

**TUGAS BESAR PERGUDANGAN DATA
PEMBANGUNAN DATA MART**

Domain: Satuan Pengawas Internal



Anggota Kelompok:

- | | | |
|----|-----------------------|-----------|
| 1. | Anggi Puspita Ningrum | 123450012 |
| 2. | Anadia Carana | 123450019 |
| 3. | Iqfina Haula Halika | 123450076 |
| 4. | Muhammad Dzikra | 123450124 |

**PROGRAM STUDI SAINS DATA
FAKULTAS SAINS
INSTITUT TEKNOLOGI SUMATERA
2025**

MISI 3: IMPLEMENTASI PRODUKSI

3.1 Production Deployment

Initial Data Load

- Execute full ETL untuk historical data
- Verify data integrity
- Document load statistics

3.2 Dashboard Development

1. Create Analytical Views

```
USE DM_SPI_DW;
GO

CREATE OR ALTER VIEW dbo.vw_Auditor_Performance_Summary AS
SELECT
    -- Dimensi Auditor
    a.Nama_Auditor,
    a.Tim_Audit,

    -- Measures
    COUNT(f.Fakta_SK) AS Total_Temuan_Ditemukan, -- Jumlah temuan yang ditemukan
    AVG(CAST(f.Usia_Rekomendasi_Hari AS DECIMAL(10, 2))) AS Rata_Rata_Usia_Rekomendasi_Hari,

    -- Metrik Tindak Lanjut
    SUM(CASE WHEN r.Status_Tindak_Lanjut = 'Closed' THEN 1 ELSE 0 END) AS
    Rekomendasi_Closed_Count,

    -- Persentase Penyelesaian Rekomendasi (KPI)
    CAST(SUM(CASE WHEN r.Status_Tindak_Lanjut = 'Closed' THEN 1 ELSE 0 END) * 100.0
        / NULLIF(COUNT(f.Fakta_SK), 0) AS DECIMAL(5, 2)) AS Persen_Rekomendasi_Selesai

FROM dbo.Fact_Temuan_Rekomendasi f
INNER JOIN dbo.Dim_Auditor a
    ON f.Auditor_SK = a.Auditor_SK
INNER JOIN dbo.Dim_Rekomendasi r
    ON f.Rekomendasi_SK = r.Rekomendasi_SK

WHERE
    a.IsCurrent = 1 -- Hanya sertakan auditor dengan status aktif saat ini
    -- Catatan: Jika Anda ingin historical performance, hapus a.IsCurrent = 1
GROUP BY
    a.Nama_Auditor,
    a.Tim_Audit;
```

```
GO
```

```
USE DM_SPI_DW;
```

```
GO
```

```
CREATE OR ALTER VIEW dbo.vw_Unit_Risk_Profile AS
```

```
SELECT
```

```
-- Dimensi Unit Kerja dan Waktu
```

```
u.Nama_Unit,
```

```
u.Jenis_Unit,
```

```
w.Tahun AS Tahun_Audit,
```

```
w>Nama_Kuartal,
```

```
-- Measures
```

```
COUNT(f.Fakta_SK) AS Total_Temuan,
```

```
SUM(f.Potensi_Kerugian_IDR) AS Total_Potensi_Kerugian_IDR,
```

```
AVG(f.Skor_Risiko_Temuan) AS Rata_Rata_Skor_Risiko
```

```
FROM dbo.Fact_Temuan_Rekomendasi f
```

```
INNER JOIN dbo.Dim_Unit_Kerja u
```

```
ON f.Unit_Kerja_SK = u.Unit_Kerja_SK
```

```
INNER JOIN dbo.Dim_Waktu w
```

```
ON f.Waktu_SK = w.Waktu_SK
```

```
WHERE
```

```
u.IsCurrent = 1 -- Hanya sertakan Unit Kerja yang statusnya aktif
```

```
GROUP BY
```

```
u.Nama_Unit,
```

```
u.Jenis_Unit,
```

```
w.Tahun,
```

```
w>Nama_Kuartal;
```

```
GO
```

```
USE DM_SPI_DW;
```

```
GO
```

```
-- VIEW 1: vw_Risk_Profile_Unit
```

```
-- Setara dengan vw_Student_Performance: Analisis performa/risiko per entitas (Unit Kerja)
```

```
CREATE OR ALTER VIEW dbo.vw_Risk_Profile_Unit
```

```
AS
```

```
SELECT
```

```
du.Kode_Unit,
```

```
du.Nama_Unit,
```

```
du.Jenis_Unit,
```

```
du.Kepala_Unit,
```

```
dt.Kategori_Risiko,
```

```

dt.Tingkat_Materialitas,
COUNT(f.Fakta_SK) AS Total_Temuan_Audit,
AVG(f.Skor_Risiko_Temuan) AS Avg_Skor_Risiko,
SUM(f.Potensi_Kerugian_IDR) AS Total_Potensi_Kerugian,
MAX(f.Usia_Rekomendasi_Hari) AS Max_Aging_Rekomendasi,
CAST(SUM(CASE WHEN dt.Tingkat_Materialitas = 'Material' THEN 1 ELSE 0 END) * 100.0
      / NULLIF(COUNT(f.Fakta_SK), 0) AS DECIMAL(5,2)) AS Persen_Temuan_Material,
-- Hitung Jumlah Rekomendasi yang Overdue (Overdue: Usia Rekomendasi Hari > 0)
SUM(CASE WHEN f.Usia_Rekomendasi_Hari > 0 THEN 1 ELSE 0 END) AS Total_Rekomendasi_Overdue
FROM
dbo.Fact_Temuan_Rekomendasi f
INNER JOIN
dbo.Dim_Unit_Kerja du ON f.Unit_Kerja_SK = du.Unit_Kerja_SK
INNER JOIN
dbo.Dim_Temuan dt ON f.Temuan_SK = dt.Temuan_SK
INNER JOIN
dbo.Dim_Rekomendasi dr ON f.Rekomendasi_SK = dr.Rekomendasi_SK
WHERE
du.IsCurrent = 1 -- Hanya unit kerja dengan versi dimensi yang sedang berlaku
AND dt.IsCurrent = 1 -- Hanya temuan dengan versi dimensi yang sedang berlaku
AND dr.IsCurrent = 1 -- Hanya rekomendasi dengan versi dimensi yang sedang berlaku
GROUP BY
du.Kode_Unit, du.Nama_Unit, du.Jenis_Unit, du.Kepala_Unit, dt.Kategori_Risiko, dt.Tingkat_Materialitas;
GO

