

**LAPORAN TUGAS MISI KETIGA  
PERGUDANGAN DATA RC**



**Disusun oleh:**

**Kelompok 19**

|                    |           |
|--------------------|-----------|
| Jeremia Susanto    | 122450022 |
| Feryadi Yulius     | 122450087 |
| Dede Masita        | 121450007 |
| Nisrina Nur Afifah | 122450052 |
| Dwi sulistiani     | 121450079 |

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

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Industri penerbangan merupakan hal penting dalam perekonomian global karena menghubungkan manusia dan barang antarnegara. Sektor ini memerlukan investasi yang besar, tunduk pada regulasi ketat, serta sangat dipengaruhi oleh dinamika ekonomi dan politik. Maskapai penerbangan dihadapkan pada tantangan seperti fluktuasi harga bahan bakar, persaingan tarif rendah, serta peraturan keselamatan dan emisi yang semakin ketat. Pandemi COVID-19 menunjukkan kerentanannya terhadap gangguan besar, sehingga mendorong pemanfaatan teknologi seperti sistem pemesanan cerdas dan pemeliharaan berbasis data untuk meningkatkan efisiensi dan kepuasan pelanggan.

### **1.2 Rumusan Masalah**

Berdasarkan latar belakang tersebut, maka permasalahan yang diangkat dalam penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana merancang dan membangun gudang data untuk industri penerbangan menggunakan pendekatan skema bintang (star schema)?
2. Bagaimana struktur data dalam tabel fakta dan tabel dimensi dapat mendukung proses analisis tren penerbangan.
3. Bagaimana penerapan gudang data ini dapat membantu pengguna dalam menyajikan informasi yang relevan dan mendukung pengambilan keputusan strategis dalam industri penerbangan?

### **1.3 Metode Pengembangan Gudang Data**

Dalam membangun gudang data, pemilihan metode pengembangan sangat menentukan keberhasilan implementasi sistem. Terdapat dua pendekatan umum yang dapat digunakan, yaitu pendekatan waterfall dan pendekatan iterative. Pendekatan waterfall merupakan model pengembangan sistem yang bersifat sekuensial, di mana setiap tahap mulai dari analisis kebutuhan, perancangan, implementasi, hingga pengujian dilaksanakan secara berurutan. Pendekatan ini cocok digunakan ketika kebutuhan sistem sudah didefinisikan secara lengkap dan jelas sejak awal pengembangan.

Sementara itu, pendekatan iterative lebih menekankan pada pengembangan sistem secara bertahap dan berulang. Dalam pendekatan ini, sistem dikembangkan dalam bentuk prototipe-prototipe kecil yang dievaluasi dan diperbaiki secara terus-menerus berdasarkan masukan dari pengguna. Pendekatan iterative sangat sesuai dalam proyek gudang data yang bersifat dinamis dan kompleks, karena memberikan fleksibilitas dalam menyesuaikan sistem terhadap perubahan kebutuhan. Pemilihan metode pengembangan yang tepat akan sangat mempengaruhi efektivitas gudang data yang dibangun, baik dari segi ketepatan analisis maupun kecepatan dalam mendukung kebutuhan pengguna akhir.

## BAB 2

### DESAIN KONSEPTUAL

#### 2.1 Identifikasi Proses Bisnis

Dalam industri penerbangan, pergudangan data memegang peranan penting dalam mendukung berbagai proses bisnis yang strategis. Data warehouse berperan penting dalam mendukung keputusan strategis maskapai. Sehingga maskapai dapat merancang strategi bisnis dengan lebih akurat dan efisien. Seperti beberapa pertanyaan berikut :

1. Berapa rata-rata keterlambatan untuk rute Jakarta–Surabaya per bulan pada tahun 2024?
2. Bandara mana yang memiliki total biaya operasional tertinggi dalam 3 bulan terakhir?
3. Bagaimana tren jumlah penumpang per jenis kelas layanan setiap kuartal?
4. Jenis pesawat apa yang paling sering mengalami delay lebih dari 30 menit?
5. Apakah pesawat yang lebih tua cenderung menyebabkan lebih banyak delay?

