

FINAL PRESENTATION

# **IMPLEMENTASI PRODUKSI DATA MART AKADEMIK DAN KEMAHASISWAAN**

Data Warehouse

# KELOMPOK 4

1. Adil Aulia Rahma Nurhidayah (122450058)
2. Rosalia Siregar (123450036)
3. Muhammad Hanif Dzaky Arifin (123450064)
4. Haikall Fransisko Simbolon (123450106)

# **Business Domain Overview**

## **Domain: Kemahasiswaan ITERA**

### **Ruang Lingkup Data:**

- Data mahasiswa dan organisasi kemahasiswaan
- Kegiatan dan partisipasi mahasiswa
- Prestasi mahasiswa (regional, nasional, internasional)
- Program beasiswa dan bantuan dana
- Pengajuan dan realisasi dana kegiatan

### **Project Objectives**

### **Tujuan Proyek:**

1. Mengintegrasikan data kemahasiswaan dari berbagai sumber
2. Menyediakan akses data cepat, akurat, dan konsisten
3. Mendukung data-driven decision making
4. Monitoring keaktifan mahasiswa melalui dashboard
5. Pelacakan tren partisipasi dan prestasi dari waktu ke waktu

# Business Requirements Summary

## Stakeholders & Decision Makers

- Wakil Rektor Bidang Kemahasiswaan
- Kepala Bagian Kemahasiswaan
- Ketua Organisasi Mahasiswa (Ormawa/UKM)
- (Ormawa/UKM)
- Staff Kemahasiswaan

## 6 Proses Bisnis Utama:

1. Pengelolaan Data Mahasiswa
2. Pendataan Kegiatan Mahasiswa (UKM/Ormawa)
3. Pengajuan dan Realisasi Dana Kegiatan
4. Pendataan Penerima Beasiswa
5. Pencatatan Prestasi Mahasiswa
6. Layanan Konseling dan Bimbingan

# BUSINESS REQUIREMENTS SUMMARY

## KEY PERFORMANCE INDICATORS (KPI):

1. Jumlah kegiatan kemahasiswaan per semester
2. Jumlah mahasiswa aktif dalam kegiatan
3. Jumlah penerima beasiswa per periode
4. Jumlah prestasi mahasiswa (regional/nasional/internasional)
5. Tingkat efisiensi penggunaan dana kegiatan

# Data Sources

Sumber	Tipe	Volume	Update Frequency
Siakad	SQL Server	50 rows	Real-time
Sistem Beasiswa	Oracle DB	20 rows	Per semester
Sistem Keuangan	MySQL	50 rows	Harian
Laporan Kegiatan	Excel/CSV	100 rows	bulanan

# DIMENSIONAL MODEL OVERVIEW

## Arsitektur: Kimball (Star Schema)

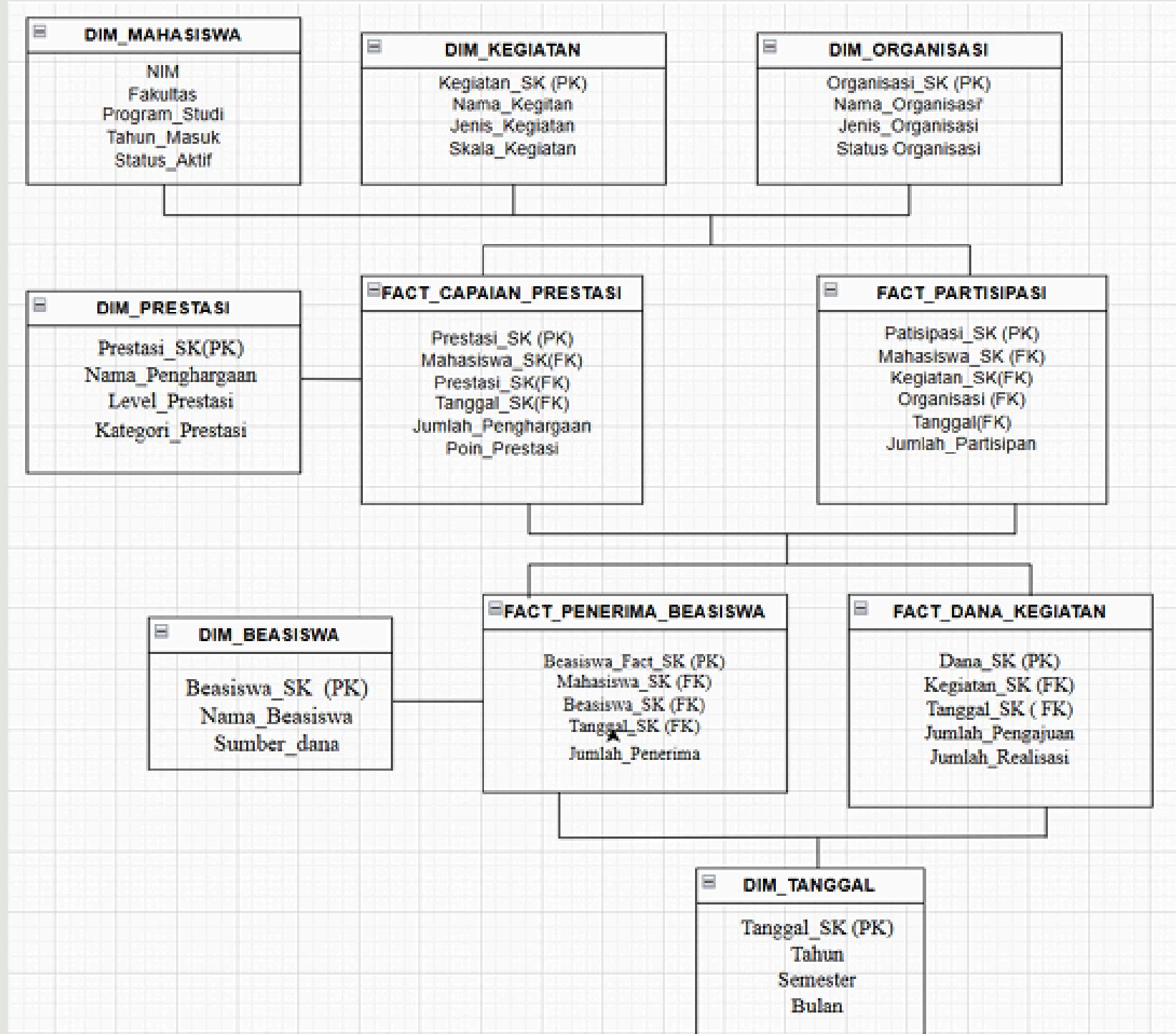
### Fact Tables (4):

