

BAB 3

Pelaksanaan Kerja Magang

3.1 Kedudukan dan Koordinasi

Selama periode magang, posisi magang ditempatkan di Divisi *Data Governance and Quality Assurance* pada BTPN Syariah. Divisi ini bertanggung jawab atas pengelolaan kualitas dan integritas data nasabah, serta menyediakan dukungan terhadap unit lain melalui penyediaan data yang telah tervalidasi secara sistematis.

Selama kegiatan berlangsung, komunikasi dan koordinasi dilakukan secara langsung dengan supervisor, anggota tim data, serta stakeholder terkait lainnya. Arahan diberikan secara berkala, dan pelaksanaan tugas mengikuti prosedur kerja yang berlaku di divisi. Semua proses pelaporan dan penyelesaian pekerjaan dilakukan dalam sistem hybrid, terdiri atas kombinasi *Work From Office* (WFO) dan *Work From Home* (WFH).

3.2 Tugas yang Dilakukan

Ruang lingkup pekerjaan selama magang difokuskan pada pengembangan dan penerapan sistem validasi data serta pembangunan dashboard visualisasi berbasis Tableau. Kegiatan utama yang dilaksanakan mencakup:

- Pembelajaran dan penggunaan Amazon Redshift sebagai sistem basis data utama.
- Koneksi dan integrasi Tableau dengan Amazon Redshift untuk menampilkan data secara interaktif.
- Penyusunan *query* SQL untuk validasi atribut seperti `nama_lengkap`, `no_identitas`, `tanggal_lahir`, dan atribut lainnya.
- Implementasi logika klasifikasi validitas dan kelengkapan data menggunakan struktur `CASE WHEN`.
- Desain dan pembuatan grafik interaktif seperti *pie chart*, *bar chart*, dan *stacked chart* untuk menyajikan metrik *validity* dan *completeness*.

- Pengembangan fitur filtering dan *drill-down* untuk keperluan analisis lebih dalam.
- Dokumentasi teknis yang mencakup struktur *query*, koneksi Tableau–Redshift, serta konfigurasi dashboard.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan magang dilaksanakan selama 640 jam dari 3 Maret 2025 hingga 30 Juni 2025. Seluruh kegiatan dirancang dan dijalankan berdasarkan tahapan berikut:

1. Fase onboarding dan pengenalan sistem internal.
2. Pemetaan atribut data yang relevan untuk validasi.
3. Penyusunan dan pengujian *query* SQL untuk deteksi kesalahan.
4. Normalisasi dan pemrosesan data untuk konektivitas Tableau.
5. Desain visualisasi interaktif berbasis metrik kualitas data.
6. Evaluasi kinerja dashboard dan revisi sesuai masukan supervisor.
7. Finalisasi dokumentasi dan persiapan pelaporan.

3.3.1 Rincian Kegiatan Magang Mingguan

Aktivitas mingguan terdokumentasi dalam tabel aktivitas mingguan, mencakup tanggal pelaksanaan, aktivitas inti, hasil, kendala, dan rencana tindak lanjut.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

Tabel 3.1. Aktivitas dan capaian mingguan (Minggu 1–8)

Minggu	Aktivitas dan Capaian Mingguan
1	Orientasi awal terhadap lingkungan kerja dan teknologi yang digunakan di divisi Data Governance and Quality Assurance. Kegiatan berfokus pada pembelajaran mandiri mengenai struktur database Amazon Redshift.
2	Eksplorasi aplikasi Tableau sebagai alat visualisasi utama. Langkah teknis dilakukan untuk menghubungkan Tableau ke Amazon Redshift melalui konfigurasi ODBC dan pengujian koneksi database. Selain itu, query sederhana mulai dijalankan untuk menampilkan data ke Tableau sebagai latihan awal.
3	Analisis struktur data nasabah dan identifikasi kebutuhan validasi berdasarkan pedoman pelaporan SLIK OJK. Penyusunan query SQL dilakukan untuk memberikan flag pada entri yang tidak sesuai, menggunakan logika CASE WHEN.
4	Restrukturisasi dataset untuk mendukung skema visualisasi di Tableau. Dataset awal yang terlalu kompleks dipecah menjadi dua tabel terpisah agar lebih ringan dan mudah digunakan. Proses ini mencakup pembersihan data serta penyesuaian skema field agar kompatibel dengan format long table Tableau.
5	Implementasi ulang query langsung di dalam Tableau menggunakan fitur Custom SQL. Tujuannya adalah untuk menghasilkan dataset yang siap digunakan secara dinamis dalam berbagai visualisasi seperti pie chart, stacked bar chart, dan line chart.
6	Penambahan logika validasi lanjutan untuk mengklasifikasikan status data pada tingkat baris. Flag tambahan dibuat untuk mengidentifikasi jenis error spesifik yang disesuaikan dengan pedoman SLIK.
7	Proses desain dan pembuatan visualisasi metrik <i>validity</i> menggunakan pie chart dan horizontal bar chart. Visualisasi ini digunakan untuk menunjukkan distribusi data valid, tidak valid, serta jenis kesalahan dominan yang ditemukan dalam dataset.
8	Pengembangan metrik visual untuk <i>completeness</i> , yaitu mengukur jumlah field yang kosong (null) pada setiap atribut penting. Output divisualisasikan dalam bentuk pie chart dan stacked bar chart untuk mendukung analisis terhadap kelengkapan data nasabah.

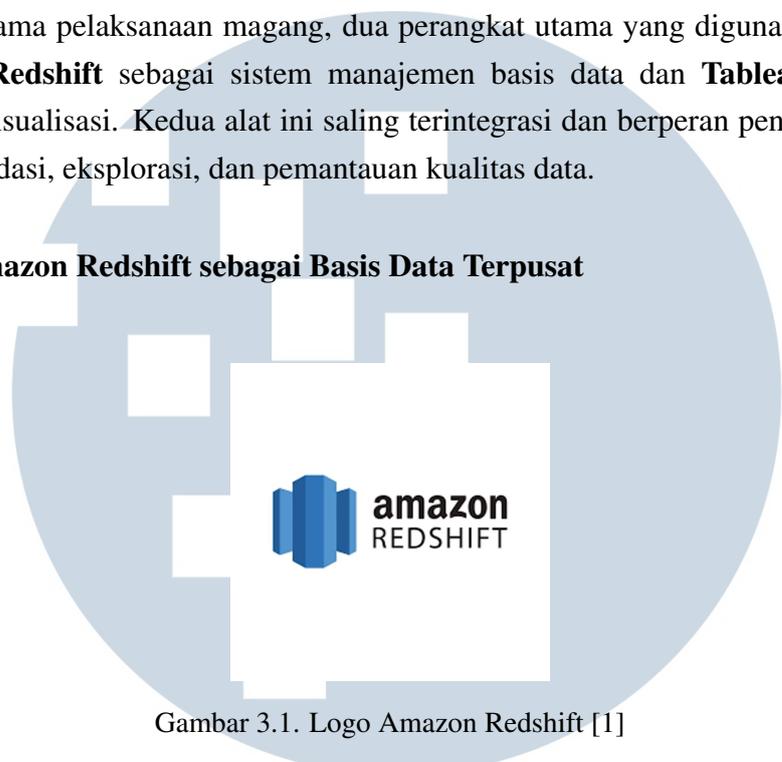
Tabel 3.2. Aktivitas dan capaian mingguan (Minggu 9–16)

