

Laporan Analisis Kebutuhan Bisnis dan Teknis Perancangan Data Warehouse di Sektor Pemerintahan pada Dinas Kependudukan dan Pencatatan Sipil (Disdukcapil)



Disusun Oleh:

Sesilia Putri Subandi	122450012
Oktavia Nurwinda Puspitasari	122450041
Safitri	122450071
Baruna Abirawa	122450097
Dinda Nababan	122450120

**Program Studi Sains Data
Fakultas Sains
Institut Teknologi Sumatera
Lampung Selatan
2025**

1. Alur Aliran Data

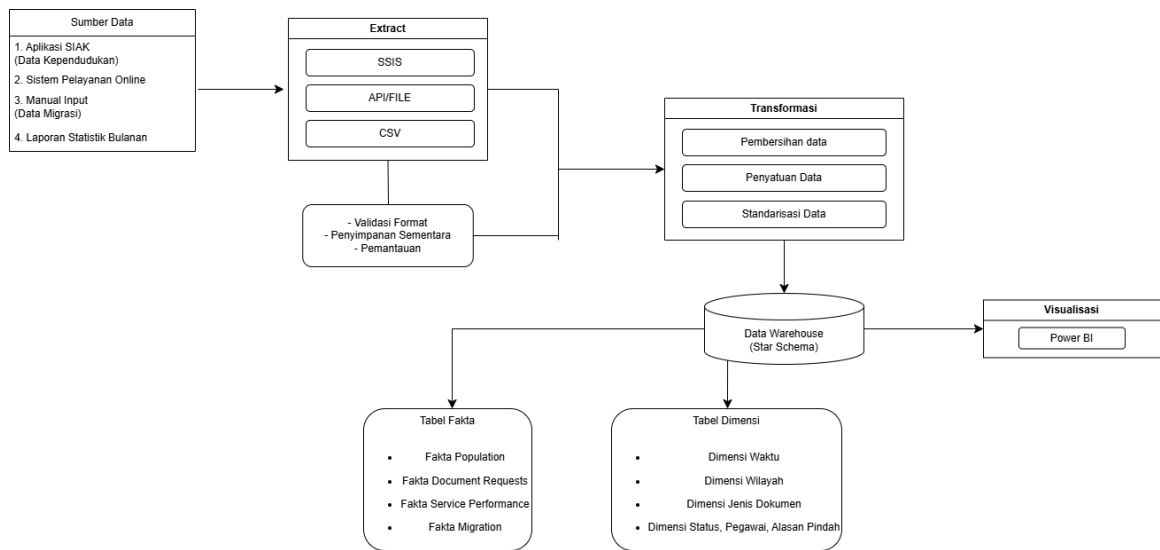


Diagram alur data ini menggambarkan proses pengolahan data dari berbagai sumber menuju Data Warehouse Disdukcapil dengan metode ETL (Extract, Transform, Load) menggunakan SSIS (SQL Server Integration Services) sebagai platform utama. Diagram dimulai dengan tiga sumber data utama, yaitu Aplikasi SIAK yang menyediakan data kependudukan dan peristiwa vital seperti NIK, nama, alamat, usia, dan jenis kelamin. Manual Input dari Kantor Kecamatan yang mencatat data migrasi penduduk, termasuk wilayah asal, tujuan, dan alasan perpindahan serta Laporan Statistik Bulanan yang berisi data validasi dan agregasi penduduk, seperti total jumlah penduduk dan statistik layanan.

Pada tahap Extract, SSIS digunakan untuk mengambil data dari sumber menggunakan metode koneksi seperti OLEDB untuk terhubung dengan database SQL Server atau sumber lainnya, API/File untuk mengambil data dalam format JSON, XML, Excel, serta CSV sebagai metode lain untuk mengambil data dari file statis. Data yang diekstrak disimpan sementara di staging area dan divalidasi formatnya. Pemantauan dilakukan menggunakan SSIS Logs dan SQL Profiler untuk memastikan proses berjalan dengan baik.