```

```
-- VIEW 2: vw_Audit_Performance_Trend
-- Setara dengan vw_Program_Analytics: Analisis performa Audit/Unit (Tren Siklus Audit)
```

```

CREATE OR ALTER VIEW dbo.vw_Audit_Performance_Trend
AS
SELECT
dsa.Jenis_Audit,
dsa.Status_Siklus,
dw.Tahun AS Tahun_Audit,
da.Tim_Audit,

COUNT(DISTINCT f.Unit_Kerja_SK) AS Total_Unit_Diaudit,
COUNT(DISTINCT f.Temuan_SK) AS Total_Temuan_Unik,
SUM(f.Jumlah_Temuan) AS Total_Rekomendasi,
AVG(f.Skor_Risiko_Temuan) AS Avg_Skor_Risiko_Siklus,
SUM(f.Potensi_Kerugian_IDR) AS Total_Potential_Loss_Siklus,

-- Hitung Persentase Rekomendasi Selesai (Closed)
CAST(SUM(CASE WHEN dr.Status_Tindak_Lanjut = 'Closed' THEN 1 ELSE 0 END) * 100.0
      / COUNT(f.Fakta_SK) AS DECIMAL(5,2)) AS Persen_Rekomendasi_Selesai
FROM

```

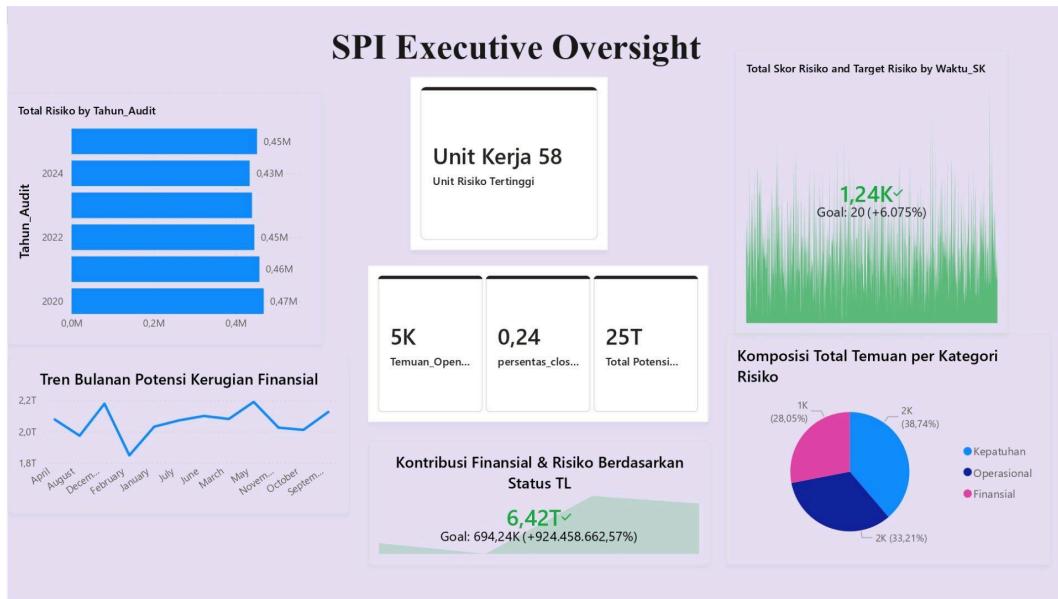
```

dbo.Fact_Temuan_Rekomendasi f
INNER JOIN
    dbo.Dim_Siklus_Audit dsa ON f.Siklus_Audit_SK = dsa.Siklus_Audit_SK
INNER JOIN
    dbo.Dim_Waktu dw ON f.Waktu_SK = dw.Waktu_SK
LEFT JOIN
    dbo.Dim_Auditor da ON f.Auditor_SK = da.Auditor_SK
INNER JOIN
    dbo.Dim_Rekomendasi dr ON f.Rekomendasi_SK = dr.Rekomendasi_SK
WHERE
    da.IsCurrent = 1 -- Pastikan dimensi auditor yang dipakai adalah yang Current (jika digunakan untuk filtering)
GROUP BY
    dsa.Jenis_Audit, dsa.Status_Siklus, dw.Tahun, da.Tim_Audit;
GO