#### 2.2 Data

Dataset ini berisi 1.000 entri data penerbangan yang mencakup informasi seperti jumlah penumpang, pendapatan, biaya operasional, serta waktu keberangkatan dan kedatangan. Setiap baris mewakili satu catatan fakta penerbangan dengan identifikasi seperti FaktaID, WaktuID, dan berbagai ID relasional lainnya (LokasiID, PesawatID, dll). Kolom WaktuID merekam tanggal penerbangan dalam format YYYYMMDD, dengan rentang data dari Januari 2024 hingga Mei 2025. Data ini juga mencatat keterlambatan dalam menit, yang memungkinkan analisis performa ketepatan waktu dan efisiensi operasional maskapai.

#### 2.3 Tabel Dimensi dan Fakta

##### 2.3.1 Tabel Fakta

Tabel fakta pada sistem ini adalah Fakta\_Penerbangan, yang merekam informasi kuantitatif terkait aktivitas penerbangan. Tabel ini memuat ukuran numerik seperti jumlah penumpang, pendapatan, dan jumlah penerbangan yang menjadi inti dari analisis operasional dan bisnis maskapai. Selain itu, tabel ini memiliki enam kunci asing yang mengarah ke masing-masing tabel dimensi, yaitu waktu, lokasi, penerbangan, pesawat, kelas layanan, dan status penerbangan. Tabel fakta ini menjadi pusat dari skema bintang yang dirancang, karena seluruh analisis berbasis dimensi akan diturunkan dari data yang tersimpan di dalamnya.

Tabel 2.1 Tabel Fakta

| Nama Tabel      | Primary Key    | Atribut Deskriptif                                      | Tipe Data          |
|-----------------|----------------|---|--------------------|
| Dim_Waktu       | ID_Waktu       | Tanggal, Hari, Bulan, Tahun, Kuartal                    | DATE, VARCHAR, INT |
| Dim_Lokasi      | ID_Lokasi      | Nama_Bandara, Kota, Negara                              | VARCHAR            |
| Dim_Penerbangan | ID_Penerbangan | Kode_Penerbangan, Maskapai, Waktu_Berangkat, Waktu_Tiba | VARCHAR, TIME      |

|                       |                      |  |                  |
|-----------------------|----------------------|--|------------------|
| Dim_Pesawat           | ID_Pesawat           | Nomor_Pesawat,<br>Tipe_Pesawat,<br>Kapasitas | VARCHAR,<br>INT  |
| Dim_KelasLayanan      | ID_KelasLayanan      | Nama_Kelas,<br>Deskripsi                     | VARCHAR,<br>TEXT |
| Dim_StatusPenerbangan | ID_StatusPenerbangan | Status, Deskripsi                            | VARCHAR,<br>TEXT |

### 2.3.2 Tabel Dimensi

Tabel dimensi berfungsi untuk memberikan konteks deskriptif terhadap data numerik dalam tabel fakta. Setiap tabel dimensi berisi atribut yang menjelaskan detail dari entitas bisnis tertentu. Misalnya, tabel Dim\_Waktu memungkinkan analisis berdasarkan kalender seperti hari, bulan, dan kuartal; Dim\_Lokasi mendeskripsikan bandara dan kota; Dim\_Penerbangan menjelaskan kode, maskapai, serta jadwal keberangkatan dan kedatangan; Dim\_Pesawat berisi tipe dan kapasitas pesawat; Dim\_KelasLayanan menunjukkan jenis layanan seperti ekonomi atau bisnis; dan Dim\_StatusPenerbangan mendeskripsikan status seperti tepat waktu atau tertunda. Informasi dari tabel-tabel ini sangat penting untuk melakukan analisis multidimensi terhadap performa penerbangan.