Setelah data diekstrak, tahap Transformasi dilakukan dengan tiga proses utama, yaitu Pembersihan Data untuk menghapus data duplikat dan mengisi nilai kosong (null), Penyatuan Data untuk menggabungkan data dari berbagai sumber menjadi satu format standar, dan Standarisasi Data untuk memastikan data memiliki format seragam (tanggal, angka, teks) sehingga mudah dianalisis.

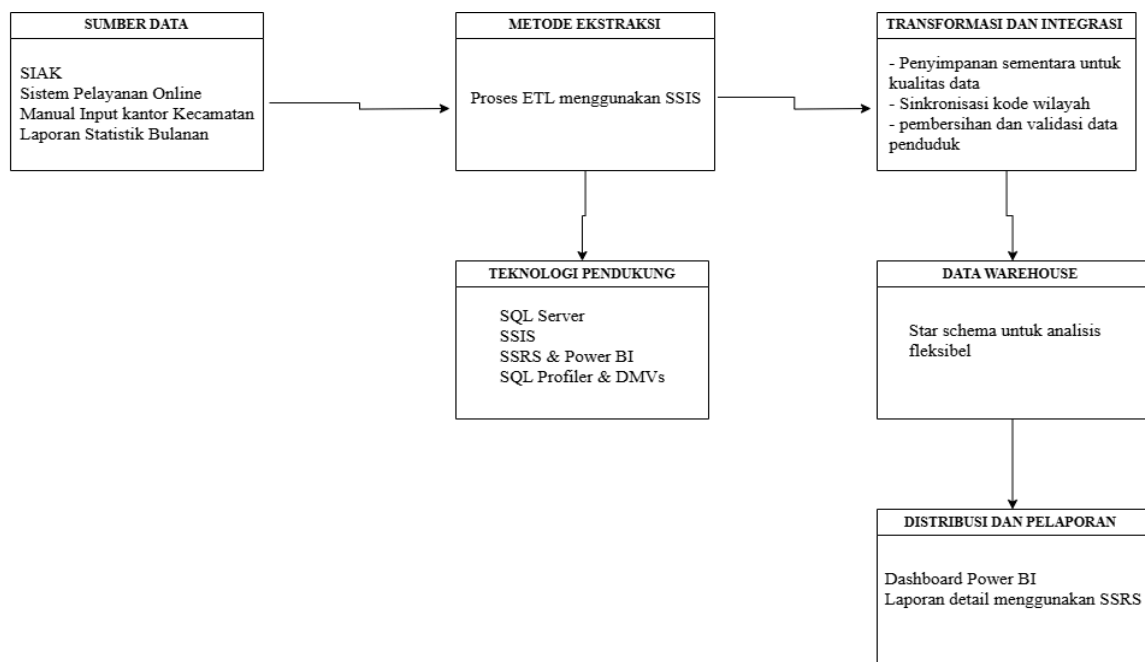
Selanjutnya, data yang sudah bersih dan distandarisasi dimuat ke dalam Data Warehouse yang menggunakan model Star Schema. Data dibagi menjadi dua bagian utama: Tabel Fakta yang menyimpan data kuantitatif termasuk Fakta Population (Jumlah Penduduk), Document Requests (Permohonan Dokumen), Service Performance (Kinerja Layanan), dan Migration (Data Migrasi); serta Tabel Dimensi yang berisi informasi deskriptif seperti Dimensi Waktu, Wilayah, Jenis Dokumen, Status, Pegawai, dan Alasan Pindah.

Tahap terakhir adalah Visualisasi, di mana data yang telah dimuat ke Data Warehouse divisualisasikan menggunakan Power BI. Power BI dipilih karena mampu menghasilkan dashboard interaktif dengan berbagai filter dan analisis data, termasuk analisis penduduk, permohonan dokumen, dan kinerja layanan. Diagram ini mencerminkan alur pengolahan data yang efisien dan terstruktur, memanfaatkan SSIS untuk proses ETL dan Power BI untuk analisis data secara interaktif.