```

2. Design Power BI Dashboards

Dashboard 1 : SPI Executive Oversight



- KPI Cards : Total Potensi kerugian, persentase selesai, temuan open ovedue, unit risiko tertinggi
- KPI : total skor risiko, waktu sk, target risiko, potensi kerugian IDR, status tindak lanjut, skor risiko temuan
- Bar Chart total risiko : Tahun Audit, Total risiko
- Pie Chart komposisi total temuan : kategori risiko, temuan open overdue
- Line Chart tren bulanan potensi kerugian finansial: Bulan, total potensi kerugian

Dashboard ini dirancang sebagai Ringkasan Eksekutif untuk memberikan pandangan menyeluruh (*helicopter view*) kepada pimpinan (Kepala SPI atau Rektorat) mengenai kesehatan risiko organisasi secara *real-time*.

1. Kartu KPI Utama (KPI Cards)

- Unit Kerja 58 (Unit Risiko Tertinggi): Mengidentifikasi unit kerja mana yang memiliki akumulasi skor risiko paling besar saat ini. Ini membantu pimpinan memprioritaskan area mana yang butuh perhatian khusus.
- 5K (Temuan Open): Menunjukkan jumlah total temuan audit yang statusnya masih "Terkunci" (belum diselesaikan). Angka 5.000 menunjukkan beban kerja tindak lanjut yang masih harus dikerjakan.
- 0,24 (Percentase Closed): Menunjukkan rasio penyelesaian rekomendasi. Angka ini mengindikasikan seberapa efektif unit kerja dalam mendukung pelaksanaan rekomendasi audit.
- 25T (Total Potensi Kerugian): Angka estimasi dampak finansial (dalam Triliun) dari temuan yang ada. Ini adalah metrik *additive* untuk melihat besaran risiko finansial organisasi.

2. Grafik Batang: Total Risiko by Tahun Audit

- Fungsi: Menampilkan tren total akumulasi risiko dari tahun ke tahun (2020 - 2024).
- Analisis: Grafik ini membantu eksekutif melihat apakah tren risiko organisasi menurun atau justru meningkat setiap tahunnya. Pada gambar, terlihat risiko relatif stabil di angka sekitar 0,45M - 0,47M.

3. Grafik Garis: Tren Bulanan Potensi Kerugian Finansial

- Fungsi: Memvisualisasikan fluktuasi potensi kerugian finansial dari bulan ke bulan.
- Analisis: Grafik ini berguna untuk mendeteksi anomali musiman atau lonjakan risiko finansial pada bulan-bulan tertentu.

4. Grafik : Total Skor Risiko dan Target

- Fungsi: Membandingkan skor risiko aktual (1,24K) dengan target risiko yang ditetapkan (Goal: 20).
- Analisis: Grafik ini menunjukkan penyimpangan (*deviation*) dari toleransi risiko yang ditetapkan perusahaan. Angka persentase (+6.075%) menunjukkan bahwa risiko saat ini jauh melampaui batas aman yang ditargetkan.