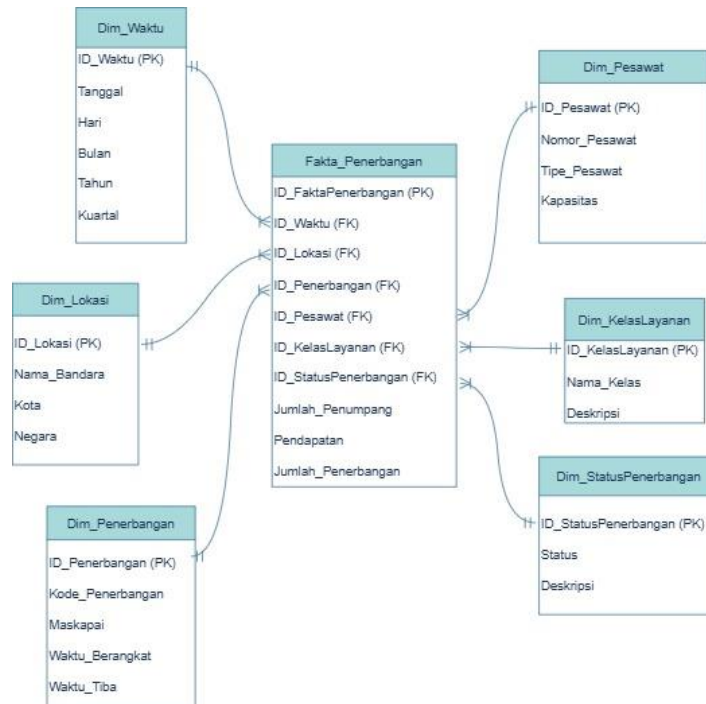
**Tabel 2.2 Tabel Dimensi**

| Atribut              | Tipe Data | Keterangan  |
|----------------------|-----------|---|
| ID_FaktaPenerbangan  | INT       | Primary key   |
| ID_Waktu             | INT       | Kunci asing yang merujuk ke `Dim_Waktu`             |
| ID_Lokasi            | INT       | Kunci asing yang merujuk ke `Dim_Lokasi`            |
| ID_Penerbangan       | INT       | Kunci asing yang merujuk ke `Dim_Penerbangan`       |
| ID_Pesawat           | INT       | Kunci asing yang merujuk ke `Dim_Pesawat`           |
| ID_KelasLayanan      | INT       | Kunci asing yang merujuk ke `Dim_KelasLayanan`      |
| ID_StatusPenerbangan | INT       | Kunci asing yang merujuk ke `Dim_StatusPenerbangan` |
| Jumlah_Penumpang     | INT       | Jumlah penumpang dalam penerbangan                  |
| Pendapatan           | FLOAT     | Total pendapatan dari penerbangan                   |
| Jumlah_Penerbangan   | INT       | Jumlah penerbangan yang terjadi                     |

## 2.1 Skema Data

Dalam pembangunan data warehouse ini menggunakan star schema yang digunakan untuk menyimpan dan menganalisis data penerbangan. Skema ini terdiri dari satu tabel fakta utama dan beberapa tabel dimensi yang saling terhubung. Star schema biasanya digunakan dalam sistem data warehouse karena bentuknya yang simpel dan mudah digunakan untuk keperluan analisis.

Berdasarkan gambar tersebut menunjukkan bahwa tabel utamanya bernama Fakta\_Penerbangan, yang berisi data inti seperti jumlah penumpang, pendapatan, dan jumlah penerbangan. Tabel ini juga memiliki beberapa kolom yang terhubung ke tabel-tabel lain, yaitu waktu, lokasi, penerbangan, pesawat, kelas layanan, dan status penerbangan. Dengan skema ini, pengguna dapat melakukan analisis seperti melihat jumlah penumpang berdasarkan waktu, mengevaluasi kinerja maskapai, atau membandingkan pendapatan antar kelas layanan. Struktur seperti ini memudahkan proses pelaporan dan pengambilan keputusan.