2. Arsitektur

Dalam mendukung integrasi data kependudukan yang tersebar di berbagai sistem dan meningkatkan kualitas pengambilan keputusan berbasis data, Dinas Kependudukan dan Pencatatan Sipil (Disdukcapil) memerlukan sistem pengelolaan data yang andal dan terstruktur. Oleh karena itu, perancangan sistem data warehouse dilakukan dengan pendekatan arsitektur tiga lapis (Three-Tier Architecture), yang memisahkan tahapan ekstraksi dan validasi data (staging area), proses transformasi ke dalam skema bintang (star schema), serta penyajian data kepada pengguna akhir melalui data mart atau tampilan agregat. Pemisahan ini memungkinkan efisiensi proses, skalabilitas sistem, serta kemudahan dalam pengelolaan jangka panjang.

Data diambil dari berbagai sumber, seperti SIAK, aplikasi layanan mandiri, serta file lokal, menggunakan SSIS sebagai alat ekstraksi. Setelah melalui proses validasi dan transformasi, data dimuat ke dalam data warehouse yang dirancang dengan tabel fakta dan dimensi. Informasi ini kemudian disajikan melalui SSRS dan Power BI dalam bentuk laporan dan dashboard analitik. Dengan pendekatan ini, Disdukcapil dapat memastikan integritas data, mempercepat proses analisis, serta mendukung transparansi dan akuntabilitas pelayanan publik secara digital dan berkelanjutan.



1. Aliran Data dan Integrasi Sistem

- Sumber Utama Data: Sistem Informasi Kependudukan Nasional (SIAK), Aplikasi Pelayanan online, manual input kantor kecamatan, laporan statistik bulanan dan format data eksternal (CSV, Excel) dari berbagai wilayah administratif.
- Metode Ekstraksi: Data dikumpulkan melalui proses ETL menggunakan SSIS, dengan koneksi ke API internal, database relasional, dan file input manual.
- Area Penyangga (Staging): Digunakan sebagai media penyimpanan sementara untuk memastikan kualitas data sebelum transformasi lanjutan.
- Transformasi dan Integrasi: Proses pembersihan data meliputi normalisasi nama, validasi data penduduk, serta penyelarasan kode wilayah.
- Gudang Data (Data Warehouse): Dirancang menggunakan pendekatan Star Schema, mencakup tabel fakta terkait layanan dan tabel dimensi untuk analisis yang fleksibel.
- Distribusi dan Pelaporan: Visualisasi dilakukan melalui dashboard Power BI serta laporan detail menggunakan SSRS untuk pelaporan periodik ke pemangku kepentingan.

2. Rangkaian Proses ETL (Extract, Transform, Load)

- Extract: Mengambil data dari berbagai sistem sumber di pusat dan daerah, termasuk data permohonan layanan, pembaruan data, dan dokumen kependudukan.
- Transform: Melakukan konsolidasi, standarisasi format, dan klasifikasi jenis layanan.
- Load: Menyimpan hasil akhir transformasi ke dalam data warehouse dengan struktur skema bintang yang mendukung analisis multidimensi.

3. Teknologi Pendukung Sistem

- SQL Server: Digunakan sebagai platform utama penyimpanan dan pemrosesan data.
- SSIS (SQL Server Integration Services): Menangani seluruh proses ETL secara otomatis dan terjadwal.
- SSMS & SSDT: Untuk administrasi database dan pengelolaan skema serta versi.
- SSRS & Power BI: Alat pelaporan dan visualisasi data untuk analisis real-time dan evaluasi kinerja pelayanan.
- SQL Profiler & Dynamic Management Views (DMVs): Untuk pemantauan dan optimasi performa kueri dan sistem.

4. Desain Star Schema dalam Data Warehouse Disdukcapil

- Tabel Fakta:
 1. Population
 2. Document_Requests
 3. Service_Performance
 4. Migration