5. Pie Chart: Komposisi Total Temuan per Kategori Risiko

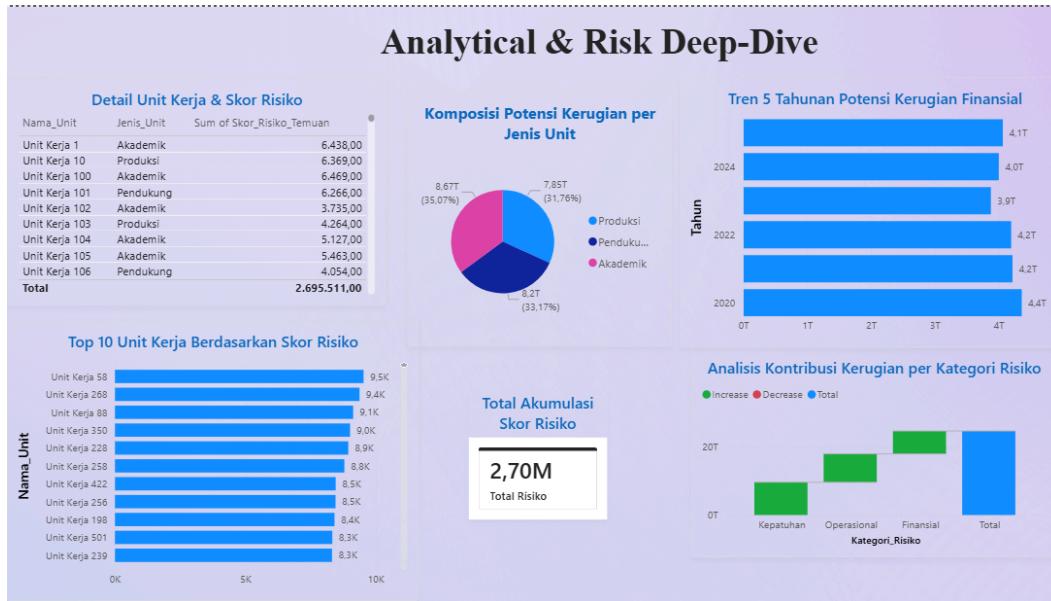
- Fungsi: Memecah total temuan berdasarkan jenis risikonya:
 - Kepatuhan (Biru Muda): Pelanggaran aturan/regulasi.
 - Operasional (Biru Tua): Masalah pada proses bisnis.
 - Finansial (Pink): Masalah yang berdampak pada keuangan.
- Analisis: Membantu pimpinan memahami "sifat" masalah di organisasi, apakah dominan masalah uang, aturan, atau teknis operasional.

6. Indikator Target: Kontribusi Finansial & Risiko

- Fungsi: Menampilkan angka 6,42T (Triliun) yang kemungkinan merujuk pada nilai finansial yang berhasil diselamatkan atau dikelola berdasarkan status Tindak Lanjut (TL).

- Analisis: Membandingkan pencapaian dengan *Goal* (694,24K), menunjukkan dampak finansial yang signifikan dari proses audit yang dilakukan.

Dashboard 2 : Analytical & Risk Deep-Dive



- Table detail unit kerja & skor risiko: Nama unit, jenis Unit, skor risiko temuan
- Bar chart Top ten unit kerja berdasarkan skor risiko : Nama unit, Skor risiko temuan
- Pie Chart komposisi potensi kerugian jenis unit : Jenis unit, Potensi kerugian IDR,
- KPI Cards akumulasi skor risiko : Total Risiko
- Bar Chart tren 5 tahunan potensi kerugian : Tahun, potensi kerugian IDR
- Waterfall Chart analisis kontribusi kerugian : Kategori risiko, potensi kerugian IDR

Dashboard ini dirancang untuk analisis mendalam (deep-dive). Tujuannya adalah membantu *Manager Audit* atau *Analis Risiko* untuk mengidentifikasi akar masalah, membandingkan kinerja antar unit, dan melihat komposisi risiko secara lebih granular.

1. Tabel: Detail Unit Kerja & Skor Risiko

- Fungsi: Menyajikan data tabular yang merinci setiap unit kerja, jenis unitnya (Akademik, Produksi, Pendukung), dan total skor risiko mereka.
- Analisis: Tabel ini memungkinkan pengguna untuk melihat angka pasti (*exact values*) per unit. Ini sangat berguna jika auditor ingin mencari unit spesifik (misalnya "Unit Kerja 102") untuk melihat profil risikonya secara cepat tanpa harus menebak dari grafik.

2. Grafik Batang: Top 10 Unit Kerja Berdasarkan Skor Risiko

- Fungsi: Memperlihatkan peringkat 10 unit kerja dengan akumulasi skor risiko tertinggi.

- Analisis: Ini adalah alat prioritas audit. Terlihat Unit Kerja 58 berada di posisi paling atas (sekitar 9.5K), diikuti oleh Unit Kerja 268. Auditor akan menggunakan grafik ini untuk menentukan unit mana yang harus segera diaudit ("Red Flags").

3. Pie Chart: Komposisi Potensi Kerugian per Jenis Unit

- Fungsi: Memvisualisasikan proporsi potensi kerugian finansial berdasarkan kategori fungsional unit kerja:
 - Produksi (Biru): Unit yang menghasilkan produk/jasa.
 - Pendukung (Ungu): Unit administrasi/support.
 - Akademik (Pink): Fakultas atau program studi.
- Analisis: Grafik ini membantu manajemen melihat sektor mana yang paling "boros" atau berisiko secara finansial. Pada gambar, terlihat distribusi risiko cukup merata di ketiga sektor tersebut (masing-masing sekitar 30-35%).