Gambar.1



## BAB 3

### DESAIN LOGIKAL DAN FISIK GUDANG DATA

#### 3.1 Translasi Desain Konseptual ke Model Relasional

Translasi desain konseptual ke model relasional adalah proses mengubah gambaran awal database (biasanya berupa ERD) menjadi bentuk tabel-tabel yang bisa digunakan di sistem database seperti MySQL, PostgreSQL, atau SQL Server.

```
-- Tabel Dimensi
CREATE TABLE Dim_Waktu (
  ID_Waktu INT PRIMARY KEY,
  Tanggal DATE,
  Hari VARCHAR(20),
  Bulan INT,
  Tahun INT,
  Kuartal INT
);

CREATE TABLE Dim_Lokasi (
  ID_Lokasi INT PRIMARY KEY,
  Nama_Bandara VARCHAR(100),
  Kota VARCHAR(100),
  Negara VARCHAR(100)
);

CREATE TABLE Dim_Penerbangan (
  ID_Penerbangan INT PRIMARY KEY,
  Kode_Penerbangan VARCHAR(20),
  Maskapai VARCHAR(100),
  Waktu_Berangkat TIME,
  Waktu_Tiba TIME
);

CREATE TABLE Dim_Pesawat (
  ID_Pesawat INT PRIMARY KEY,
  Nomor_Pesawat VARCHAR(50),
  Tipe_Pesawat VARCHAR(100),
  Kapasitas INT
);

CREATE TABLE Dim_KelasLayanan (
  ID_KelasLayanan INT PRIMARY KEY,
  Nama_Kelas VARCHAR(50),
  Deskripsi TEXT
);

CREATE TABLE Dim_StatusPenerbangan (
  ID_StatusPenerbangan INT PRIMARY KEY,
  Status VARCHAR(50),
  Deskripsi TEXT
);

CREATE TABLE Fakta_Penerbangan (
```

```

ID_FaktaPenerbangan INT PRIMARY KEY,
ID_Waktu INT,
ID_Lokasi INT,
ID_Penerbangan INT,
ID_Pesawat INT,
ID_KelasLayanan INT,
ID_StatusPenerbangan INT,
Jumlah_Penumpang INT,
Pendapatan BIGINT,
Jumlah_Penerbangan INT,
FOREIGN KEY (ID_Waktu) REFERENCES Dim_Waktu(ID_Waktu),
FOREIGN KEY (ID_Lokasi) REFERENCES Dim_Lokasi(ID_Lokasi),
FOREIGN KEY (ID_Penerbangan) REFERENCES Dim_Penerbangan(ID_Penerbangan),
FOREIGN KEY (ID_Pesawat) REFERENCES Dim_Pesawat(ID_Pesawat),
FOREIGN KEY (ID_KelasLayanan) REFERENCES Dim_KelasLayanan(ID_KelasLayanan),
FOREIGN KEY (ID_StatusPenerbangan) REFERENCES
Dim_StatusPenerbangan(ID_StatusPenerbangan) );

```

### 3.2 Strategi Perancangan dan Implementasi Index

Indeks merupakan komponen penting dalam optimalisasi kinerja query pada gudang data. Dalam sistem penerbangan, indeks dirancang untuk mempercepat akses terhadap data yang sering digunakan dalam proses analisis, seperti pencarian penerbangan berdasarkan waktu, rute, atau jenis pesawat. Strategi perancangan indeks dimulai dengan mengidentifikasi kolom-kolom yang sering digunakan seperti WHERE, JOIN, dan GROUP BY, karena kolom-kolom inilah yang paling berdampak terhadap performa. Kolom yang diberi indeks antara lain adalah ID\_Waktu, ID\_Lokasi, ID\_Penerbangan, ID\_Pesawat, ID\_KelasLayanan, dan ID\_StatusPenerbangan.



```

-- Index pada kolom foreign key Fakta_Penerbangan CREATE INDEX idx_fp_id_waktu ON
Fakta_Penerbangan(ID_Waktu); CREATE INDEX idx_fp_id_lokasi ON
Fakta_Penerbangan(ID_Lokasi); CREATE INDEX idx_fp_id_penerbangan ON
Fakta_Penerbangan(ID_Penerbangan); CREATE INDEX idx_fp_id_pesawat ON
Fakta_Penerbangan(ID_Pesawat); CREATE INDEX idx_fp_id_kelaslayanan ON
Fakta_Penerbangan(ID_KelasLayanan); CREATE INDEX idx_fp_id_status ON
Fakta_Penerbangan(ID_StatusPenerbangan);

-- Index pada Dim_Waktu untuk query waktu CREATE INDEX idx_waktu_bulan_tahun ON
Dim_Waktu(Bulan, Tahun);

-- Index pada Maskapai untuk agregasi per maskapai CREATE INDEX
idx_penerbangan_maskapai ON Dim_Penerbangan(Maskapai);

-- Index pada Status untuk laporan status penerbangan CREATE INDEX
idx_status_penerbangan_status ON Dim_StatusPenerbangan(Status);

```

### 3.3 Optimalisasi Penyimpanan dan Organisasi Data

Dalam pengelolaan gudang data sistem penerbangan yang berskala besar dan bersifat transaksional serta analitis, optimalisasi penyimpanan dilakukan melalui pendekatan desain skema yang adaptif terhadap volume data dan kebutuhan analisis.

Pemilihan tipe data disesuaikan dengan sifat data yang disimpan. Atribut ID seperti ID\_Waktu, ID\_Pesawat, dan ID\_Penerbangan menggunakan tipe INT untuk efisiensi penyimpanan dan kecepatan akses. Untuk atribut deskriptif seperti Nama\_Bandara, Maskapai, dan Tipe\_Pesawat, digunakan tipe data VARCHAR dengan panjang yang ditentukan berdasarkan analisis data historis, guna menghindari alokasi memori yang berlebihan.