Minggu	Aktivitas dan Capaian Mingguan
9	Persiapan pengembangan dashboard kedua yang difokuskan pada klasifikasi data bisnis. Kegiatan mencakup sesi diskusi bersama supervisor untuk menentukan field yang digunakan, hierarki filter, serta indikator visual yang diperlukan agar dashboard sesuai dengan kebutuhan operasional.
10	Penggabungan dua tabel data untuk pembuatan dashboard bisnis, yang mengakibatkan beban berat pada Tableau akibat ukuran data yang besar. Proses analisis dilakukan untuk mengidentifikasi bottleneck dan solusi alternatif untuk memecah atau mengurangi kompleksitas data.
11	Berdasarkan hasil evaluasi, diputuskan bahwa dashboard perlu diunggah ke server Tableau internal. Koordinasi dilakukan dengan tim IT untuk permintaan akses hosting, serta penyesuaian credential dan pengaturan koneksi jaringan untuk konektivitas Tableau–Redshift secara stabil.
12	Penggabungan tiga tabel menjadi satu struktur data yang lebih komprehensif. Dataset ini digunakan untuk dashboard ketiga yang menampilkan informasi kualitas data lintas sumber dengan kemampuan filter dan segmentasi multi-level.
13	Penambahan dimensi waktu ke dalam data (tahun, bulan, tanggal) guna memungkinkan visualisasi temporal yang mendalam. Visualisasi dirancang untuk memudahkan analisis tren dan perbandingan performa kualitas data antar periode.
14	Penerapan klasifikasi tambahan berupa kolom <code>priority_tier</code> dan <code>check_status</code> , yang digunakan untuk menandai data berisiko tinggi atau memerlukan verifikasi manual. Langkah ini membantu tim dalam menyusun prioritas tindak lanjut berdasarkan kondisi aktual data.
15	Penyempurnaan tampilan visualisasi dashboard mencakup pengaturan tata letak, penggunaan warna yang konsisten, dan penambahan parameter interaktif untuk meningkatkan user experience. Dashboard diuji untuk memastikan kompatibilitas dan performa baik di server maupun desktop.
16	Penyampaian hasil akhir dalam bentuk presentasi internal yang menjelaskan tujuan, proses, tantangan, dan outcome dari ketiga dashboard yang dikembangkan. Feedback dari tim digunakan untuk menyusun saran pengembangan lanjutan.

3.4 Peran Tools dalam Visualisasi dan Validasi Data

Selama pelaksanaan magang, dua perangkat utama yang digunakan adalah **Amazon Redshift** sebagai sistem manajemen basis data dan **Tableau** sebagai platform visualisasi. Kedua alat ini saling terintegrasi dan berperan penting dalam proses validasi, eksplorasi, dan pemantauan kualitas data.

3.4.1 Amazon Redshift sebagai Basis Data Terpusat



Gambar 3.1. Logo Amazon Redshift [1]

Amazon Redshift digunakan sebagai *cloud data warehouse* untuk menyimpan data hasil validasi serta data nasabah lainnya. Seluruh query SQL ditulis dan dijalankan terhadap tabel-tabel di Redshift, termasuk proses pembersihan data, klasifikasi, dan penggabungan antar sumber data. Redshift dipilih karena kemampuannya dalam menangani skala data besar dan dukungannya terhadap koneksi langsung dengan Tableau.

3.4.2 Tableau sebagai Alat Visualisasi dan Investigasi Data



Gambar 3.2. Logo Tableau [2]

Tableau dimanfaatkan untuk membangun dashboard interaktif yang menampilkan metrik kualitas data secara real-time. Fitur-fitur utama Tableau yang dimanfaatkan selama magang antara lain:

- Koneksi langsung ke Amazon Redshift menggunakan *live connection*,
- Pembuatan grafik pie chart, bar chart, dan summary sheet,
- Penggunaan parameter dinamis dan filter interaktif untuk eksplorasi data,
- Penggunaan *drill-down hierarchy* dan navigasi antar halaman dashboard.

3.4.3 Fitur *View Full Data* pada Tableau

Salah satu fitur penting yang sangat membantu proses investigasi data adalah “**View Full Data**” yang disediakan oleh Tableau. Fitur ini memungkinkan pengguna untuk:

- Melihat seluruh baris data yang telah terfilter berdasarkan field atau status validasi tertentu,
- Mengekspor data hasil filter ke format CSV atau Excel untuk keperluan koreksi manual atau audit,
- Menghindari kebutuhan akses langsung ke basis data, sehingga memudahkan pengguna non-teknis dalam menelusuri anomali data.

Penggunaan fitur ini sangat relevan dalam konteks dashboard kualitas data dan gray area. Setelah pengguna mengklik sebuah atribut atau status tertentu, filter akan diterapkan secara otomatis ke seluruh sheet, dan pengguna dapat melihat baris-baris data spesifik yang masuk dalam kondisi tersebut melalui fungsi *View Full Data*.

priority_tier	Month of Month Date	check_status	_Hoodie Commit Seqno	_Hoodie Commit Time	_Hoodie File Name
Priority 1	May 2025	Unchecked	20250520224739089_1_5919	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_14219	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_46798	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_36367	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_68935	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_43743	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_18301	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_12215	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_67949	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_55393	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_24232	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_23039	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_27993	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_65492	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_26014	20250520224739089	d3b6a435-b853-438d-a26f
Priority 1	May 2025	Unchecked	20250520224739089_1_8344	20250520224739089	d3b6a435-b853-438d-a26f

Gambar 3.3. Fungsi “View Full Data” pada Tableau yang memperlihatkan data terfilter secara utuh

3.5 Alur Umum Pengembangan Dashboard

Selama pelaksanaan kerja magang di Divisi Data Governance and Quality Assurance PT Bank BTPN Syariah, seluruh dashboard yang dikembangkan, baik dashboard kualitas data, gray area, maupun prioritas, dibangun dengan mengikuti alur proses yang terstruktur dari awal hingga akhir. Proses ini mencerminkan prinsip-prinsip dalam pengembangan sistem *Business Intelligence* (BI) modern yang menggabungkan validasi data, pengolahan di *data warehouse*, dan visualisasi interaktif.

Tahapan alur pengembangan dashboard secara umum terdiri dari:

1. **Pengumpulan dan Penyaringan Data:** Data nasabah dan data operasional lainnya dikumpulkan dari berbagai sistem internal perusahaan. Pada tahap

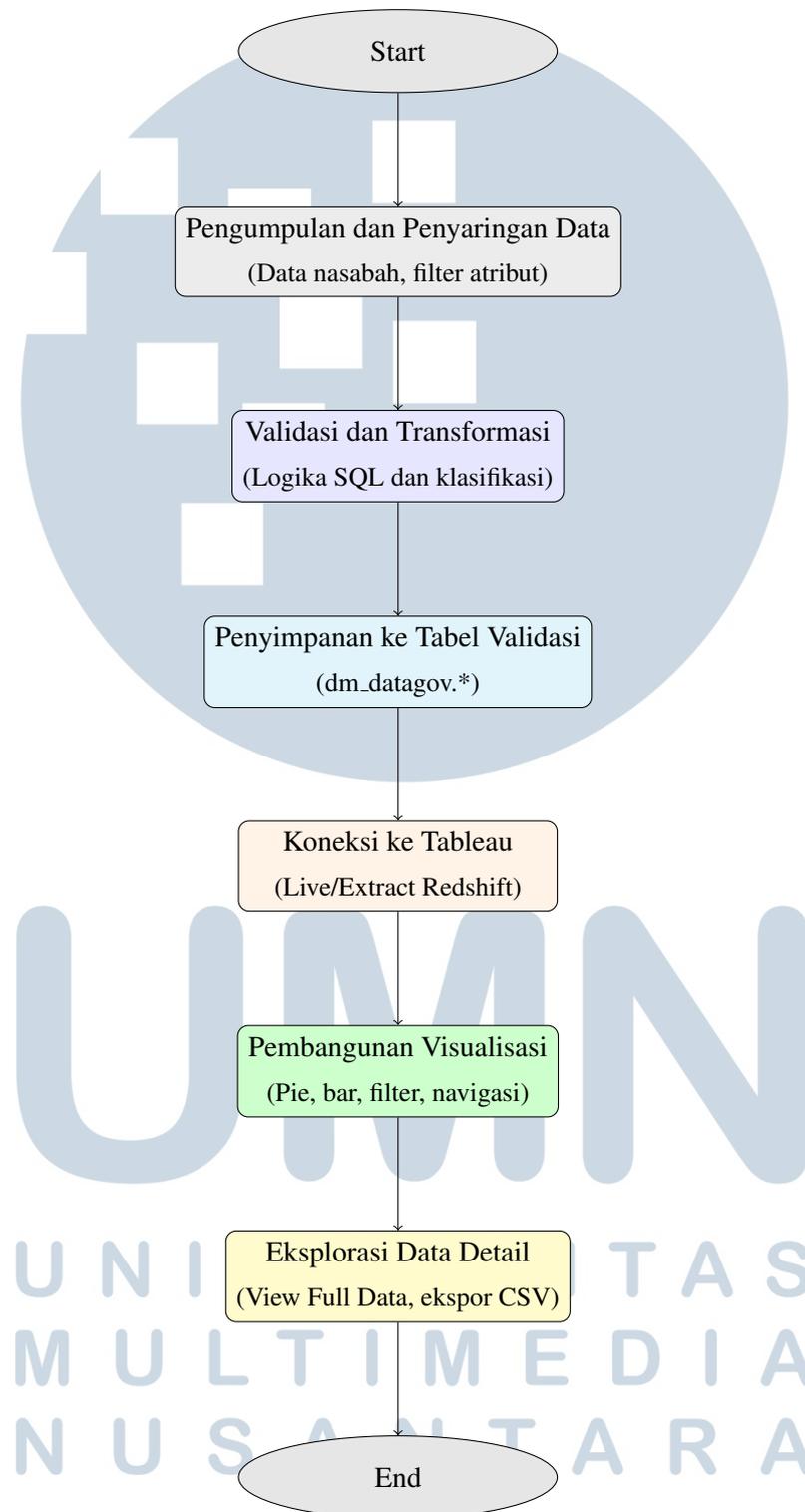
ini, dilakukan penyaringan awal berdasarkan atribut tertentu seperti jenis identitas, sumber sistem, dan cakupan waktu.