- Tabel Dimensi:
 1. Dim_time
 2. Dim_wilayah
 3. Dim_gender
 4. Dim_age_group
 5. Dim_jenis_dokumen
 6. Dim_status
 7. Dim_jenis_layanan
 8. Dim_pegawai
 9. Dim_alasan_pindah
- 5. Pengolahan dan Visualisasi Data
 - Data diproses harian/mingguan untuk menyajikan insight aktual.
 - Power BI digunakan untuk menyajikan analitik geografis, performa pelayanan, dan tren permohonan dokumen.
 - Laporan SSRS digunakan untuk rekapitulasi tahunan, laporan kinerja unit, serta akuntabilitas pelayanan publik.
- 6. Keamanan Sistem dan Pengawasan Kinerja
 - SQL Server Profiler dan DMVs digunakan untuk mendeteksi potensi perlambatan akses atau kueri tidak efisien.
 - SQL Server Agent menjalankan proses ETL terjadwal dan pemantauan kesehatan sistem.
 - Strategi Optimasi: Indexing dan partisi data berdasarkan dimensi waktu dan lokasi untuk akses cepat dan terstruktur.
- 7. Kebutuhan Analitik dan Pelaporan
 - Analisis Pertumbuhan dan Persebaran Penduduk
 - Evaluasi Pelayanan Publik
 - Monitoring Data Vital (Kelahiran, Kematian, Perkawinan, Perceraian)
 - Pengelolaan SDM dan Anggaran Operasional.

3. ETL Pipeline

Proses ETL (Extract, Transform, Load) pada sistem data kependudukan ini dirancang untuk mengintegrasikan berbagai sumber data menjadi satu basis data yang siap dianalisis dalam gudang data (data warehouse). Pada tahap Extract, data diambil dari beragam sumber seperti Aplikasi SIAK, Sistem Pelayanan Online, Manual Input Kantor Kecamatan, dan Laporan Statistik Bulanan. Pengambilan data dilakukan menggunakan koneksi OLEDB atau melalui Flat File Source pada SSIS (SQL Server Integration Services), tergantung pada format sumber seperti Excel, CSV, atau API.

Selanjutnya, pada tahap Transform, data yang telah diekstraksi melewati berbagai proses pembersihan dan transformasi. Validasi dilakukan terhadap format tanggal dan kode wilayah agar sesuai dengan standar sistem. Proses lookup digunakan untuk mencocokkan nilai-nilai foreign key ke dalam tabel referensi. Selain itu, dilakukan conditional split untuk memisahkan alur data berdasarkan kriteria tertentu (misalnya berdasarkan jenis dokumen atau wilayah), serta derived column untuk membentuk kolom baru dari data

yang sudah ada, seperti menggabungkan nama depan dan belakang atau menghitung usia dari tanggal lahir.

Pada tahap Load, data yang telah dibersihkan dan ditransformasikan dimuat terlebih dahulu ke dalam tabel-tabel dimensi (misalnya dimensi waktu, wilayah, dan jenis layanan), kemudian dilanjutkan dengan pemuatan ke tabel fakta yang menyimpan data transaksi atau peristiwa (misalnya permohonan dokumen, perpindahan penduduk, dan statistik layanan). Proses pemuatan ini dijadwalkan secara otomatis menggunakan SQL Server Agent, sehingga integrasi data berjalan berkala sesuai frekuensi update masing-masing sumber data, seperti real-time, harian, mingguan, maupun bulanan.

Dengan pendekatan ETL ini, data dari berbagai sumber yang memiliki struktur dan frekuensi yang berbeda dapat diolah menjadi informasi yang konsisten dan siap digunakan untuk analisis dan pelaporan kinerja layanan kependudukan.

4. Alat

Dalam membangun sistem gudang data (data warehouse) untuk Dinas Kependudukan dan pencatatan Sipil (Disdukcapil), pemilihan alat dan teknologi yang tepat merupakan faktor kunci dalam menjamin keberhasilan integrasi data, efisiensi proses ETL, akurasi laporan, serta kemudahan dalam pemantauan dan analisis kinerja pelayanan publik. Sistem dirancang dengan memanfaatkan ekosistem Microsoft SQL Server sebagai platform utama, serta Power BI sebagai alat visualisasi.

Alat-alat dalam SQL Server seperti SSMS, SSIS, SSDT, dan SSRS memungkinkan pengguna dari berbagai level teknis untuk berkontribusi dalam proses pembangunan gudang data.