1. Fact\_Partisipasi\_Kegiatan
2. Fact\_Dana\_Kegiatan
3. Fact\_Capaian\_Prestasi
4. Fact\_Penerima\_Basiswa

### Dimension Tables (6):

1. Dim\_Mahasiswa (SCD Type 2)
2. Dim\_Kegiatan
3. Dim\_Organisasi (SCD Type 2)
4. Dim\_Prestasi
5. Dim\_Basiswa
6. Dim\_Tanggal

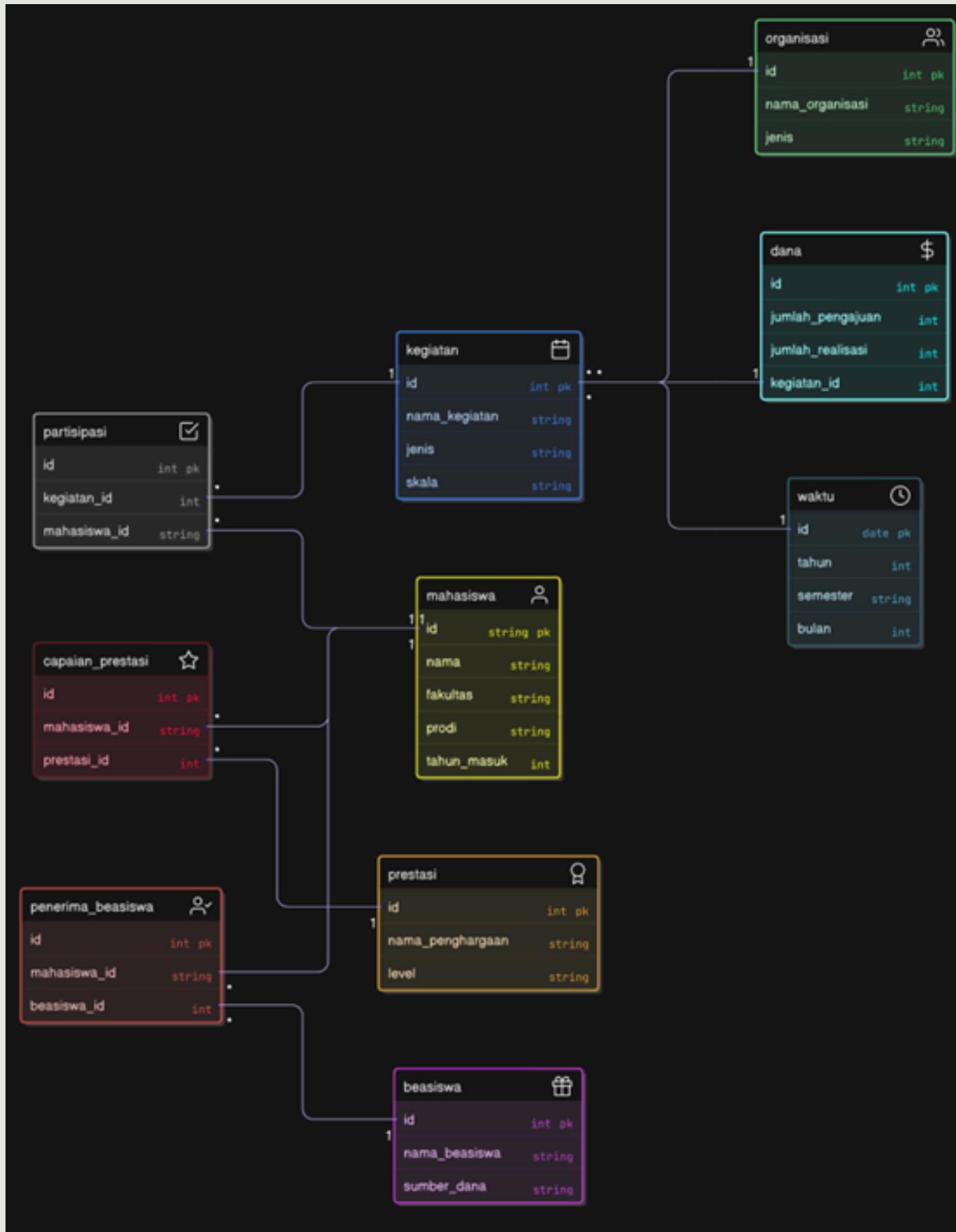
# STAR SCHEMA DIAGRAM



- Kimball Approach dipilih untuk kemudahan query dan performa optimal
- Denormalized structure untuk analisis cepat
- Surrogate Keys untuk integritas data dan mendukung SCD
- Grain Level: 1 mahasiswa per 1 kegiatan per 1 tanggal