2. **Validasi dan Transformasi:** Data yang telah disaring kemudian divalidasi menggunakan logika SQL. Proses ini melibatkan pengecekan terhadap kelengkapan atribut, kesesuaian format, serta konsistensi antar data. Beberapa contoh logika yang digunakan antara lain pemeriksaan panjang karakter identitas, validitas kode gender, hingga ketepatan tanggal lahir. Hasil validasi kemudian diklasifikasikan menjadi berbagai status seperti Valid, Invalid, atau Null.
3. **Penyimpanan ke Tabel Validasi:** Output dari proses validasi disimpan ke dalam tabel khusus di schema `dm_datagov`. Beberapa tabel hasil akhir yang digunakan antara lain:
 - `dm_datagov.Validasi_Data_Detail`
 - `dm_datagov.mdm_test`
 - `dm_datagov.cif_test`
 - `dm_datagov.priority_test`

Tabel-tabel ini berfungsi sebagai dasar visualisasi dalam dashboard Tableau.

4. **Koneksi ke Tableau:** Tableau dikonfigurasi untuk membaca data dari Amazon Redshift menggunakan koneksi langsung (*Live*) maupun mode *Extract*. Koneksi ini memungkinkan Tableau menampilkan data terkini secara otomatis, sekaligus menjaga performa saat digunakan di perangkat pengguna.
5. **Pembangunan Visualisasi Interaktif:** Dashboard dibuat dengan memanfaatkan fitur visualisasi Tableau seperti pie chart, bar chart, dan parameter filter. Fitur drill-down, navigation sheet, dan filter global juga digunakan agar pengguna dapat menelusuri data secara lebih spesifik sesuai kebutuhan.
6. **Eksplorasi Data Detail:** Tableau menyediakan fitur "*View Full Data*" yang memungkinkan pengguna untuk melihat seluruh baris data hasil filter secara rinci. Fitur ini sangat bermanfaat bagi pengguna non-teknis karena memungkinkan mereka mengeksplor data yang telah difilter untuk proses investigasi atau koreksi lanjutan tanpa perlu mengakses database secara langsung.

Alur tersebut dirangkum dalam Gambar 3.4 berikut:



Gambar 3.4. Alur Pembuatan Dashboard

3.6 Implementasi Validasi dan Visualisasi Data untuk Dashboard Data Quality

Bagian ini menjelaskan proses pengolahan dan validasi data yang digunakan untuk membangun dashboard pemantauan kualitas data nasabah. Proses ini dilakukan melalui kombinasi beberapa *query*, salah satunya adalah *validasi_flags*, yang berfungsi memberikan *flag* terhadap nilai-nilai data yang tidak sesuai dengan pedoman dari Otoritas Jasa Keuangan (OJK). Output dari *query* ini digunakan sebagai dasar untuk membangun visualisasi yang menggambarkan tingkat *validitas* dan *kelengkapan* atribut data utama seperti nama, NIK, tanggal lahir, nomor telepon, dan kode pos.

3.6.1 Tujuan dan Konteks Dashboard

Data Quality Dashboard dikembangkan untuk membantu tim Data Governance dalam memantau kualitas atribut data nasabah, terutama dari sisi validitas dan kelengkapan. Dashboard ini menyajikan visualisasi metrik *validity* dan *completeness* yang berasal dari hasil validasi terhadap atribut-atribut penting, seperti *nama_lengkap*, *no_identitas*, *tanggal_lahir*, *telepon*, dan lainnya.

Tujuan utama dari dashboard ini adalah:

- Menyediakan pemantauan real-time terhadap kualitas data nasabah.
- Mengidentifikasi jenis kesalahan umum seperti penggunaan singkatan, jumlah digit yang salah, atau nilai kosong.
- Menyajikan hasil validasi dalam bentuk visual yang mudah dianalisis oleh stakeholder non-teknis.

Dashboard ini bersumber dari *query validasi_flags* dan *custom_query_data_quality*, yang kemudian divisualisasikan dalam bentuk grafik pie chart, bar chart, serta summary sheet yang terstruktur.

3.6.2 Validasi Atribut Data Nasabah

Langkah pertama dalam pengembangan dashboard *data quality* adalah melakukan proses validasi terhadap atribut-atribut wajib dari data nasabah berdasarkan format dan ketentuan yang ditetapkan oleh Otoritas Jasa Keuangan (OJK). Validasi ini dilakukan melalui *query Validasi_Flags*, yang menghasilkan

kolom tambahan seperti nama_lengkap_validity, no_identitas_validity, birthdate_validity, dan lainnya.

Query ini dijalankan terhadap dataset utama berisi data identitas nasabah, lalu hasilnya disimpan dalam tabel baru bernama DM_datagov.ValidasiData_Detail. Setiap nilai dari atribut akan diberi flag sebagai *Valid*, *Invalid*, atau *Null*, tergantung pada kondisi dan format data.

Tabel 3.3. Logika Validasi Atribut dan Output

Atribut	Kondisi Validasi	Output
nama_lengkap	Mengandung singkatan satu huruf atau simbol titik/koma	Invalid Singkatan
nama_lengkap	Terisi tanpa kesalahan format	Valid
nama_lengkap	Tidak terisi (null)	Null
nama_sesuai_identitas	Mengandung singkatan atau tanda baca	Invalid Singkatan
nama_sesuai_identitas	Terisi dengan benar	Valid
nama_sesuai_identitas	Tidak terisi	Null
jenis_kelamin_kode	Nilai bukan 1 atau 2 / panjang tidak sesuai	Invalid Kode
jenis_kelamin_kode	Nilai adalah 1 atau 2	Valid
jenis_kelamin_kode	Tidak terisi	Null
no_identitas	Panjang bukan 16 digit	Invalid Digit
no_identitas	Mengandung huruf atau simbol	Invalid Huruf
no_identitas	Format numerik 16 digit	Valid
no_identitas	Tidak terisi	Null
gender_validity	Kode sesuai NIK dan gender (1: male, 2: female)	Valid Male / Valid Female
gender_validity	Tidak sesuai antara kode dan NIK	Gender Mismatch
gender_validity	Format salah / tidak cocok	Invalid Gender
birthdate_validity	Format cocok dengan data di NIK	Valid
birthdate_validity	Tidak cocok (angka/tanggal berbeda)	Birthdate Mismatch
birthdate_validity	Format atau digit salah	Invalid Digit / Null
tanggal_lahir	Format dapat dikonversi ke tanggal	Valid
tanggal_lahir	Format tidak terbaca	Invalid Tanggal