4. Kartu KPI: Total Akumulasi Skor Risiko

- Fungsi: Menampilkan angka tunggal 2,70M (Juta).
- Analisis: Ini adalah angka agregat dari seluruh risiko yang terdeteksi di dalam sistem saat ini. Angka ini menjadi *baseline* untuk mengukur apakah upaya mitigasi risiko secara keseluruhan berhasil menurunkan skor ini di masa depan.

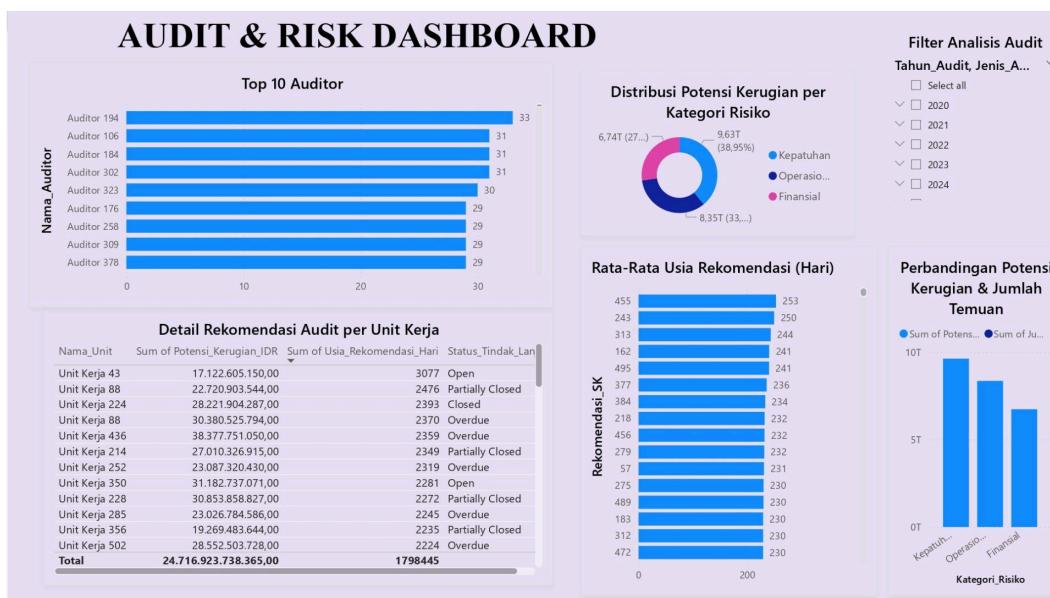
5. Grafik Batang: Tren 5 Tahunan Potensi Kerugian Finansial

- Fungsi: Menampilkan pergerakan potensi kerugian finansial dalam jangka panjang (2020 - 2024).
- Analisis: Berbeda dengan Dashboard 1 yang fokus pada tren bulanan, grafik ini melihat *historical performance* jangka panjang. Jika batangnya terus meninggi dari tahun ke tahun, berarti kontrol internal organisasi semakin memburuk. Di sini terlihat nilainya fluktuatif di kisaran 4 Triliun per tahun.

6. Waterfall Chart: Analisis Kontribusi Kerugian per Kategori Risiko

- Fungsi: Grafik air terjun (*waterfall*) ini menunjukkan bagaimana setiap kategori risiko (Kepatuhan, Operasional, Finansial) berkontribusi terhadap total kerugian.
- Analisis:
 - Balok hijau menunjukkan penambahan nilai kerugian dari masing-masing kategori.
 - Balok biru di ujung kanan adalah Total.
 - Grafik ini sangat bagus untuk memvisualisasikan "penyumbang terbesar". Misalnya, terlihat bahwa risiko Finansial dan Operasional memberikan kontribusi yang sangat signifikan terhadap total kerugian dibandingkan risiko Kepatuhan.

Dashboard 3 : AUDIT & RISK DASHBOARD



- Table Detail rekomendasi audit : Nama unit, potensi kerugian IDR, usia rekomendasi hari, status tindak lanjut
- Bar Chart Top 10 auditor : Nama auditor, Jumlah temuan,
- Donut Chart Distribusi potensi kerugian per kategori risiko : kategori risiko, potensi per risiko
- Bar Chart Rata-rata usia rekomendasi : Rekomendasi SK, rata usia rekomendasi
- Column Chart perbandingan potensi kerugian & jumlah temuan : kategori risiko, potensi kerugian, jumlah temuan

Dashboard 3 ini berfungsi sebagai Dashboard Operasional Audit. Fokus utamanya adalah memantau kinerja tim auditor dan status penyelesaian rekomendasi secara mendetail.

1. Grafik Batang: Top 10 Auditor

- Fungsi: Menampilkan peringkat 10 auditor yang paling produktif berdasarkan jumlah temuan yang dihasilkan.
- Analisis:
 - Terlihat Auditor 194 berada di peringkat teratas dengan 33 temuan, diikuti oleh Auditor 106 dan 184.
 - Grafik ini digunakan oleh Manajer Audit untuk mengevaluasi kinerja tim, menentukan beban kerja, atau memberikan insentif kinerja.