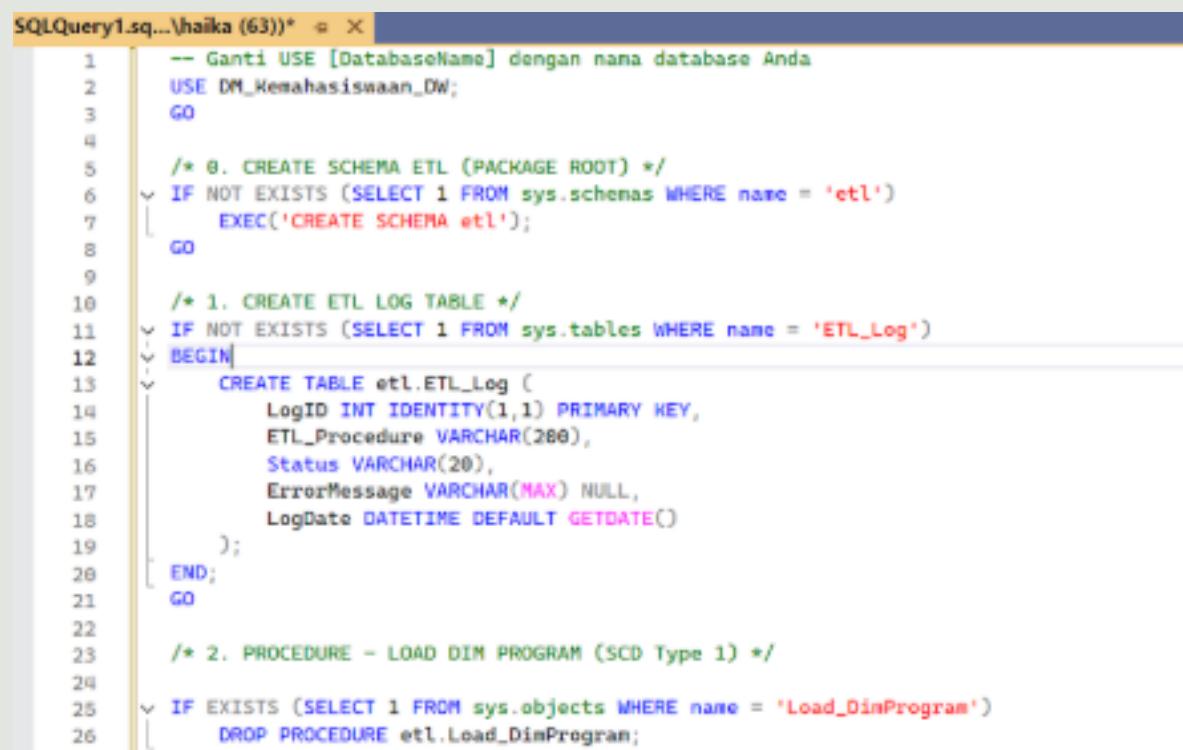
# DATABASE IMPLEMENTATION

## ERD



# DATABASE IMPLEMENTATION

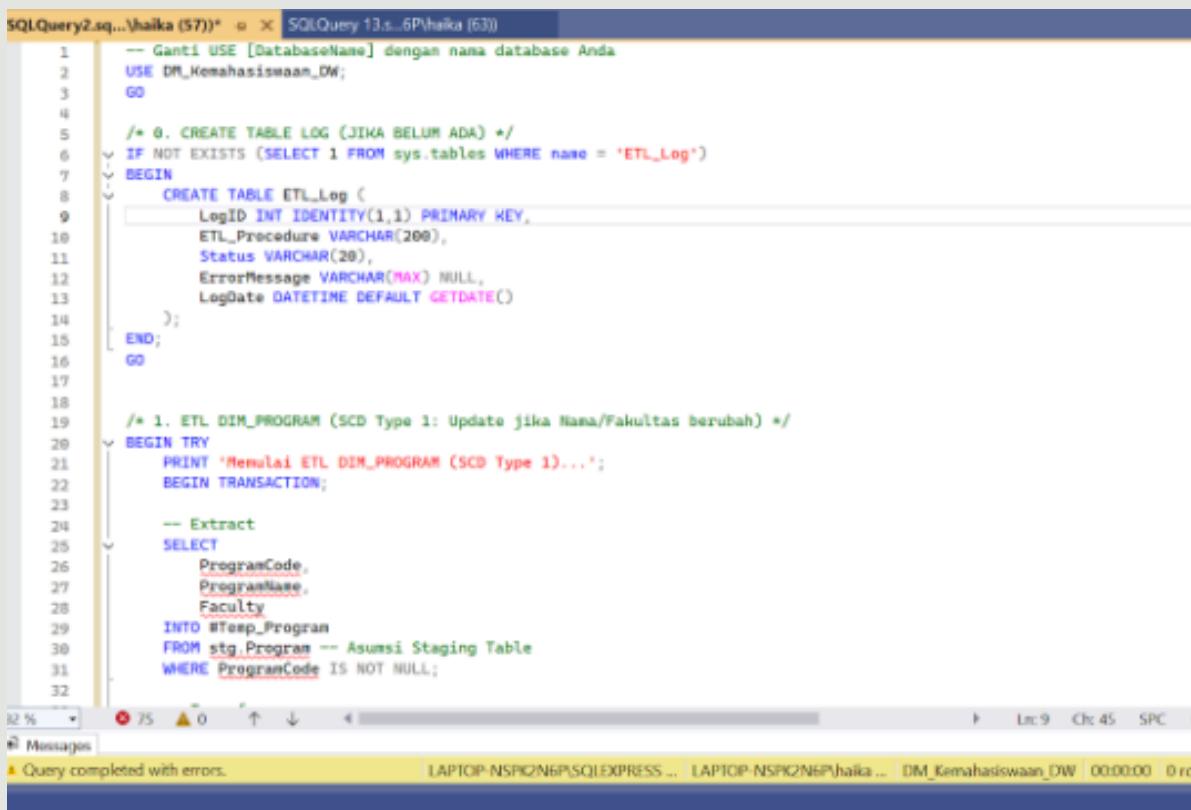
## ETL PACKAGE



```
SQLQuery1.sql...\\haika (63)*  □ X
1   -- Ganti USE [DatabaseName] dengan nama database Anda
2   USE DM_Kemahasiswaan_DW;
3   GO
4
5   /* 0. CREATE SCHEMA ETL (PACKAGE ROOT) */
6   IF NOT EXISTS (SELECT 1 FROM sys.schemas WHERE name = 'etl')
7   EXEC('CREATE SCHEMA etl');
8   GO
9
10  /* 1. CREATE ETL LOG TABLE */
11  IF NOT EXISTS (SELECT 1 FROM sys.tables WHERE name = 'ETL_Log')
12  BEGIN
13      CREATE TABLE etl.ETL_Log (
14          LogID INT IDENTITY(1,1) PRIMARY KEY,
15          ETL_Procedure VARCHAR(200),
16          Status VARCHAR(20),
17          ErrorMessage VARCHAR(MAX) NULL,
18          LogDate DATETIME DEFAULT GETDATE()
19      );
20  END;
21  GO
22
23  /* 2. PROCEDURE – LOAD DIM PROGRAM (SCD Type 1) */
24
25  IF EXISTS (SELECT 1 FROM sys.objects WHERE name = 'Load_DimProgram')
26      DROP PROCEDURE etl.Load_DimProgram;
```