4.1 Alat Pemodelan dan Desain Skema

4.1.1 SQL Server Management Studio (SSMS)

SSMS digunakan sebagai alat utama dalam pembuatan dan pengolahan skema database SQL Server. Di dalam proyek Disdukcapil, SSMS digunakan untuk:

- Membuat tabel fakta (fact_population, fact_document_requests, fact_migration) dan tabel dimensi (dim_time, dim_wilayah, dim_jenis_layanan, dll.).
- Menjalankan query SQL untuk mendefinisikan relasi antar tabel (FK), membuat indeks, serta validasi struktur skema.
- Memvisualisasikan hubungan antar entitas melalui fitur Database Diagrams, sehingga memudahkan pemahaman struktur data oleh tim pengembang dan analis.

4.1.2. SQL Server Data Tools (SSDT)

SSDT adalah ekstensi Visual Studio untuk pengembangan proyek basis data. SSDT digunakan untuk:

- Mendesain skema database secara terstruktur.
- Mendukung version control menggunakan Git, memudahkan pelacakan perubahan skema dari waktu ke waktu.

- Melakukan schema compare antara versi pengembangan dan produksi, sehingga menjamin integritas struktur saat deployment ke lingkungan operasional Disdukcapil.

4.3 Alat Pemantauan dan Optimasi

4.3.1 SQL Server Profiler

SQL Server Profiler adalah alat pelacak aktivitas server SQL yang sangat berguna dalam proses debugging dan optimasi. Profiler digunakan untuk memantau aktivitas server SQL, seperti:

- Mendeteksi query yang lambat dieksekusi saat pengguna mengakses laporan jumlah penduduk atau permohonan dokumen.
- Mengidentifikasi deadlock dan transaksi yang terlalu panjang.
- Menganalisis penggunaan indeks dan beban server pada jam-jam sibuk

4.3.2 Dynamic Management Views (DMVs)

DMVs memberikan kemampuan pemantauan berbasis query, meliputi:

- Pemantauan konsumsi sumber daya (CPU, I/O).
- Melacak tabel dan indeks yang paling sering diakses.
- Mengevaluasi statistik kinerja sistem.

4.4 Alat Visualisasi dan Pelaporan

4.4.1 SQL Server Reporting Services (SSRS)

SSRS digunakan untuk menyusun laporan formal berkala, seperti:

- Jumlah penduduk per wilayah dan kelompok umur.
- Statistik pengajuan dokumen kependudukan (KTP, KK, Akta).
- Ringkasan layanan publik dan kinerja pegawai berdasarkan wilayah dan waktu.
- Laporan dapat dikirim otomatis melalui email ke kepala dinas atau pimpinan.

4.4.2 Power BI (Integrasi Tambahan)

Power BI digunakan untuk membuat dashboard interaktif, seperti:

- Peta sebaran penduduk per kecamatan.
- Grafik permohonan dokumen harian atau bulanan.
- Evaluasi waktu rata-rata layanan per unit kerja.
- Visualisasi tren migrasi masuk dan keluar perbulan.

5. Script Query

Membangun sistem data warehouse menjadi langkah strategis bagi Disdukcapil dalam menyatukan berbagai data pelayanan administrasi kependudukan. Melalui skrip SQL yang dirancang khusus, instansi ini dapat membentuk struktur data yang mencakup tabel-tabel fakta dan dimensi, yang saling terhubung untuk merepresentasikan informasi penting seperti jumlah penduduk, permohonan dokumen, migrasi penduduk, hingga performa layanan publik. Pendekatan ini memungkinkan analisis yang lebih menyeluruh, sehingga pengambilan keputusan oleh pimpinan dapat dilakukan secara lebih cepat, akurat, dan berbasis data yang terintegrasi lintas wilayah dan waktu.

5.1 Implementasi Struktur Data Warehouse (DDL)

Pada bagian ini, dilakukan implementasi skema star schema melalui rangkaian skrip SQL yang dirancang untuk membentuk infrastruktur basis data mencakup:

- Tabel staging sebagai tempat penampungan awal data mentah dari berbagai sumber (SIAK, Pelayanan Online, Input Manual, Statistik Bulanan),
- Tabel dimensi sebagai referensi atribut deskriptif untuk mendukung analisis,
- Tabel fakta yang berisi data transaksional atau agregat dari masing-masing domain kependudukan,
- Serta proses ETL (Extract, Transform, Load) dari staging ke warehouse utama.