Lanjut ke halaman berikutnya

Tabel 3.3 – lanjutan dari halaman sebelumnya

Atribut	Kondisi Validasi	Output
tanggal_lahir	Kosong / null	Null
telepon	Kurang dari 5 digit / salah awalan	Invalid Digit / Invalid Awalan
telepon	Mengandung huruf / simbol	Invalid Huruf
telepon	Awalan ganda nol	Invalid 0
telepon	Format benar	Valid
telepon	Tidak terisi	Null
mobile_phone	Kurang dari 10 digit atau lebih dari 13	Invalid Digit
mobile_phone	Mengandung karakter non-numerik	Invalid Huruf
mobile_phone	Awalan bukan 0 atau 00	Invalid Awalan / Invalid 0
mobile_phone	Format benar	Valid
mobile_phone	Kosong	Null
kode_pos	Bukan 5 digit angka	Invalid Digit
kode_pos	Mengandung huruf / simbol	Invalid Huruf
kode_pos	Format benar (5 digit angka)	Valid
kode_pos	Kosong / null	Null

Potongan query Validasi_Flags ditampilkan pada Kode 3.1:

```

1 WITH Validations AS (
2     SELECT
3         nama_lengkap,
4         nama_sesuai_identitas,
5         no_identitas,
6         jenis_kelamin_kode,
7         tanggal_lahir,
8         telepon,
9         mobile_phone,
10        kode_pos
11    FROM ORIG
12    WHERE jenis_identitas = 'KTP'
13 )
14 SELECT

```

```

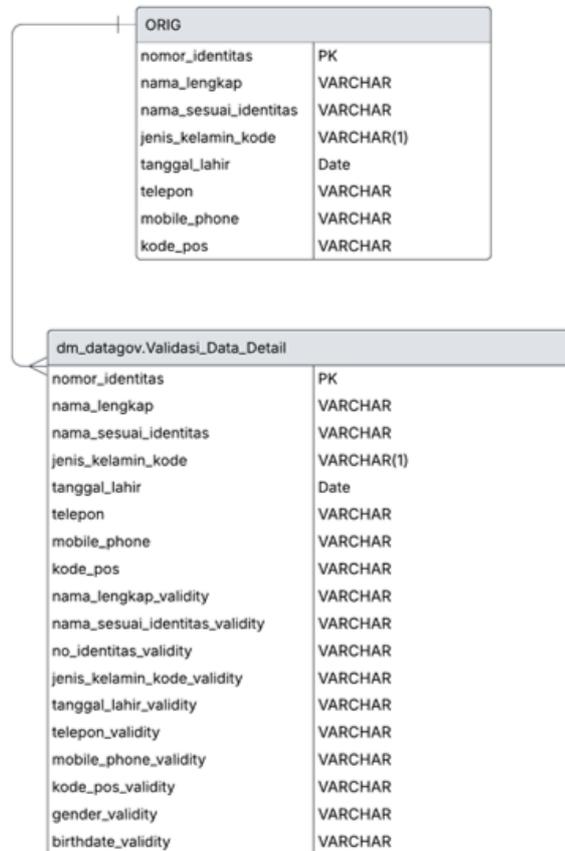
15     nama_lengkap, nama_sesuai_identitas, no_identitas,
16     jenis_kelamin_kode, tanggal_lahir,
17     telepon, mobile_phone, kode_pos,
18
19     CASE
20     WHEN nama_lengkap IS NULL THEN 'NULL'
21     WHEN nama_lengkap ~ '[[<:]] [A-Za-z] [[>:]]' OR
22     nama_lengkap ~ '[.,]' THEN 'Invalid Singkatan'
23     ELSE 'Valid'
24
25     END AS nama_lengkap_validity,
26
27     -- Potongan selanjutnya disederhanakan untuk kejelasan
28     ...
29
30 INTO DM_datagov.Validasi_Data_Detail
31 FROM Validations

```

Kode 3.1: Query Validasi Flags

Output dari query tersebut menghasilkan tabel yang memuat status validitas untuk setiap atribut utama seperti nama lengkap, jenis kelamin, dan tanggal lahir. Contoh hasil dari output tersebut ditampilkan pada Gambar





Gambar 3.6. Entity Relationship Diagram (ERD) Validasi Data Quality

Setiap entri pada tabel `Validasi_Data_Detail` menyimpan nilai asli serta status validitas dari atribut-atribut utama, seperti `nama_lengkap`, `tanggal_lahir`, dan `no_identitas`. Relasi utama antara kedua tabel mengacu pada kolom `nomor_identitas`, yang berperan sebagai *primary key* di tabel `ORIG` dan sebagai *foreign key* di tabel `Validasi_Data_Detail`.

Struktur ini memungkinkan pemisahan antara data asli dan hasil validasinya secara terorganisir, sekaligus memudahkan proses pelacakan dan pelaporan kualitas data pada dashboard.

3.6.4 Transformasi Data untuk Visualisasi

Setelah proses validasi selesai, hasil dari `Validasi_Flags` ditransformasikan untuk keperluan visualisasi dengan menggunakan query tambahan, yaitu `Custom Query for Data Quality Dashboard`. Tujuan dari query ini adalah mengubah struktur data menjadi bentuk `long-table`, di mana setiap

baris mewakili atribut tertentu dari satu nasabah dengan status `validity` dan `completeness`.

```
1 SELECT
2   no_identitas,
3   nama_lengkap,
4   nama_sesuai_identitas,
5   jenis_kelamin_kode,
6   tanggal_lahir,
7   telepon,
8   mobile_phone,
9   kode_pos,
10  'nama_lengkap' AS field_name,
11  CASE
12    WHEN nama_lengkap_validity = 'Valid' THEN 'Valid'
13    WHEN nama_lengkap_validity = 'Invalid Singkatan' THEN '
14     Mengandung Singkatan'
15    ELSE nama_lengkap_validity
16  END AS validity,
17  CASE WHEN nama_lengkap IS NULL THEN 'Null' ELSE 'Complete' END
18    AS completeness
19 FROM DM_datagov.Validasi_Data_Detail
20
21 UNION ALL
22
23 -- (continue with the rest of your UNION ALL blocks as needed)
```

Kode 3.2: Custom SQL Query untuk Validasi Data Quality

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

field_name	validity	completeness
nama_lengkap	Valid	Complete

Gambar 3.7. Contoh hasil output dari query validasi data quality

Gambar 3.7 menunjukkan hasil dari eksekusi *custom query* yang digunakan dalam pembangunan Dashboard Data Quality. Setiap baris pada tabel mewakili validasi terhadap satu atribut dari seorang nasabah, dengan kolom *field_name* yang menunjukkan nama atribut (seperti *nama_lengkap*, *no_identitas*, dan lain-lain), kolom *validity* yang menunjukkan status keabsahan nilai atribut tersebut (misalnya Valid, Mengandung Singkatan, Format Tanggal Salah, dll), dan kolom *completeness* yang menunjukkan apakah nilai atribut tersebut kosong (Null) atau terisi (Complete). Struktur tabel ini digunakan agar dapat divisualisasikan secara fleksibel di Tableau menggunakan filter dinamis dan agregasi berdasarkan atribut.

Struktur ini mempermudah pembuatan visualisasi seperti:

- **Pie Chart** untuk melihat proporsi data valid dan tidak valid pada masing-masing field dan juga untuk perbandingan data lengkap dan null.

- **Bar Chart Horizontal** untuk klasifikasi error seperti “Jumlah Digit Salah” atau “Mengandung Singkatan”.

3.6.5 Visualisasi Dashboard Kualitas Data

Dashboard yang dikembangkan dalam proyek ini terdiri dari dua halaman utama (*pages*) yang saling terhubung melalui fitur interaktif. Tujuan utama dashboard ini adalah untuk memantau dan menganalisis kualitas data nasabah berdasarkan hasil validasi dan kelengkapan atribut yang telah diproses melalui query *custom SQL*.