ETL Package ini menunjukkan proses pembuatan package ETL yang bertugas memindahkan data dari sumber ke staging. Script membuat log table untuk mencatat proses ETL, kemudian package disiapkan untuk melakukan load data harian ke staging table. Package juga mencatat waktu mulai, selesai, jumlah baris yang diproses, serta status proses. Tujuannya adalah memastikan proses ETL berjalan terkontrol dan dapat diaudit.

## ETL SCRIPT



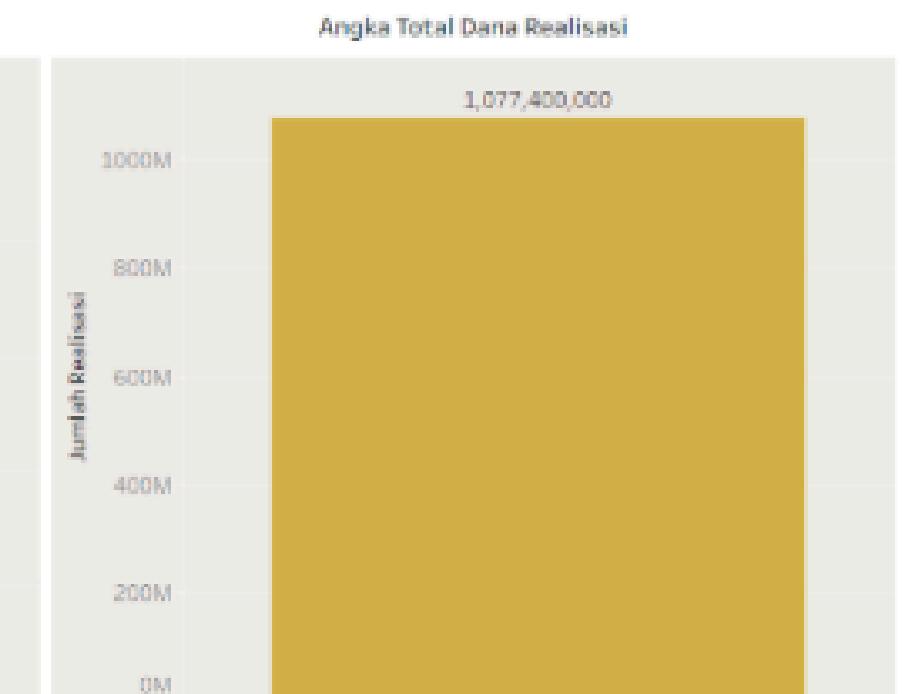
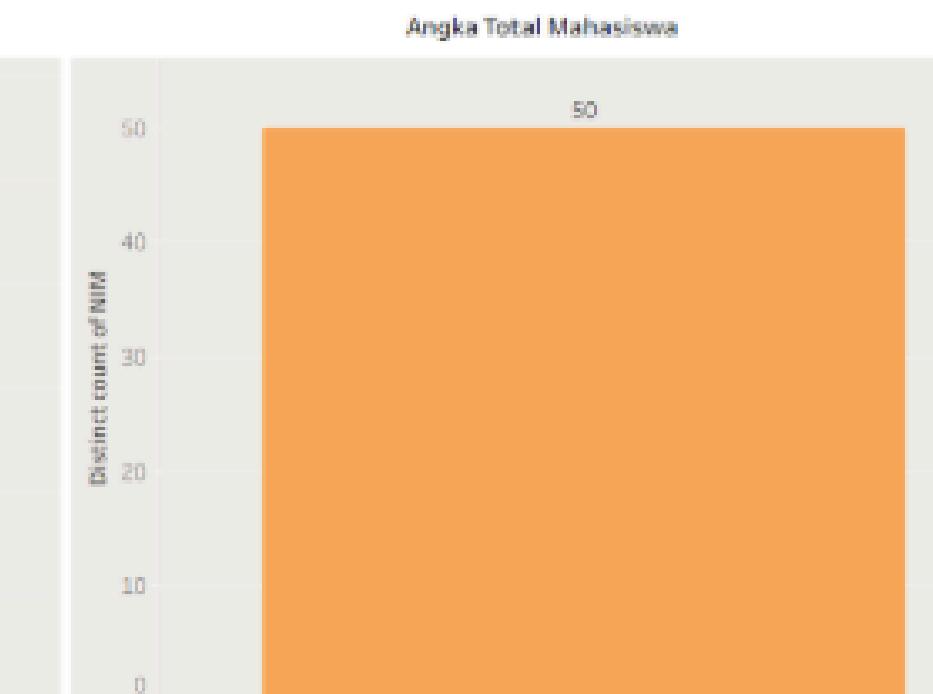
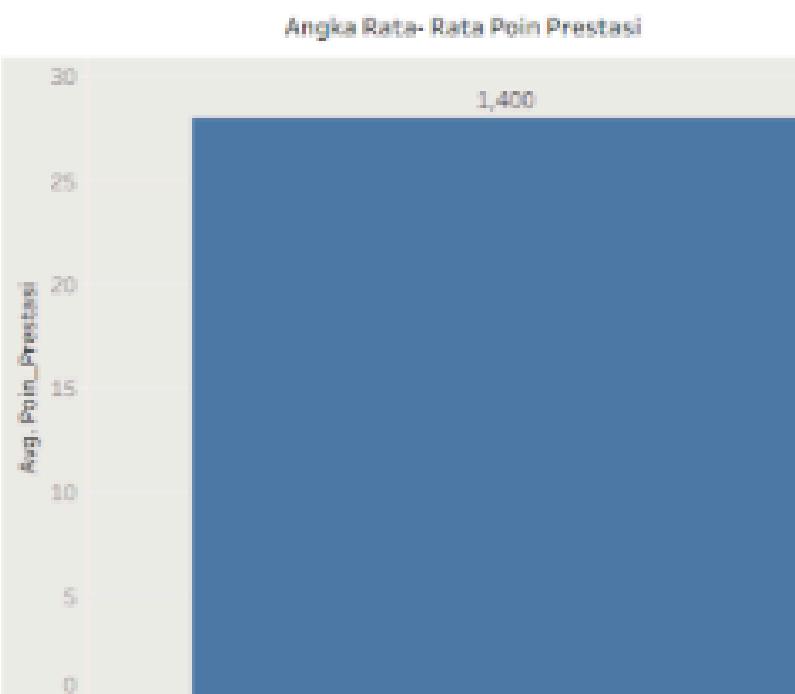
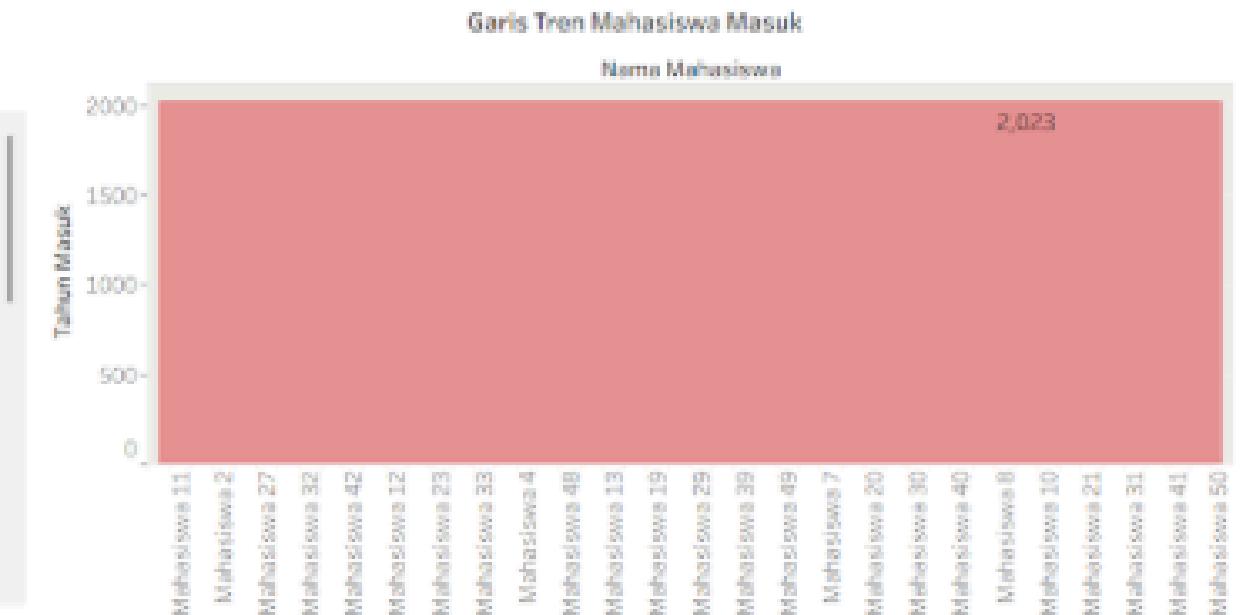
```
SQLQuery2.sql...\\haika (57)*  □ X SQLQuery 13.s...\\haika (63)
1   -- Ganti USE [DatabaseName] dengan nama database Anda
2   USE DM_Kemahasiswaan_DW;
3   GO
4
5   /* 0. CREATE TABLE LOG (JIKA BELUM ADA) */
6   IF NOT EXISTS (SELECT 1 FROM sys.tables WHERE name = 'ETL_Log')
7   BEGIN
8       CREATE TABLE ETL_Log (
9           LogID INT IDENTITY(1,1) PRIMARY KEY,
10          ETL_Procedure VARCHAR(200),
11          Status VARCHAR(20),
12          ErrorMessage VARCHAR(MAX) NULL,
13          LogDate DATETIME DEFAULT GETDATE()
14      );
15  END;
16  GO
17
18  /* 1. ETL DIM_PROGRAM (SCD Type 1: Update jika Nama/Fakultas berubah) */
19  BEGIN TRY
20      PRINT 'Mulai ETL DIM_PROGRAM (SCD Type 1)...';
21      BEGIN TRANSACTION;
22
23      -- Extract
24      SELECT
25          ProgramCode,
26          ProgramName,
27          Faculty
28      INTO #Temp_Program
29      FROM stg_Program -- Asumsi Staging Table
30      WHERE ProgramCode IS NOT NULL;
31
32  END TRY
33  BEGIN CATCH
34      ROLLBACK TRANSACTION;
35  END CATCH;
```