2. Donut Chart: Distribusi Potensi Kerugian per Kategori Risiko

- Fungsi: Menunjukkan proporsi nilai kerugian finansial yang dibagi berdasarkan kategori risiko (Finansial, Operasional, Kepatuhan).
- Analisis:
 - Kategori Operasional (warna biru tua) dan Finansial (warna pink) terlihat mendominasi porsi kerugian.

- Visual ini membantu memfokuskan sumber daya audit ke area yang "paling mahal" dampaknya bagi organisasi.

3. Tabel: Detail Rekomendasi Audit per Unit Kerja

- Fungsi: Ini adalah komponen paling operasional. Tabel ini menyajikan daftar unit kerja lengkap dengan total potensi kerugian, rata-rata usia rekomendasi, dan status tindak lanjutnya (Open, Closed, Overdue).
- Analisis:
 - Pengguna dapat melihat Unit Kerja mana yang memiliki status "Overdue" (terlambat) atau "Open" dengan nilai kerugian yang besar.
 - Contoh: Unit Kerja 43 memiliki status "Open" dengan usia rekomendasi yang sangat tua (3.077 hari), ini adalah *red flag* yang memerlukan intervensi segera.

4. Grafik Batang: Rata-Rata Usia Rekomendasi

- Fungsi: Menampilkan durasi (dalam hari) seberapa lama rekomendasi audit tertentu (berdasarkan ID Rekomendasi_SK) belum diselesaikan.
- Analisis:
 - Semakin panjang batangnya, semakin lambat respon perbaikan terhadap temuan tersebut.
 - Data ini menyoroti rekomendasi spesifik yang "macet" atau terabaikan dalam waktu lama (di atas 200-400 hari).

5. Column Chart: Perbandingan Potensi Kerugian & Jumlah Temuan

- Fungsi: Grafik ini membandingkan dua variabel sekaligus untuk setiap kategori risiko: Jumlah Temuan (frekuensi kejadian) vs Potensi Kerugian (dampak uang).
- Analisis:
 - Grafik ini menjawab pertanyaan: *"Apakah area yang paling sering bermasalah juga yang paling merugikan?"*
 - Terlihat bahwa kategori Operasional memiliki potensi kerugian yang sangat tinggi (batang biru tua tinggi) dibandingkan kategori lainnya.

6. Filter Analisis Audit

- Fungsi: Fitur interaktif yang memungkinkan pengguna memfilter seluruh data di dashboard berdasarkan Tahun Audit (2020-2024). Ini memungkinkan analisis historis untuk melihat kinerja auditor atau penyelesaian rekomendasi di tahun tertentu.

3.3 Security Implementation

1. Create User Roles

```
-- 1. Create Database Roles
CREATE ROLE db_kepala_spi;
```

```

CREATE ROLE db_analis_audit;
CREATE ROLE db_viewer;
CREATE ROLE db_etl_operator;
GO

-- 2. Grant Permissions (Diperbarui dengan Views Analitik Anda)
-- Peran Kepala SPI
GRANT SELECT ON SCHEMA :: dbo TO db_kepala_spi;
GRANT EXECUTE ON SCHEMA :: dbo TO db_kepala_spi;
-- Peran Analis Audit
GRANT SELECT ON SCHEMA :: dbo TO db_analis_audit;
GRANT SELECT, INSERT, UPDATE, DELETE ON SCHEMA :: stg TO db_analis_audit;

-- Peran Viewer (Hanya akses baca ke views analitik dan Dimensi Waktu)
GRANT SELECT ON dbo.vw_Risk_Profile_Unit TO db_viewer;
GRANT SELECT ON dbo.vw_Audit_Performance_Trend TO db_viewer;
GRANT SELECT ON dbo.Dim_Waktu TO db_viewer;

-- Peran ETL Operator
GRANT EXECUTE ON dbo.usp_Master_ETL_Load TO db_etl_operator; -- Mengasumsikan nama SP ETL Master
GRANT SELECT, INSERT, UPDATE, DELETE ON SCHEMA :: stg TO db_etl_operator;
GRANT INSERT, UPDATE ON SCHEMA :: dbo TO db_etl_operator;
GO

```

2. Create Users and Assign Roles

```

-- 1. Create SQL Logins (Ganti dengan password yang kuat!)
CREATE LOGIN kepala_spi WITH PASSWORD = 'StrongP@ssw0rd!';
CREATE LOGIN analis_user WITH PASSWORD = 'StrongP@ssw0rd!';
CREATE LOGIN viewer_user WITH PASSWORD = 'StrongP@ssw0rd!';
CREATE LOGIN etl_service WITH PASSWORD = 'StrongP@ssw0rd!';
GO