Field Name	
Telepon	0,064%
HP	41,672%
Kode Pos	82,843%
Tanggal Lahir dibandingkan KTP	90,855%
Jenis Kelamin dibandingkan KTP	97,398%
Nomor Identitas	98,158%
Nama Sesuai Identitas	99,263%
Nama Lengkap	99,665%
Kode Jenis Kelamin	100,000%

Gambar 3.8. Tampilan utama Dashboard Kualitas Data (Page 1)

- **Halaman 1 – Threshold Sheet:** Halaman ini menampilkan satu *sheet* utama berupa tabel metrik validitas dalam bentuk persentase untuk setiap atribut. Tabel ini membantu pengguna dalam mengidentifikasi atribut mana

yang memiliki tingkat validasi terendah atau tertinggi. Setiap baris pada tabel ini bersifat interaktif — pengguna dapat mengklik nama atribut untuk mengaktifkan filter dinamis yang akan mengarahkan ke halaman kedua.



Gambar 3.9. Tampilan lanjutan Dashboard Kualitas Data (Page 2)

- **Halaman 2 – Visualisasi Detail:** Setelah pengguna memilih atribut dari halaman pertama, halaman ini menampilkan tiga *sheet* visualisasi interaktif berdasarkan field yang difilter:
 1. **Pie Chart – Valid vs Invalid:** Menampilkan proporsi antara data valid dan data yang tidak valid untuk atribut yang dipilih.
 2. **Pie Chart – Complete vs Null:** Menampilkan perbandingan antara data yang terisi (*complete*) dan data kosong (*null*).
 3. **Bar Chart – Jenis Validitas:** Menampilkan distribusi jumlah data berdasarkan jenis validitas yang ada.

Seluruh visualisasi tersebut dirancang dengan prinsip *interactive filtering*, di mana satu klik pada atribut di halaman pertama akan secara otomatis menyaring seluruh visualisasi di halaman kedua berdasarkan atribut tersebut. Pendekatan ini sangat membantu dalam melakukan analisis mendalam terhadap atribut yang bermasalah dalam skema validasi data.

3.7 Pembuatan dan Pengolahan Data untuk Dashboard Gray Area

Bagian ini menjelaskan proses pengolahan data yang digunakan untuk membangun dashboard klasifikasi *gray area* berdasarkan identitas dan jumlah CIF (*Customer Information File*) yang dimiliki oleh nasabah. Proses ini dilakukan dengan menggunakan kombinasi dari beberapa *query* seperti *mdm_test*, *cif_test*, dan *check_status*. Dashboard ini bertujuan untuk mengidentifikasi potensi duplikasi data identitas, serta menilai status verifikasi data berdasarkan proses *maker-checker* yang dilakukan oleh tim verifikator internal.

Tujuan dan Konteks Dashboard

Dashboard ini dirancang untuk mendeteksi kejanggalan dalam jumlah CIF yang terhubung ke satu identitas nasabah. Dengan memantau metrik seperti *cif_count*, *cif_tier*, dan *check_status*, pengguna dashboard dapat menilai apakah data sudah diverifikasi secara menyeluruh, serta mengantisipasi terjadinya redundansi atau kesalahan dalam proses pengelolaan data nasabah.

3.7.1 Query *mdm_test*: Label Prioritas dan Tanggal Open CIF

Langkah pertama dalam pembangunan dashboard Gray Area adalah menyaring data dari dua sistem sumber utama, yaitu *PROSPERA* dan *T24*. Data yang diambil diproses untuk menambahkan label prioritas berdasarkan keberadaan data dalam tabel *BIAYA*. Hasil akhir dari proses ini dimasukkan ke dalam tabel *dm_datagov.mdm_test*.

```
1 -- Query: Custom Query for Data Gray Area
2
3 DROP TABLE IF EXISTS dm_datagov.mdm_test;
4
5 WITH stg_filtered AS (
6     SELECT no_cif
7     FROM Tabel Pembiayaan
```

```

8     WHERE source IN ('prospera', 't24')
9 ),
10 orig_combined AS (
11     SELECT
12         orig.cif_orig,
13         orig.nama_lengkap,
14         orig.nama_sesuai_identitas,
15         orig.jenis_kelamin,
16         orig.tempat_lahir,
17         orig.tanggal_lahir,
18         orig.status_perkawinan_debitur,
19         orig.nama_pasangan,
20         orig.nama_gadis_ibu_kandung,
21         cr.*,
22         (TIMESTAMP '1970-01-01 00:00:00' + INTERVAL '1 second' * (
23             orig.open_date / 1000000)) AS converted_open_date
24     FROM MDM ORIG
25     JOIN MDM ORIG CR
26         ON orig.match_group = cr.match_group AND orig.crn = cr.crn
27     WHERE ORIG IN ('PROSPERA', 'T24')
28 ),
29 final_labeled AS (
30     SELECT
31         oc.*,
32         CASE
33             WHEN sf.no_cif IS NOT NULL THEN 'Priority 1'
34             ELSE 'Priority 2'
35         END AS priority_tier
36     FROM orig_combined oc
37     LEFT JOIN stg_filtered sf
38         ON TRIM(oc.cif_orig) = sf.no_cif
39 )
40 SELECT *
41 INTO dm_datagov.mdm_test
42 FROM final_labeled;

```

Kode 3.3: Query untuk Membuat Tabel `mdm_test`

3.7.2 Query `cif_test`: Penandaan `ID_Flag` untuk Duplikasi Identitas

Tahap kedua adalah membuat query `cif_test`, yang bertujuan untuk mengidentifikasi apakah satu `no_identitas` digunakan oleh lebih dari satu CIF.

Hal ini sangat penting untuk mendeteksi potensi duplikasi atau ketidaksesuaian data identitas nasabah.

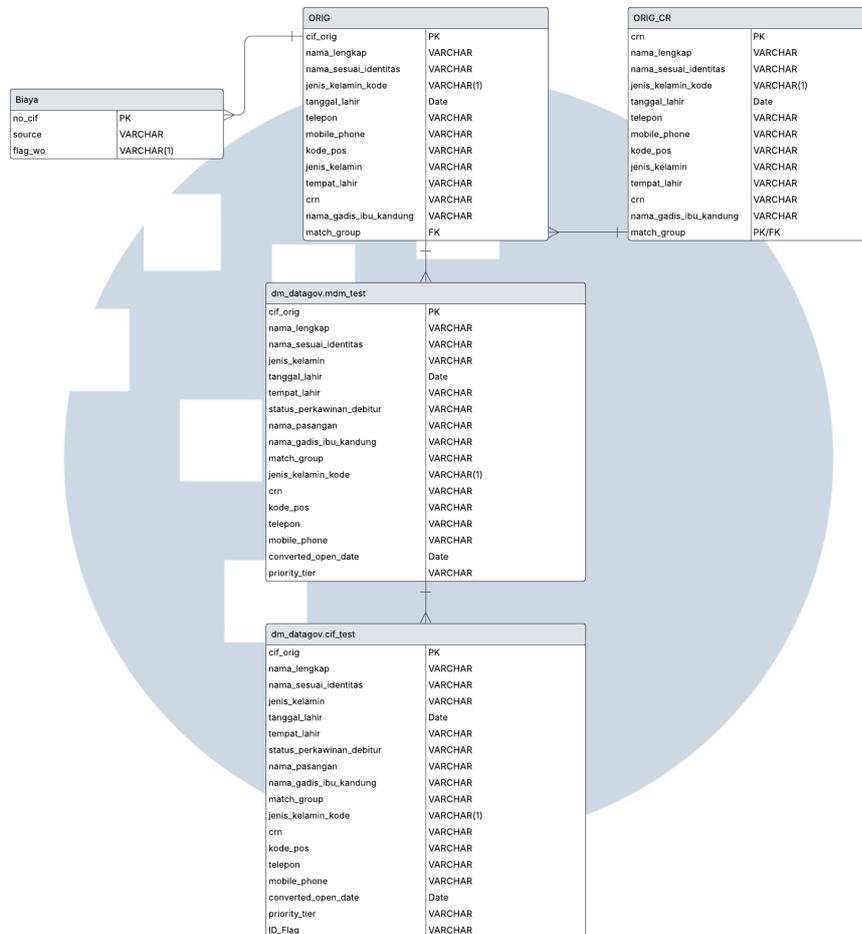
```
1 DROP TABLE dm_datagov.cif_test;
2
3 SELECT
4     *,
5     CASE
6         WHEN no_identitas IN (
7             SELECT no_identitas
8             FROM dm_datagov.mdm_test
9             GROUP BY no_identitas
10            HAVING COUNT(DISTINCT cif_orig) > 1
11        )
12        THEN 'Multiple ID'
13        ELSE 'Unique ID'
14    END AS ID_Flag
15 INTO dm_datagov.cif_test
16 FROM dm_datagov.mdm_test;
```

Kode 3.4: Query untuk Menambahkan Kolom ID_Flag

3.7.3 ERD Dashboard Gray Area

Gambar 3.10 menunjukkan hubungan antar entitas yang digunakan dalam pembangunan dashboard Gray Area. Diagram ini mencakup dua tabel utama, yaitu dm_datagov.mdm_test dan dm_datagov.cif_test, yang membentuk basis data monitoring prioritas nasabah dan deteksi potensi duplikasi identitas.