ETL Script adalah prosedur SQL yang benar-benar mengeksekusi proses ETL. Script melakukan langkah-langkah:

1. Membersihkan staging terlebih dahulu (TRUNCATE).
2. Melakukan INSERT data baru dari sumber.
3. Memproses transformasi jika diperlukan.
4. Memindahkan hasil akhir ke tabel fact/dim.
5. Script ini merupakan implementasi teknis dari alur ETL Package agar data warehouse selalu terbarui dengan data terbaru.

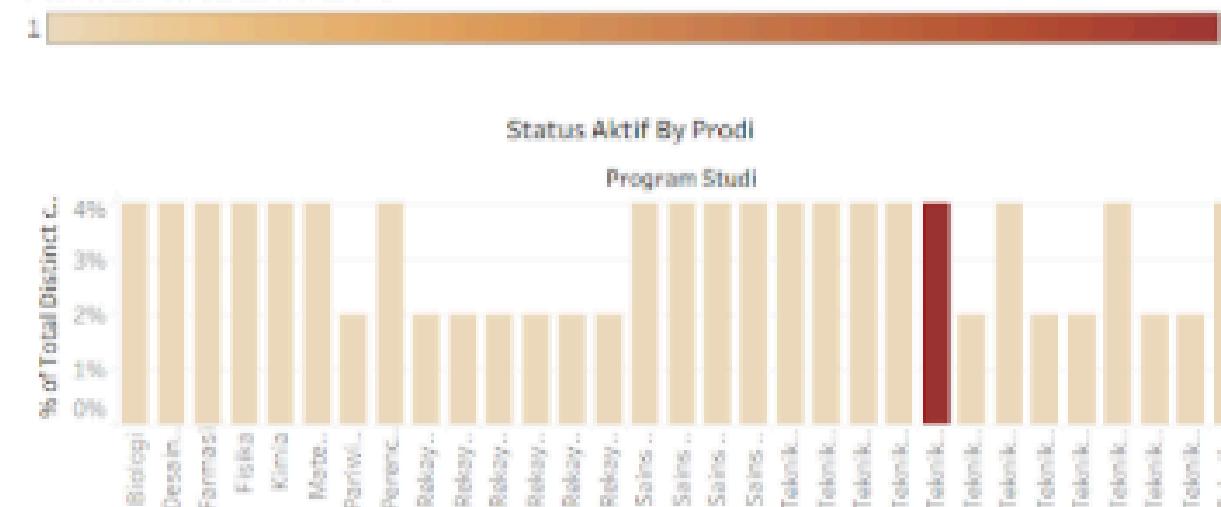
# DASHBOARD

## **DASBOARD 1 : Executive Summary**

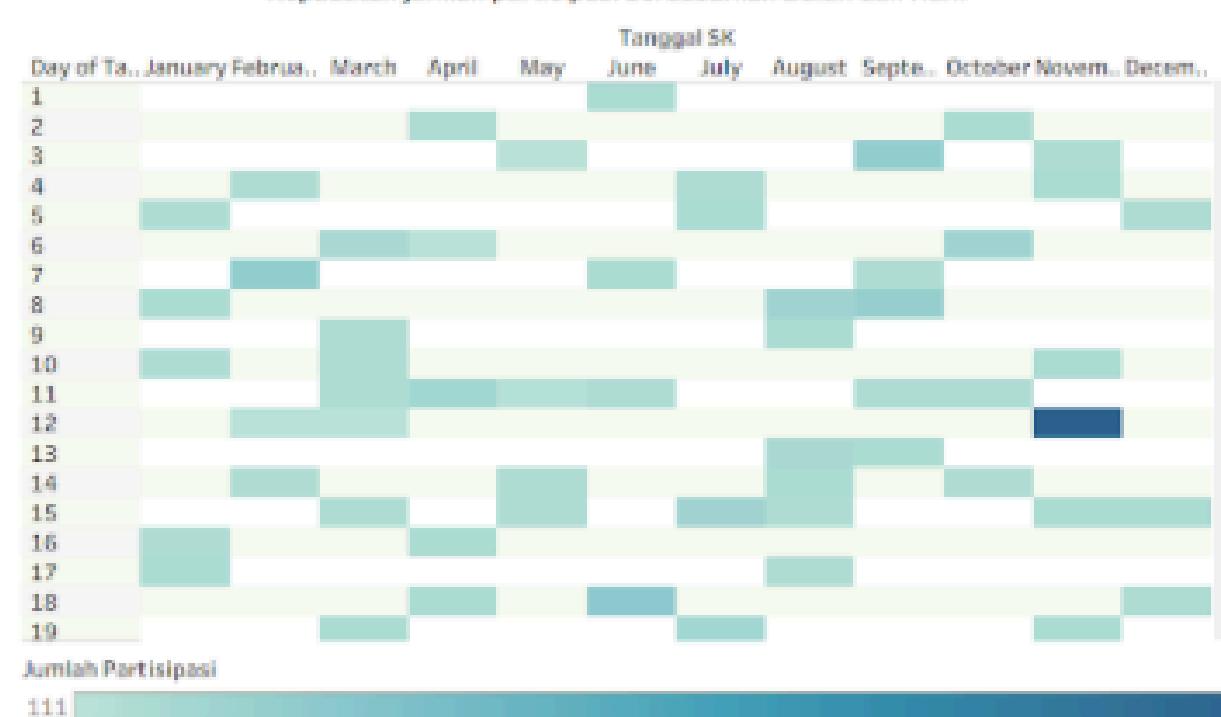


DASHBOARD 2: Academic Performance

## Distinct count of Status Maharashtra



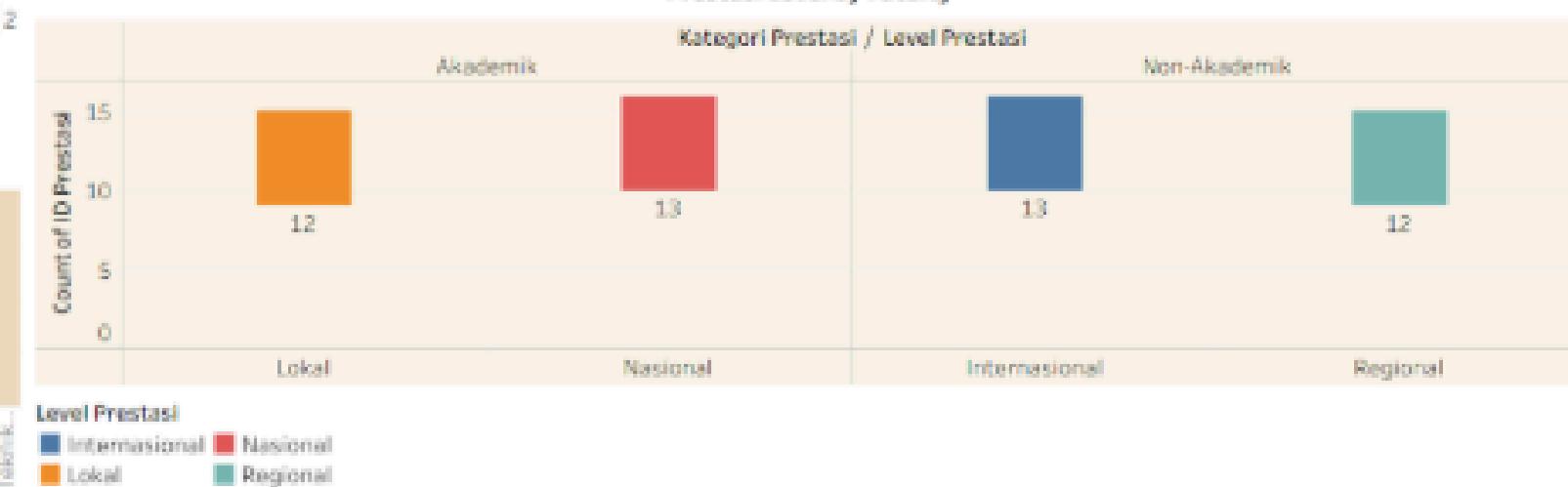
#### Kepadatan jumlah partisipasi berdasarkan Bulan dan Hari