Berikut merupakan beberapa skrip query yang ditampilkan:

```
-- =====
-- DATABASE & STAGING TABLES
-- =====
CREATE DATABASE DW_Disdukcapil;
GO

USE DW_Disdukcapil;
GO

-- STAGING: Population (SIAK)
CREATE TABLE staging_population (
    tanggal DATE,
    kode_wilayah NVARCHAR(20),
    gender NVARCHAR(10),
    age_group NVARCHAR(20),
    jumlah_penduduk INT
);

-- =====
-- DIMENSI
-- =====
CREATE TABLE dim_time (
    time_id INT PRIMARY KEY,
    tanggal DATE,
    hari INT,
    bulan INT,
    tahun INT,
```



```

    kuartal INT,
    semester INT
);

CREATE TABLE dim_wilayah (
    wilayah_id INT IDENTITY(1,1) PRIMARY KEY,
    kode_wilayah NVARCHAR(20),
    provinsi NVARCHAR(50),
    kabupaten_kota NVARCHAR(50),
    kecamatan NVARCHAR(50),
    kelurahan NVARCHAR(50)
);

CREATE TABLE dim_gender (
    gender_id INT IDENTITY(1,1) PRIMARY KEY,
    gender NVARCHAR(10)
);

CREATE TABLE dim_age_group (
    age_group_id INT IDENTITY(1,1) PRIMARY KEY,
    rentang_usia NVARCHAR(20)
);

-- =====
-- TABEL FAKTA
-- =====

CREATE TABLE fact_population (
    population_id INT IDENTITY(1,1) PRIMARY KEY,
    time_id INT,
    wilayah_id INT,
    gender_id INT,
    age_group_id INT,
    jumlah_penduduk INT
);

-- =====
-- ETL Staging - Dimensi
-- =====

# dim_time
INSERT INTO dim_time (time_id, tanggal, hari, bulan, tahun, kuartal, semester)
SELECT DISTINCT
    CONVERT(INT, FORMAT(tanggal, 'yyyyMMdd')),
    tanggal,
    DAY(tanggal),
    MONTH(tanggal),
    YEAR(tanggal),
    DATEPART(QUARTER, tanggal),
    CASE
        WHEN MONTH(tanggal) <= 6 THEN 1
        ELSE 2
    END
FROM staging_population;

# dim_wilayah

```

```

INSERT INTO dim_wilayah (kode_wilayah, provinsi, kabupaten_kota, kecamatan, kelurahan)
SELECT DISTINCT kode_wilayah, provinsi, kabupaten_kota, kecamatan, kelurahan
FROM staging_wilayah;

# dim_gender, dim_age
INSERT INTO dim_gender (gender)
SELECT DISTINCT gender FROM staging_population;

INSERT INTO dim_age_group (rentang_usia)
SELECT DISTINCT age_group FROM staging_population;

-- =====
-- ETL Staging - Fakta
-- =====

# fact_population
INSERT INTO fact_population (time_id, wilayah_id, gender_id, age_group_id, jumlah_penduduk)
SELECT
    CONVERT(INT, FORMAT(SP.tanggal, 'yyyyMMdd')),
    W.wilayah_id,
    G.gender_id,
    A.age_group_id,
    SP.jumlah_penduduk
FROM staging_population SP
JOIN dim_wilayah W ON SP.kode_wilayah = W.kode_wilayah
JOIN dim_gender G ON SP.gender = G.gender
JOIN dim_age_group A ON SP.age_group = A.rentang_usia;