U M M N
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.10. Entity Relationship Diagram (ERD) Dashboard Gray Area

Tabel `dm_datagov.mdm_test` merupakan hasil transformasi dari beberapa sumber data internal, termasuk data nasabah dari sistem *orig* dan informasi tambahan dari *staging area*. Proses ini mencakup penggabungan atribut nasabah, penambahan tanggal `open_date` dalam format terkonversi, serta pemberian label prioritas berdasarkan keberadaan data dalam sistem *Biaya*. Setiap baris pada tabel ini mewakili satu entitas CIF, dengan `cif_orig` sebagai atribut utama.

Selanjutnya, tabel `dm_datagov.cif_test` dibuat dari `mdm_test` dengan menambahkan kolom `ID_Flag`, yang menunjukkan apakah satu nomor identitas digunakan oleh lebih dari satu CIF. Jika ditemukan satu `no_identitas` digunakan oleh lebih dari satu `cif_orig`, maka entri tersebut diberi label *Multiple ID*, sebaliknya diberi label *Unique ID*. Proses ini penting untuk mengidentifikasi potensi duplikasi identitas dan mendukung validasi data yang lebih mendalam.

Relasi utama antara kedua tabel adalah pada kolom `no_identitas`, yang

berfungsi sebagai referensi untuk mengevaluasi keberulangan identitas dalam sistem. Struktur ini memungkinkan pemantauan granular terhadap status identitas dan prioritas akun nasabah dalam dashboard interaktif.

3.7.4 Query Tambahan: Penambahan Kolom `cif_tier` dan `check_status`

Query ketiga merupakan bagian dari **Custom Query for Data Gray Area**, yang menambahkan dua kolom analitik tambahan ke dalam dataset akhir, yaitu:

- `cif_tier` – Mengelompokkan nasabah berdasarkan jumlah CIF unik yang terkait dengan satu nomor identitas.
- `check_status` – Menandai status pengecekan manual berdasarkan nilai waktu pada kolom `maker_time`, `checker_time1`, dan `checker_time2`.

```
1 SELECT
2     t.*,
3     c.cif_count,
4     CASE
5         WHEN c.cif_count = 1 THEN '1 CIF'
6         WHEN c.cif_count = 2 THEN '2 CIFs'
7         WHEN c.cif_count = 3 THEN '3 CIFs'
8         WHEN c.cif_count = 4 THEN '4 CIFs'
9         WHEN c.cif_count = 5 THEN '5 CIFs'
10        ELSE '6+ CIFs'
11    END AS cif_tier,
12    CASE
13        WHEN t.maker_time IS NULL THEN 'Unchecked'
14        WHEN t.checker_time1 IS NULL THEN 'Maker'
15        WHEN t.checker_time2 IS NULL THEN 'Check 1'
16        ELSE 'Check 2'
17    END AS check_status
18 FROM dm_datagov.cif_test t
19 LEFT JOIN (
20     SELECT
21         no_identitas,
22         COUNT(DISTINCT cif_orig) AS cif_count
23     FROM dm_datagov.cif_test
24     GROUP BY no_identitas
25 ) c ON t.no_identitas = c.no_identitas;
```

Kode 3.5: Query untuk Menambahkan Kolom `cif_tier` dan `check_status`

Ketiga query di atas, yaitu `mdm_test`, `cif_test`, dan *Custom Query for Data Gray Area*, secara bersama-sama menghasilkan dataset akhir yang digunakan sebagai sumber data untuk pembangunan Dashboard Gray Area. Dataset ini memberikan informasi yang holistik mengenai klasifikasi prioritas nasabah, validitas identitas, dan status pengecekan data untuk mendukung proses pengawasan dan investigasi internal.

3.7.5 Proses Visualisasi dan Navigasi Dashboard Gray Area

Dashboard Gray Area dirancang sebagai alat pemantauan yang interaktif untuk menelusuri status validasi data berdasarkan waktu dan tingkat pengecekan. Dashboard ini terdiri dari **5 halaman (pages)** yang saling terhubung melalui *action filters* dan *navigational buttons*, memungkinkan pengguna untuk melakukan analisis bertingkat dari agregasi tahunan hingga daftar nasabah dengan prioritas tertentu.

- **Halaman 1 – Summary Status Sheet:** Halaman pertama menampilkan total agregat jumlah kasus berdasarkan status validasi manual, yaitu Unchecked, Maker, dan Check 1. Terdapat sebuah tombol interaktif berwarna hijau bertuliskan `Go to Monitoring` yang berfungsi sebagai pintasan untuk berpindah ke halaman monitoring detail berdasarkan filter global.



Check Status

Check 1	10.000
Maker	20.000
Unchecked	30.000

[Go to Monitoring](#)

Gambar 3.11. Tampilan Halaman 1 – Summary Status Sheet

- **Halaman 2 – Ringkasan Tahunan dan Breakdown:** Pada halaman ini terdapat dua *sheet* utama:

1. **Total Tahunan:** Menampilkan total jumlah kasus (semua status) per tahun.
2. **Bar Chart Breakdown:** Memisahkan kasus Unchecked, Maker, dan Check 1 dalam bar chart berdasarkan tahun.

Pengguna dapat mengklik salah satu tahun untuk mengaktifkan filter dan berpindah ke halaman ketiga.

Return to Check Status

Total		Unchecked	Maker	Check 1
2001	60.000	20.000	20.000	20.000
2008	60.000	20.000	20.000	20.000
2009	60.000	20.000	20.000	20.000
2010	60.000	20.000	20.000	20.000
2011	60.000	20.000	20.000	20.000
2012	60.000	20.000	20.000	20.000
2013	60.000	20.000	20.000	20.000
2014	60.000	20.000	20.000	20.000
2015	60.000	20.000	20.000	20.000
2016	60.000	20.000	20.000	20.000
2017	60.000	20.000	20.000	20.000
2018	60.000	20.000	20.000	20.000
2019	60.000	20.000	20.000	20.000
2020	60.000	20.000	20.000	20.000
2021	60.000	20.000	20.000	20.000
2022	60.000	20.000	20.000	20.000
2023	60.000	20.000	20.000	20.000
2024	60.000	20.000	20.000	20.000
2025	60.000	20.000	20.000	20.000

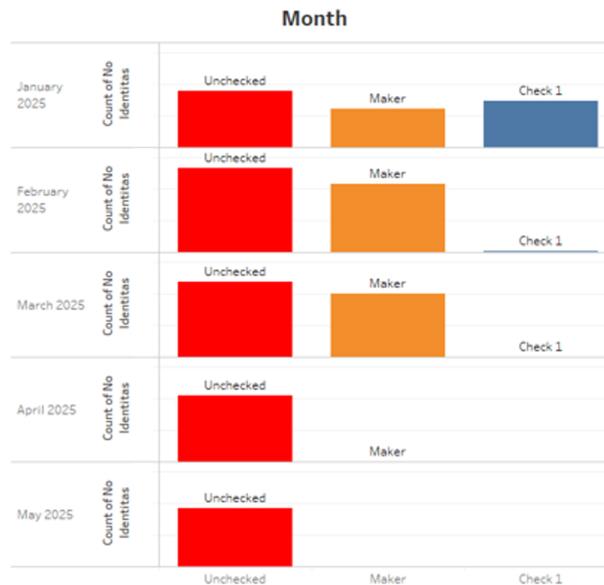
Gambar 3.12. Tampilan Halaman 2 – Ringkasan Tahunan dan Breakdown

- **Halaman 3 – Monitoring Bulanan:** Halaman ini memvisualisasikan data yang telah difilter berdasarkan tahun sebelumnya. Ditampilkan dalam bentuk:
 1. **Bar Chart Status Bulanan:** Jumlah kasus per bulan yang dibedakan berdasarkan Unchecked, Maker, dan Check 1.
 2. **Total Kasus Bulanan:** Ringkasan agregat jumlah kasus per bulan.