```

```

-- 2. Create Database Users
CREATE USER kepala_spi FOR LOGIN kepala_spi;
CREATE USER analis_user FOR LOGIN analis_user;
CREATE USER viewer_user FOR LOGIN viewer_user;
CREATE USER etl_service FOR LOGIN etl_service;
GO

```

```

-- 3. Assign Users to Roles
ALTER ROLE db_kepala_spi ADD MEMBER kepala_spi;
ALTER ROLE db_analis_audit ADD MEMBER analis_user;
ALTER ROLE db_viewer ADD MEMBER viewer_user;
ALTER ROLE db_etl_operator ADD MEMBER etl_service;
GO

```

3. Implement Data Masking

```
-- Data Masking untuk kolom ID Auditor (Natural Key)
ALTER TABLE dbo.Dim_Auditor
    ALTER COLUMN ID_Sistem_Sumber ADD MASKED WITH (FUNCTION = 'partial(0, "XX-XXX-", 4)');

-- Data Masking untuk nama Penanggung Jawab di Dimensi Rekomendasi
ALTER TABLE dbo.Dim_Rekomendasi
    ALTER COLUMN Penanggung_Jawab ADD MASKED WITH (FUNCTION = 'default()');

-- Memberikan izin UNMASK hanya kepada Kepala SPI dan Analis
GRANT UNMASK TO db_kepala_spi;
GRANT UNMASK TO db_analisis_audit;
GO
```

Kode di atas menerapkan *Dynamic Data Masking* untuk melindungi data sensitif pada dimensi auditor dan rekomendasi. Kolom ID_Sistem_Sumber menggunakan partial masking untuk mempertahankan pola data tanpa mengungkap identitas penuh, sedangkan kolom Penanggung_Jawab menggunakan default masking untuk menyembunyikan seluruh nilai. Hak akses UNMASK hanya diberikan kepada pengguna dengan wewenang tinggi seperti Kepala SPI dan Analis Audit, sehingga hanya mereka yang dapat melihat data asli. Implementasi ini meningkatkan keamanan data dan mematuhi praktik standar perlindungan informasi.

Tujuannya adalah mencegah:

- kebocoran data sensitif,
- akses tidak sah,
- penyalahgunaan data pribadi.

4 .Implement Audit Trail

```
-- 1. Create Audit Table
CREATE TABLE dbo.AuditLog (
    AuditID BIGINT IDENTITY (1 ,1) PRIMARY KEY ,
    EventTime DATETIME2 DEFAULT SYSDATETIME (),
    UserName NVARCHAR (128) DEFAULT SUSER_SNAME (),
    EventType NVARCHAR (50) ,
    SchemaName NVARCHAR (128) ,
    ObjectName NVARCHAR (128) ,
    SQLStatement NVARCHAR (MAX),
    RowsAffected INT ,
    IPAddress VARCHAR (50) ,
    ApplicationName NVARCHAR (128) DEFAULT APP_NAME ()
);
GO
```

```
-- 2. Create Audit Trigger (Merekam perubahan pada Dimensi Rekomendasi)
CREATE TRIGGER trg_Audit_Dim_Rekomendasi
ON dbo.Dim_Rekomendasi
```

```

AFTER INSERT, UPDATE, DELETE
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @EventType NVARCHAR (50);
    DECLARE @RowsAffected INT;

    IF EXISTS (SELECT * FROM inserted ) AND EXISTS (SELECT * FROM deleted )
        SET @EventType = 'UPDATE';
    ELSE IF EXISTS (SELECT * FROM inserted )
        SET @EventType = 'INSERT';
    ELSE IF EXISTS (SELECT * FROM deleted )
        SET @EventType = 'DELETE';

    SET @RowsAffected = @@ROWCOUNT;

    INSERT INTO dbo.AuditLog (EventType , SchemaName , ObjectName , RowsAffected)
    VALUES (@EventType , 'dbo', 'Dim_Rekomendasi', @RowsAffected);

END;
GO

-- 3. Enable SQL Server Audit (Server-level)
CREATE SERVER AUDIT DataWarehouse_Audit
TO FILE
( FILEPATH = N'D:\Audit\' ,
  MAXSIZE = 100 MB ,
  MAX_ROLLOVER_FILES = 10
)
WITH ( ON_FAILURE = CONTINUE );
GO

ALTER SERVER AUDIT DataWarehouse_Audit WITH (STATE = ON);
GO

-- 4. Create Database Audit Specification (Melacak aksi DML dan SELECT)
CREATE DATABASE AUDIT SPECIFICATION DataWarehouse_DB_Audit
FOR SERVER AUDIT DataWarehouse_Audit
ADD (SELECT , INSERT , UPDATE , DELETE ON SCHEMA :: dbo BY public);
GO

ALTER DATABASE AUDIT SPECIFICATION DataWarehouse_DB_Audit WITH (STATE = ON);
GO

```

Kode ini membangun sistem audit terintegrasi pada SQL Server. Dimulai dari pembuatan tabel AuditLog untuk mencatat aktivitas perubahan data, sebuah trigger audit pada tabel Dim_Rekomendasi untuk merekam operasi DML, dan konfigurasi SQL Server Audit untuk memonitor aktivitas SELECT, INSERT, UPDATE, dan DELETE pada seluruh schema dbo. Kombinasi ini memastikan bahwa seluruh perubahan

dan akses terhadap data dalam Data Warehouse dapat dilacak dengan jelas untuk kepentingan keamanan, kontrol, dan pemantauan, serta aktivitas pengguna pada level server dan level database. Sistem ini penting untuk: keamanan data (*data security*), pelacakan perubahan (*change tracking*), pemantauan aktivitas user, forensik jika terjadi kesalahan atau manipulasi data.