Berbeda dari pendekatan umum yang menggunakan FLOAT, nilai Pendapatan disimpan menggunakan tipe BIGINT untuk menjaga akurasi pada data finansial berskala besar dan menghindari pembulatan desimal yang dapat menimbulkan ketidaksesuaian laporan keuangan.

Optimalisasi juga dilakukan dengan mengelompokkan data berdasarkan kategori tertentu ke dalam tabel terpisah (sub-partisi manual), seperti memisahkan data penerbangan domestik dan internasional, atau berdasarkan kelas layanan. Hal ini bertujuan untuk mempercepat pencarian data spesifik tanpa harus melakukan pemindaian terhadap keseluruhan dataset.

Jika sistem basis data mendukung fitur lazy loading atau compression-aware scans (seperti pada SQL Server 2022 atau Oracle Exadata), maka fitur tersebut dapat diaktifkan untuk meningkatkan performa analitik tanpa memengaruhi performa penulisan data.

Secara fisik, sistem penyimpanan menggunakan konfigurasi RAID 10 untuk memberikan redundansi sekaligus performa baca-tulis tinggi. Hal ini penting dalam sistem penerbangan yang memerlukan ketersediaan tinggi dan ketahanan terhadap kehilangan data, terutama dalam sistem yang mencatat histori dan pergerakan penumpang secara real-time.

### 3.4 Desain Partisi Tabel dan View untuk Akses Efisien

Partisi data adalah proses membagi tabel besar menjadi bagian-bagian lebih kecil (partisi) untuk mempercepat dan mempermudah akses data dalam sistem basis data. Dalam kasus ini, partisi dapat diterapkan pada tabel Fakta\_Penerbangan berdasarkan kolom Tahun yang terdapat dalam tabel dimensi waktu. Dengan demikian, proses analisis performa penerbangan per tahun dapat dilakukan lebih cepat dan efisien karena hanya subset data dari tahun yang bersangkutan yang diproses oleh query.

Selain partisi, penggunaan view juga memberikan manfaat dalam menyederhanakan query SQL yang kompleks serta memudahkan penyajian data agregat kepada pengguna bisnis. View digunakan untuk menghasilkan ringkasan informasi dari data mentah dalam tabel fakta dan tabel dimensi. Salah satu contoh implementasi view dalam konteks penerbangan adalah untuk menampilkan total pendapatan dan jumlah penumpang per maskapai setiap tahunnya. View ini dapat digunakan oleh manajer maskapai atau analis bisnis untuk memantau kinerja operasional maskapai dari waktu ke waktu.

Berikut adalah contoh query SQL untuk membuat view yang menyajikan pendapatan dan jumlah penumpang per maskapai dan tahun:

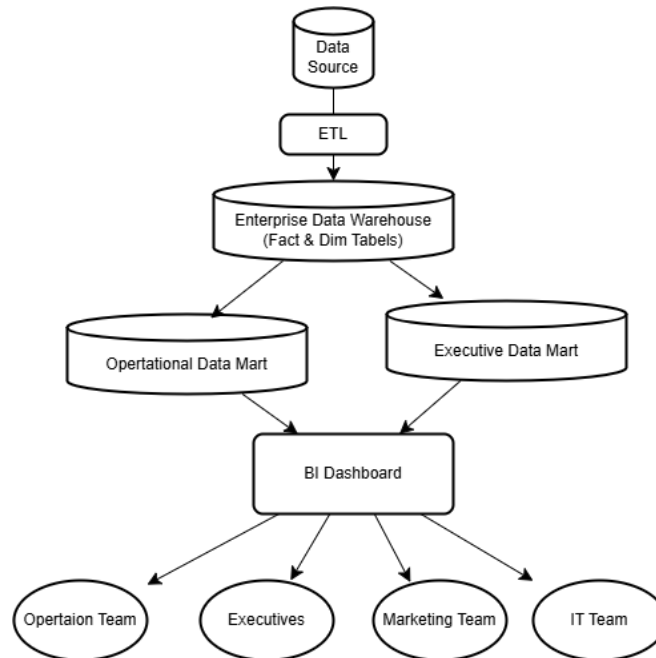
```
CREATE VIEW View_Kinerja_Maskapai_Tahunan AS
SELECT
dw.Tahun,
dp.Maskapai,
SUM(fp.Pendapatan) AS TotalPendapatan,
SUM(fp.Jumlah_Penumpang) AS TotalPenumpang
FROM Fakta_Penerbangan fp
JOIN Dim_Waktu dw ON fp.ID_Waktu = dw.ID_Waktu
JOIN Dim_Penerbangan dp ON fp.ID_Penerbangan = dp.ID_Penerbangan
GROUP BY dw.Tahun, dp.Maskapai;
```

