. 6, 2026.

[8] PostgreSQL Global Development Group, *PostgreSQL Documentation*. [Online]. Available: https://www.postgresql.org/docs/.

[9] Microsoft, *Power BI Documentation*. [Online]. Available: https://learn.microsoft.com/power-bi/.

[10] T. Dasu and T. Johnson, *Exploratory Data Mining and Data Cleaning*. Hoboken, NJ, USA: Wiley, 2003.

LAMPIRAN

Lampiran A. Repositori Kode dan Pipeline

Tautan Repositori GitHub / GitLab:
👉 https://github.com/Alfikriangelo/Tubes-UAS-Big-Data/tree/main

Lampiran B. Dataset dan Sumber Data

Tautan Dataset Utama:
👉
https://drive.google.com/file/d/1VV6aM4OtAaIT2zcqTdGEM-0ilwe0G3rf/view?usp=sharing

Tautan Dataset Pendukung / API:
👉 https://api.stlouisfed.org/fred/series/observations

Lampiran C. Diagram Arsitektur Sistem

Tautan Diagram Arsitektur (PDF/PNG):
👉
https://drive.google.com/file/d/1l3tfNhhNF3goOA0I1Kc6mQlRGYc5pOa5/view?usp=sharing

Lampiran D. Skema Data Warehouse dan Query SQL

Tautan Skema Warehouse & Query SQL:
👉 https://github.com/Alfikriangelo/Tubes-UAS-Big-Data/tree/main/warehouse

Lampiran E. Dashboard Analitik

Tautan Dashboard Analitik:
👉
https://telkomuniversityofficial-my.sharepoint.com/:u:/g/personal/alfikrii_student_telkomuniversity_ac_id/IQCmlIEF_1UJQbzhY4u8ge-QASeWFnvFwesAwhE2PPn7X5o?e=EFyCe7

Lampiran F. Log Eksekusi dan Metadata Proses

Tautan Log Proses ETL & ELT:
👉 https://github.com/Alfikriangelo/Tubes-UAS-Big-Data/tree/main/logs