3.4 Backup and Recovery Strategy

```
-- 1. Full Backup (Biasanya Mingguan)
BACKUP DATABASE [DM_SPI_DW]
TO DISK = N'D:\Backup\DM_SPI_DW_Full.bak'
WITH
COMPRESSION,
INIT,
NAME = N'Full Database Backup - SPI DW',
STATS = 10;
GO

-- 2. Differential Backup (Biasanya Harian)
BACKUP DATABASE [DM_SPI_DW]
TO DISK = N'D:\Backup\DM_SPI_DW_Diff.bak'
WITH
DIFFERENTIAL,
COMPRESSION,
INIT,
NAME = N'Differential Database Backup - SPI DW',
STATS = 10;
GO

-- 3. Transaction Log Backup
BACKUP LOG [DM_SPI_DW]
TO DISK = N'D:\Backup\DM_SPI_DW_Log.trn'
WITH
COMPRESSION,
INIT,
NAME = N'Transaction Log Backup - SPI DW',
STATS = 10;
GO
```

Kode yang diberikan berisi tiga jenis backup untuk database DM_SPI_DW. Setiap jenis backup memiliki tujuan berbeda, sehingga membentuk strategi backup yang lengkap dan aman untuk mencegah kehilangan data.

- Full Backup – Backup Utama (Biasanya Mingguan): Full backup adalah proses membuat salinan seluruh isi database, termasuk Semua tabel, Data, Index, Stored procedure, Struktur database Tujuannya untuk menjadi dasar untuk differential dan log backup, dan menjaga agar seluruh isi database dapat dipulihkan jika terjadi kegagalan. Backup ini dilakukan Mingguan (misalnya hari Minggu malam) Karena ukuran file besar dan prosesnya lebih lama

- Differential Backup – Backup Perubahan (Biasanya Harian): Differential backup hanya mencadangkan data yang berubah sejak full backup terakhir , jadi lebih cepat dan lebih kecil ukurannya. Tujuan backup ini untuk Mengurangi waktu backup harian, dan mempercepat pemulihan (restore) dibanding log-only restore
- Transaction Log Backup – Backup Riwayat Transaksi: Transaction Log Backup membackup semua transaksi yang terjadi setelah backup log sebelumnya, termasuk INSERT, UPDATE, DELETE, dan operasi transaksi lainnya. Tujuan Back up ini adalah untuk mendukung point-in-time recovery (restore sampai detik tertentu), menjaga agar log file tidak membengkak, dan merupakan bagian dari strategi FULL Recovery Model Biasanya dijalankan Setiap 15 menit, 30 menit, atau 1 jam atau Tergantung tingkat critical data

3.5. User Acceptance Testing

Test ID	Skenario Pengujian (Scenario)	Ekspektasi Hasil (Expected Result)	Status	Catatan (Notes)
TC00 1	[Dashboard 1] Validasi akurasi total "Potensi Kerugian" pada KPI Card	Angka sesuai dengan total di database (Rp 25T / 6,42T sesuai filter aktif)	Pass	Data akurat
TC00 2	[Dashboard 1] Filter visual berdasarkan "Tahun Audit" (misal: 2024)	Seluruh grafik (Total Risiko, Tren Bulanan) otomatis terupdate menampilkan data tahun 2024 saja	Pass	Interaktivitas berjalan
TC00 3	[Dashboard 2] Drill-down pada grafik "Top 10 Unit Kerja"	Mengklik "Unit Kerja 58" akan memfilter tabel detail dan grafik lain khusus untuk unit tersebut	Pass	Cross-filtering aktif
TC00 4	[Dashboard 2] Validasi perhitungan "Waterfall Chart" Kategori Risiko	Total akumulasi dari (Kepatuhan + Operasional + Finansial) sama dengan Total Risiko Global	Pass	Logika visual benar

TC00 5	[Dashboard 3] Identifikasi Auditor dengan kinerja tertinggi	Grafik "Top 10 Auditor" menampilkan "Auditor 194" di urutan pertama (sesuai data source)	Pass	Sorting benar
TC00 6	[Performance] Dashboard Load Time	Dashboard "Analytical Deep-Dive" terbuka sempurna dalam waktu < 5 detik	Pass	Performa optimal