## BAB 4

### ARSITEKTUR SISTEM DAN ETL

#### 4.1 Arsitektur Data Warehouse

Setelah merancang elemen-elemen utama dari data warehouse, diagram berikut menggambarkan interaksi antar komponen sistem dalam arsitektur ini.



Gambar. 2

Diagram menunjukkan bagaimana data dari berbagai sistem operasional penerbangan seperti sistem reservasi, operasional bandara, dan pelaporan keuangan, diproses melalui tahapan ETL (Extract, Transform, Load). Setelah melalui proses ini, data disimpan ke dalam Enterprise Data Warehouse. Data yang sudah terstruktur ini digunakan untuk membangun beberapa data mart, misalnya untuk kebutuhan divisi Operasional, Keuangan, dan Manajemen Eksekutif. Data dari data mart ini selanjutnya dianalisis melalui BI Dashboard seperti Power BI atau Tableau untuk mendukung pengambilan keputusan strategis, misalnya dalam perencanaan rute, efisiensi biaya, dan peningkatan layanan pelanggan.

#### 4.2 Perancangan dan Implementasi Proses ETL Menjelaskan tahapan extract, transform, dan load secara teknis dan praktis.

##### A. Extract (Ekstraksi Data)

Pada tahap ini, data diambil dari berbagai sistem sumber utama, yaitu:

1. Sistem Reservasi Penerbangan (untuk data penumpang, tiket, kelas layanan),
2. Sistem Operasional Bandara & ATC (untuk data jadwal, status penerbangan, lokasi),
3. Sistem Manajemen Armada (untuk data jenis dan spesifikasi pesawat),
4. Sistem Pelaporan Keuangan (untuk data pendapatan dan biaya operasional).

Proses ekstraksi menggunakan alat bantu seperti Python dan pustaka Pandas atau ETL tools seperti Talend, Apache NiFi, atau SSIS. Data dikumpulkan ke dalam staging area sebagai tahap

awal sebelum transformasi.

## B. Transform (Transformasi Data)

Data yang telah diekstrak akan dibersihkan dan diubah agar sesuai dengan struktur skema star schema dalam data warehouse. Beberapa proses transformasi yang dilakukan antara lain:

1. Pembersihan Data: Menghapus duplikasi, memperbaiki inkonsistensi (misal penulisan nama bandara atau kode pesawat).
2. Normalisasi Format: Konversi format waktu keberangkatan/kedatangan menjadi standar datetime, mengubah angka dari string ke integer/float.
3. Enrichment dan Mapping: Menghitung keterlambatan berdasarkan selisih jadwal dan waktu aktual; menghubungkan data ke tabel dimensi (seperti Dim\_Waktu, Dim\_Lokasi, Dim\_Penerbangan, Dim\_Pesawat, Dim\_StatusPenerbangan, dan Dim\_KelasLayanan).

## C. Load (Pemuatan Data)

Data hasil transformasi dimuat ke dalam Enterprise Data Warehouse. Tabel fakta seperti Fakta\_Penerbangan akan diisi dengan informasi numerik seperti:

- Jumlah\_Penumpang
- Pendapatan
- Biaya\_Operasional
- Keterlambatan (menit)

Tabel dimensi seperti Dim\_Waktu, Dim\_Lokasi, dan lainnya diisi terlebih dahulu, agar bisa direferensikan oleh tabel fakta melalui foreign key. Proses loading ini bisa dijadwalkan secara harian atau real-time tergantung kebutuhan analitik. Penggunaan teknik incremental load juga dipertimbangkan agar efisien.

## **LAMPIRAN**

Berikut merupakan link datasetnya:

<https://drive.google.com/drive/folders/1oFN4WFO6lZSKh9gCyviUGovs1tBwI9PR>