```

5.2 Query analytic

Berikut merupakan query analitik yang telah dimodifikasi untuk memenuhi kebutuhan khusus yang telah diidentifikasi:

```

-- =====
-- Query Analitik
-- =====

# Analisis Pertumbuhan dan Persebaran Penduduk
-- a. Pertumbuhan jumlah penduduk bulanan dan tahunan per kecamatan
SELECT T.year, T.month, W.nama_wilayah, SUM(F.jumlah) AS total_penduduk
FROM fact_population F
JOIN dim_time T ON F.time_id = T.time_id
JOIN dim_wilayah W ON F.wilayah_id = W.wilayah_id
GROUP BY T.year, T.month, W.nama_wilayah;

-- b. Distribusi penduduk berdasarkan kelompok umur, jenis kelamin
SELECT A.rentang_usia, G.gender, SUM(F.jumlah) AS total
FROM fact_population F
JOIN dim_age_group A ON F.age_group_id = A.age_group_id
JOIN dim_gender G ON F.gender_id = G.gender_id
GROUP BY A.rentang_usia, G.gender;

-- c. Tren migrasi masuk dan keluar antar wilayah
SELECT WA.nama_wilayah AS Asal, WT.nama_wilayah AS Tujuan, SUM(F.jumlah) AS total
FROM fact_migration F

```

```

JOIN dim_wilayah WA ON F.wilayah_asal_id = WA.wilayah_id
JOIN dim_wilayah WT ON F.wilayah_tujuan_id = WT.wilayah_id
GROUP BY WA.nama_wilayah, WT.nama_wilayah;

```

Evaluasi Pelayanan Publik

```

-- a. Waktu rata-rata penyelesaian layanan (diperluas jadi total skor kepuasan)
SELECT T.month, L.nama_layanan, AVG(S.skor) AS rata_skor
FROM fact_satisfaction S
JOIN dim_jenis_layanan L ON S.jenis_layanan_id = L.jenis_layanan_id
JOIN dim_time T ON S.time_id = T.time_id
GROUP BY T.month, L.nama_layanan;

```

```

-- b. Jumlah pengajuan layanan per jenis dokumen
SELECT D.nama_dokumen, COUNT(*) AS jumlah
FROM fact_document_requests F
JOIN dim_jenis_dokumen D ON F.jenis_dokumen_id = D.jenis_dokumen_id
GROUP BY D.nama_dokumen;

```

```

-- c. Kepuasan masyarakat berdasarkan wilayah
SELECT W.nama_wilayah, AVG(S.skor) AS rata_skor
FROM fact_satisfaction S
JOIN dim_wilayah W ON S.wilayah_id = W.wilayah_id
GROUP BY W.nama_wilayah;

```

Monitoring Data Vital

```

-- a. Jumlah peristiwa penting per bulan dan lokasi
SELECT T.month, V.jenis_peristiwa, W.nama_wilayah, SUM(V.jumlah) AS total
FROM fact_vital_event V
JOIN dim_time T ON V.time_id = T.time_id
JOIN dim_wilayah W ON V.wilayah_id = W.wilayah_id
GROUP BY T.month, V.jenis_peristiwa, W.nama_wilayah;

```

Pengelolaan SDM dan Anggaran

```

-- a. Jumlah pegawai aktif berdasarkan jabatan dan lokasi
SELECT P.nama_jabatan, W.nama_wilayah, COUNT(*) AS jumlah
FROM fact_service_performance F
JOIN dim_pegawai P ON F.pegawai_id = P.pegawai_id
JOIN dim_wilayah W ON F.wilayah_id = W.wilayah_id
GROUP BY P.nama_jabatan, W.nama_wilayah;

```

```

-- b. Evaluasi beban kerja pegawai per layanan
SELECT L.nama_layanan, P.nama_jabatan, AVG(F.beban_kerja) AS rata_beban
FROM fact_service_performance F
JOIN dim_jenis_layanan L ON F.jenis_layanan_id = L.jenis_layanan_id
JOIN dim_pegawai P ON F.pegawai_id = P.pegawai_id
GROUP BY L.nama_layanan, P.nama_jabatan;

```

-- c. Efisiensi anggaran

```

SELECT unit_kerja, program, SUM(anggaran) AS total_anggaran,
       SUM(realisasi) AS total_realisasi,
       SUM(realisasi) / NULLIF(SUM(anggaran),0) AS rasio_efisiensi
FROM fact_budget
GROUP BY unit_kerja, program;

```