Interaksi pada bar tertentu atau nama bulan akan mengarahkan ke halaman berikutnya.

Return to Years

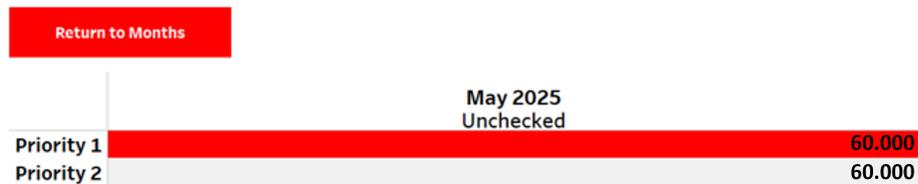
Total	
January 2025	60.000
February 2025	60.000
March 2025	60.000
April 2025	60.000
May 2025	60.000



Gambar 3.13. Tampilan Halaman 3 – Monitoring Bulanan

- **Halaman 4 – Prioritas Berdasarkan Filter:** Halaman ini menampilkan distribusi jumlah CIF yang berada pada kategori Priority 1 dan Priority 2. Filter aktif dapat berupa kombinasi dari tahun, bulan, dan status pengecekan. Halaman ini mempermudah analisis tingkat prioritas berdasarkan waktu dan status pengecekan data.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.14. Tampilan Halaman 4 – Distribusi Prioritas CIF

3.8 Pengelolaan dan Klasifikasi Data Prioritas untuk Dashboard Data Anomali

Bagian ini menjelaskan proses pengolahan data yang digunakan untuk membangun *dashboard* deteksi anomali berbasis data prioritas nasabah. Proses ini dilakukan dengan menggunakan query `priority_test`, yang bertujuan untuk mengidentifikasi akun-akun nasabah dengan kondisi khusus seperti write-off (WO), serta mengelompokkan mereka ke dalam tingkat prioritas berdasarkan status dan atribut tertentu.

3.8.1 Tujuan dan Konteks Dashboard

Dashboard ini membantu tim data governance dan operasional dalam memantau nasabah dengan jumlah akun tinggi dan status *flag WO*, serta

memberikan visualisasi kategori `priority_tier` sebagai indikator utama. Data yang ditampilkan mencerminkan kondisi aktual yang dapat berdampak terhadap kualitas pelaporan atau analisis risiko.

3.8.2 Logika dan Tahapan Query `priority_test`

Query ini dikemas dalam sebuah prosedur dan melalui beberapa tahap penting:

1. **Filter Data Staging:** Mengambil data dari tabel BIAYA dengan sumber prospera dan t24. Ini merupakan sumber informasi flag WO dan no.cif.
2. **Penandaan *Flag WO*:** Menggunakan agregasi untuk menentukan apakah suatu nasabah memiliki riwayat write-off, menghasilkan kolom `has_wo`.
3. **Klasifikasi Prioritas:** Setiap nasabah diklasifikasikan sebagai:
 - Priority 2 jika memiliki `flag_wo = 'Y'`
 - Priority 1 jika tidak memiliki WO namun ditemukan dalam data staging
 - Priority 3 jika tidak ditemukan dalam data staging sama sekali
4. **Join ke Tabel MDM:** Hasil klasifikasi dikombinasikan dengan data `mdm_ob_orig` untuk menampilkan atribut seperti nama, tanggal lahir, dan status perkawinan.
5. **Penentuan Status dan Jumlah Akun:** Ditambahkan kolom `account_status` (WO / Non-WO) serta `total_accounts_per_cif` untuk melihat berapa banyak akun per CIF.

3.8.3 Cuplikan Kode Query `priority_test`

```
1 CREATE OR REPLACE PROCEDURE dm_datagov.cmdm_untuk_rr ()
2 LANGUAGE plpgsql
3 AS $$
4 BEGIN
5 DROP TABLE IF EXISTS dm_datagov.priority_test;
6
7 WITH stg_filtered AS (
```

```

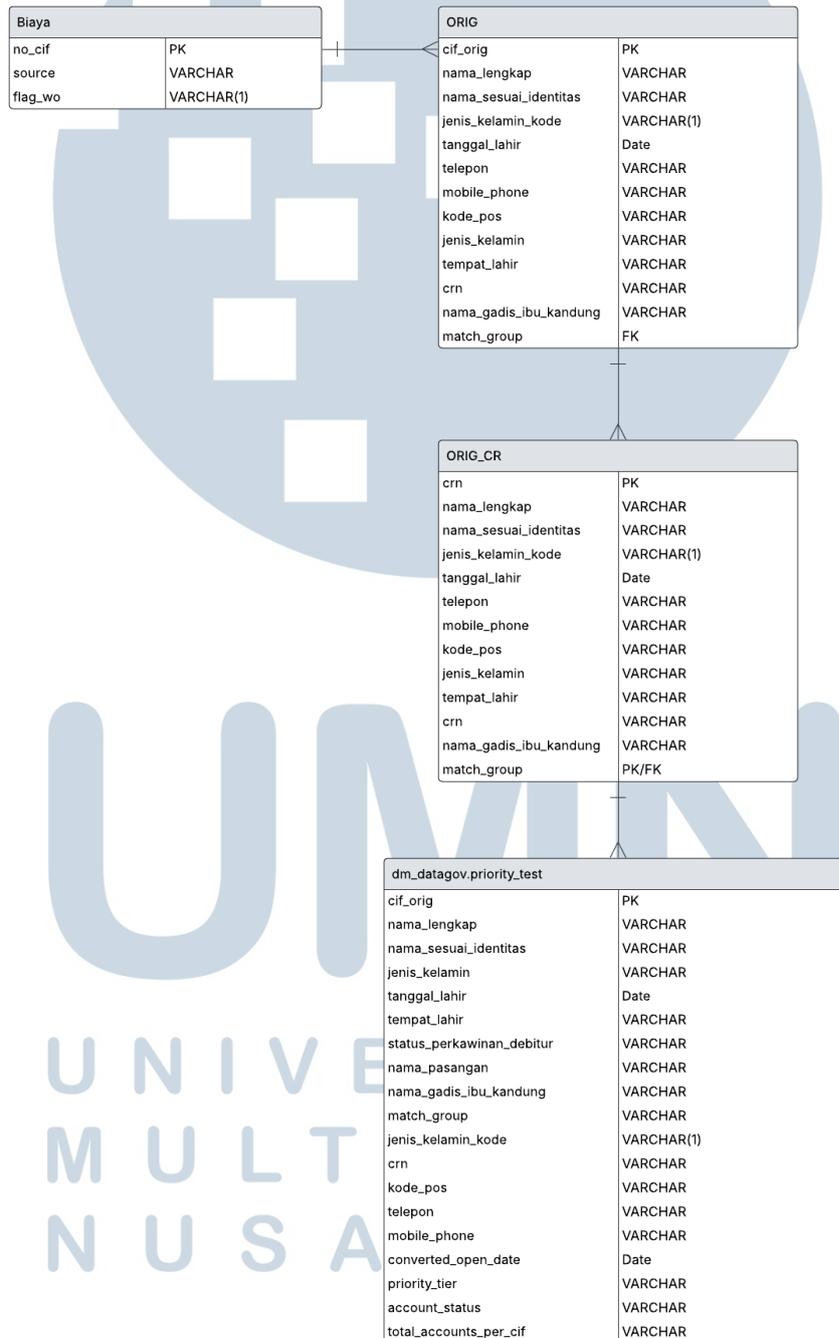
8      SELECT * FROM BIAYA
9      WHERE source IN ('prospera', 't24')
10 ),
11 stg_flags AS (
12     SELECT no_cif,
13            MAX(CASE WHEN flag_wo = 'Y' THEN 1 ELSE 0 END) AS
14     has_wo
15     FROM stg_filtered
16     GROUP BY no_cif
17 ),
18 priority_labels AS (
19     SELECT no_cif,
20            CASE
21              WHEN has_wo IS NULL THEN 'Priority 3'
22              WHEN has_wo = 1 THEN 'Priority 2'
23              ELSE 'Priority 1'
24            END AS priority_tier
25     FROM stg_flags
26 )
27 SELECT
28     orig.cif_orig,
29     orig.nama_lengkap,
30     ...
31     COALESCE(pl.priority_tier, 'Priority 3') AS priority_tier,
32     CASE
33       WHEN p.flag_wo = 'Y' THEN 'WO'
34       WHEN p.no_cif IS NOT NULL THEN 'Non-WO'
35       ELSE NULL
36     END AS account_status,
37     COUNT(*) OVER (PARTITION BY orig.cif_orig) AS
38     total_accounts_per_cif
39 INTO dm_datagov.priority_test
40 FROM ORIG
41 LEFT JOIN stg_filtered p ON TRIM(orig.cif_orig) = TRIM(p.no_cif)
42 LEFT JOIN priority_labels pl ON TRIM(orig.cif_orig) = TRIM(pl.
43     no_cif);
44 END;
45 $$