#### Prestasi Level by Faculty

Kategori Prestasi / Level Prestasi

Non-Algorithmic

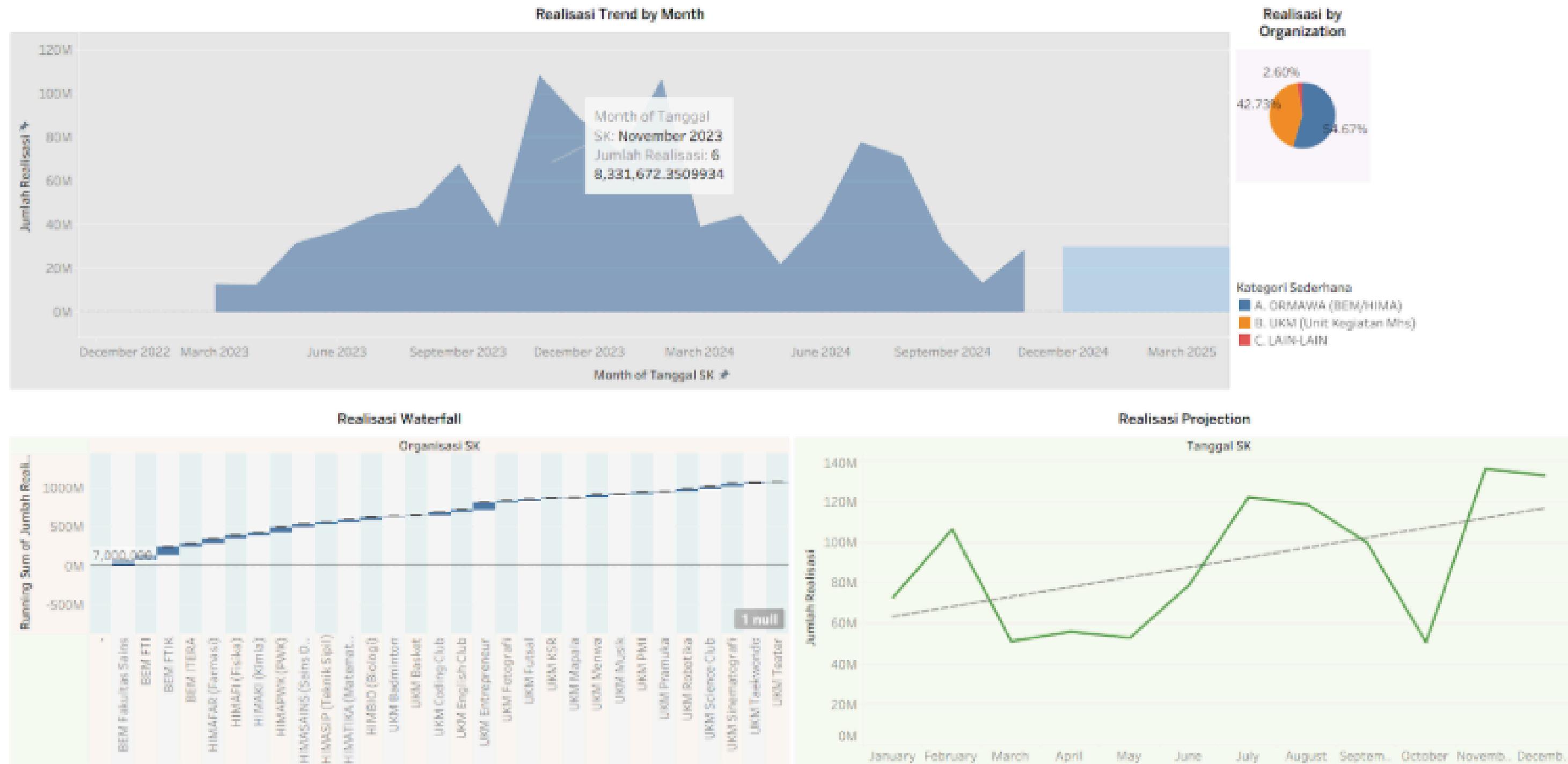


#### Top 10 Organisasi dengan Volum Kegiatan tertinggi

Organisational



## DASHBOARD 3 : Financial Analysis



# Data Volume & Quality

## Data Produksi:

- 50+ data mahasiswa
- 30+ data organisasi (UKM dan Ormawa)
- 50+ data kegiatan per tahun
- 100+ data partisipasi
- 50+ data prestasi
- 20+ program beasiswa

## Langkah Pengendalian Kualitas Data:

- Pemeriksaan Kelengkapan: validasi nilai NULL
- Pemeriksaan Konsistensi: memastikan integritas referensial antar-tabel
- Deteksi Duplikasi: memastikan keunikan Partisipasi\_ID
- Rekonsiliasi Data: membandingkan data sumber dengan data di data warehouse
- Target Skor Kualitas: >95%

# Optimization Techniques

## 1. Strategi Indexing:

- Clustered Index: diterapkan pada kolom Tanggal\_SK di tabel fakta
- Non-Clustered Index: diterapkan pada Foreign Key untuk mengoptimalkan operasi JOIN
- Columnstore Index: digunakan untuk mempercepat agregasi pada query analitis

## 2. Partitioning:

- Fungsi Partisi: berdasarkan Tahun Akademik
- Rentang Partisi: 2023/2024, 2024/2025, 2025/2026
- Manfaat: memungkinkan partition elimination sehingga query berbasis waktu menjadi lebih cepat

## 3. Pemeliharaan Statistik:

- Auto-update statistics diaktifkan
- UPDATE STATISTICS manual dilakukan setelah proses ETL besar

# Security Implementation

## Role-Based Access Control (RBAC)

1. Membuat role khusus: db\_executive, db\_operational, db\_viewer, db\_etl\_operator.
2. Setiap role diberi hak akses sesuai tugasnya (read-only, ETL, staging R/W, unmask).
3. Menerapkan principle of least privilege agar akses tidak berlebihan.

## Audit Trail

1. Membuat tabel AuditLog untuk mencatat semua perubahan data.
2. Trigger audit pada fact table mencatat who, what, when, rows affected.
3. Memastikan monitoring, jejak perubahan, dan compliance data.

## Challenges & Solutions

1. Terkendala dalam pemrograman menggunakan SSMS
2. Kurang pemahaman dalam melakukan pelaksanaan misi

# Monitoring & Maintenance

## Dashboard Kualitas Data:

- Pemantauan Quality Score (ambang batas: 95%)
- Sistem peringatan untuk masalah kualitas data
- Prosedur usp\_Generate\_DQ\_Report untuk pelaporan otomatis

## Monitoring ETL:

- Riwayat SQL Server Agent Job
- Pelacakan LoadDate pada tabel fakta
- Identifikasi SourceSystem dari setiap data yang masuk

## Monitoring Performa:

- Pelacakan waktu eksekusi query
- Pemeriksaan fragmentasi index
- Pemantauan pertumbuhan database

# Thank You

For your attention