```

Kode 3.6: Cuplikan query priority_test.sql

3.8.4 ERD Dashboard Data Anomali

Gambar 3.15 menunjukkan struktur entitas yang digunakan dalam proses pembuatan tabel `dm_datagov.priority_test`, yang menjadi basis utama untuk dashboard deteksi data anomali berdasarkan prioritas nasabah.



Gambar 3.15. Entity Relationship Diagram (ERD) Dashboard Data Anomali

Tabel `dm_datagov.priority_test` dibentuk dari penggabungan antara data nasabah dari tabel `mdm_ob_orig` dan data biaya dari tabel `Pros_stg_Biaya`. Tahapan awal melibatkan penyaringan sumber data berdasarkan `source` (`prospera` dan `t24`), yang kemudian dianalisis untuk mendeteksi status `write-off` (WO) melalui atribut `flag_wo`.

Selanjutnya, dilakukan proses pemberian label prioritas menggunakan logika kondisi sebagai berikut:

- **Priority 1:** Tidak memiliki flag WO dan tidak tercatat di `stg_filtered`.
- **Priority 2:** Memiliki flag WO.
- **Priority 3:** Tidak tercatat sama sekali di tabel biaya.

Atribut tambahan seperti `account_status` dan `total_accounts_per_cif` turut disertakan untuk mendukung visualisasi detail pada dashboard. Tabel akhir digunakan sebagai sumber utama dalam visualisasi Tableau untuk memantau akun nasabah berdasarkan tingkat prioritas dan status WO secara interaktif.

Relasi utama dalam ERD ini adalah antara entitas `mdm_ob_orig`, `Pros_stg_Biaya`, dan hasil agregasi `priority_labels`, yang semuanya menyatu dalam tabel akhir `dm_datagov.priority_test`.

3.8.5 Catatan Mengenai Contoh Hasil Query

Hingga saat laporan ini disusun, belum tersedia contoh hasil keluaran (*output*) dari query `priority_test` yang dapat ditampilkan. Hal ini disebabkan karena data sumber (*data source*) yang digunakan dalam query tersebut sedang dalam proses pemutakhiran dan migrasi oleh tim data engineering. Oleh karena itu, eksekusi penuh terhadap query belum dapat dilakukan secara aktual untuk menghasilkan tabel final.

Meski demikian, struktur logika query sudah sepenuhnya disiapkan dan disesuaikan dengan format data baru yang akan digunakan. Setelah data sumber tersebut tersedia dan terintegrasi, hasil dari query ini akan langsung mengisi tabel `dm_datagov.priority_test` secara otomatis melalui prosedur `cmdm_untuk_rr()`, dan kemudian ditarik ke dalam dashboard menggunakan koneksi Tableau–Redshift seperti dua dashboard lainnya.

3.9 Kendala dan Solusi yang Ditemukan

3.9.1 Kendala

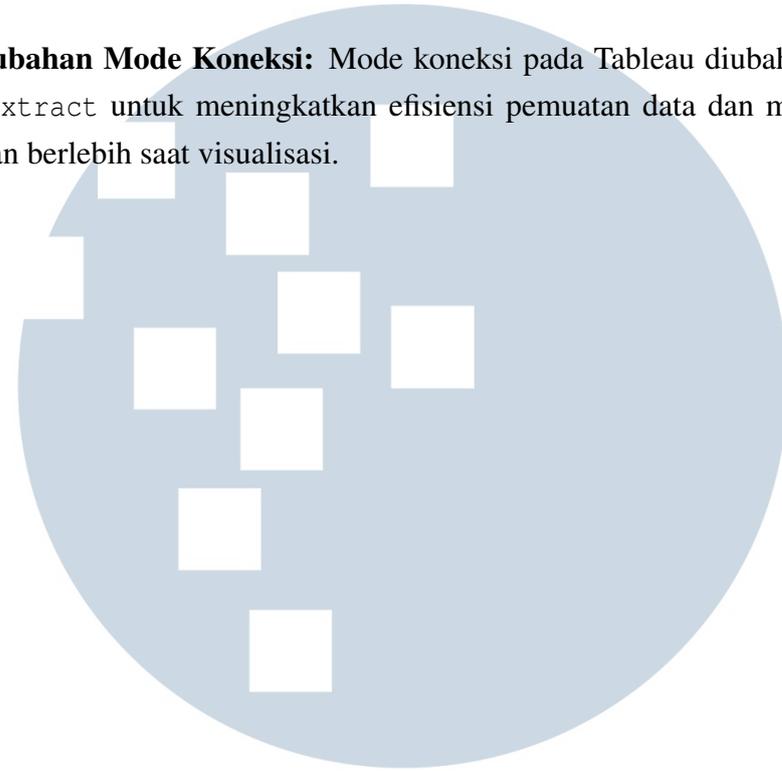
1. **Perangkat:** Perangkat kerja yang digunakan memiliki keterbatasan spesifikasi sehingga tidak optimal untuk menjalankan proses pemrosesan dan visualisasi data dalam skala besar.
2. **Integrasi Tableau:** Integrasi dengan Tableau tidak dapat dilakukan di awal karena terdapat keterlambatan dalam penyediaan lisensi Tableau yang dibutuhkan.
3. **Kompleksitas Validasi Data:** Proses penyusunan query SQL untuk validasi memerlukan kombinasi logika kondisi dari beberapa atribut, yang menyebabkan proses pengujian menjadi lebih panjang.
4. **Database:** Database yang digunakan untuk pembangunan dashboard `priority_test` memiliki ukuran dan kompleksitas yang tinggi, sehingga tidak dapat dijalankan secara lokal pada perangkat dengan memori terbatas.
5. **Keterbatasan Interaktivitas Tableau:** Beberapa fungsi interaktif Tableau tidak dapat dijalankan secara optimal ketika menggunakan koneksi langsung ke Redshift pada perangkat dengan spesifikasi standar.

3.9.2 Solusi

1. **Eksplorasi Mandiri Dokumentasi:** Selama proses penyesuaian perangkat, dilakukan studi mandiri terhadap dokumentasi resmi Redshift dan Tableau untuk memahami struktur dan konektivitas sistem.
2. **Koordinasi Teknis Internal:** Permasalahan lisensi diselesaikan melalui koordinasi dengan divisi terkait untuk memastikan lisensi Tableau dapat segera digunakan.
3. **Segmentasi Query SQL:** Struktur query dibagi menjadi beberapa blok berdasarkan atribut. Pendekatan ini memfasilitasi pengujian dan verifikasi logika per bagian.
4. **Rencana Pengunggahan ke Server Internal:** Dashboard dengan dataset besar direncanakan untuk dijalankan melalui Tableau Server agar dapat

diakses dengan performa yang sesuai. Untuk pengujian lokal, digunakan subset data.

5. **Perubahan Mode Koneksi:** Mode koneksi pada Tableau diubah dari `Live` ke `Extract` untuk meningkatkan efisiensi pemuatan data dan menghindari beban berlebih saat visualisasi.



UMMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA