

LAPORAN TUGAS BESAR
MATA KULIAH
DATA WAREHOUSE AND BUSINESS INTELLIGENCE

Dosen Pengampu:
Dr. Abdullah Fajar, S.Si, M.Sc.



Disusun Oleh
Kelas SI4706 - Kelompok 5

Ferdiansyah Adi Saputra	102022330113
Kenneth Bryan	102022330093
Syarief Saleh Madhy	102022300201
Winsenlaus Alfero Krisna I. L	102022300008
Ardy Maulana Nayottama N	102022300050

PROGRAM STUDI S1 SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
TELKOM UNIVERSITY
BANDUNG
2025

KATA PENGANTAR

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa, karena atas limpahan rahmatnya penyusun dapat menyelesaikan laporan tugas besar mata kuliah *Data Warehouse and Business Intelligence* ini tepat waktu tanpa ada halangan yang berarti dan sesuai dengan harapan penyusun.

Ucapan terima kasih kami sampaikan kepada bapak Dr. Abdullah Fajar, S.Si, M.Sc. sebagai dosen pengampu mata kuliah *Data Warehouse and Business Intelligence* yang telah membantu memberikan arahan dan pemahaman dalam penyusunan laporan tugas besar ini.

Kami menyadari bahwa dalam penyusunan laporan tugas besar ini masih banyak kekurangan karena keterbatasan kami. Maka dari itu penyusun sangat mengharapkan kritik dan saran untuk menyempurnakan laporan tugas besar ini. Semoga apa yang ditulis dapat bermanfaat bagi semua pihak yang membutuhkan.

Bandung, 1 Desember 2025

Kelompok 5

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR.....	iv
DAFTAR TABEL	v
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Tujuan Penulisan.....	1
1.3 Ruang Lingkup dan Batasan Masalah.....	2
1.4 Metodologi.....	2
BAB II PERUMUSAN OBJECTIVE, BALANCED SCORECARD, DAN KPI.....	4
2.1 Konsep Balance Scorecard	4
2.2 Penentuan Objective	5
2.3 Perumusan Key Performance Indicator (KPI)	5
BAB III ANALISIS SUMBER DATA	7
3.1 Deskripsi Sumber Data Utama.....	7
3.2 Deskripsi Sumber Data Pendukung	7
3.3 Exploratory Data Analysis (EDA).....	8
BAB IV PERANCANGAN DAN IMPLEMENTASI DATA WAREHOUSE	11
4.1 Perancangan Star Schema	11
4.2 Implementasi Star Schema pada RDBMS	14
BAB V IMPLEMENTASI PROSES ETL	17
5.1 Proses Extract.....	17
5.2 Proses Transform	17
5.3 Proses Load.....	20
BAB VI IMPLEMENTASI DATA MINING	21
6.1 Pemilihan Metode Data Mining.....	21

6.2	Proses Pengolahan Data	24
6.3	Hasil dan Analisis Data Mining	26
BAB VII PERANCANGAN DAN ANALISIS DASHBOARD KPI.....		29
7.1	Desain dan Implementasi Dashboard.....	29
7.2	Analisis Pencapaian KPI.....	30
BAB VIII PENUTUP.....		36
8.1	Kesimpulan	36
8.2	Saran	36
DAFTAR PUSTAKA.....		38
LAMPIRAN.....		39

DAFTAR GAMBAR

Gambar 3.1 Statistik Deskriptif Data.....	9
Gambar 3.2 Rata-rata Profit berdasarkan Kategori.....	9
Gambar 3.3 Tren Profit vs Besaran Diskon	10
Gambar 3.4 Identifikasi Missing Value	10
Gambar 4.1 Rancangan Star Schema.....	11
Gambar 4.2 Implementasi Star Schema pada RDBMS	14
Gambar 5.1 Proses Extract Dataset.....	17
Gambar 5.2 Proses Transform ke Tabel Dimensi.....	17
Gambar 5.3 Proses Transformasi ke Tabel Fakta	19
Gambar 6.1 Random Forest Classifier.....	22
Gambar 6.2 Prediksi Total Penjualan (Sales Forecasting).....	23
Gambar 6.3 Pra-pemrosesan Data untuk Klasifikasi	25
Gambar 6.4 Pra-pemrosesan Data untuk Estimasi.....	26
Gambar 6.5 Hasil Klasifikasi Profitabilitas	27
Gambar 6.6 Hasil Estimasi Penjualan.....	28
Gambar 7.1 Dashboard Eksekutif 7-Eleven.....	29
Gambar 7.2 Dashboard Eksekutif 7-Eleven.....	30
Gambar 7.3 Chart KPI Total Sales	31
Gambar 7.4 Chart KPI Sales Growth Rate	31
Gambar 7.5 Chart KPI Total Profit.....	32
Gambar 7.6 Chart KPI Profit Margin	32
Gambar 7.7 Chart KPI Total Order.....	33
Gambar 7.8 Chart KPI Average Order Value (AOV).....	33
Gambar 7.9 Chart KPI Profit per Category	34
Gambar 7.10 Chart KPI Discount Effectiveness	34
Gambar 7.11 Chart KPI Top Product Contribution	35
Gambar 7.12 Chart KPI Regional Sales Performance	35

DAFTAR TABEL

Tabel 2.1 Balance Scorecard.....	4
Tabel 2.2 Key Performance Indicator (KPI).....	5
Tabel 4.1 Fact_Sales	12
Tabel 4.2 Dimensi Order.....	12
Tabel 4.3 Dimensi Customer	12
Tabel 4.4 Dimensi Store	13
Tabel 4.5 Dimensi Location.....	13
Tabel 4.6 Dimensi Product	13
Tabel 6.1 Kesimpulan Implikasi terhadap KPI Finansial	28

BAB I

PENDAHULUAN

1.1 Latar Belakang

Peralihan transaksi bisnis dari tradisional ke modern membuka banyak sekali peluang bisnis, hal ini membuat banyak pelaku bisnis punya kesempatan yang besar untuk berkembang tidak terkecuali pelaku bisnis yang mempunyai supermarket. Supermarket merupakan toko besar yang menjual berbagai macam kebutuhan sehari-hari, serta menawarkan banyak pilihan produk bagi pelanggannya. Supermarket menawarkan benefit seperti kenyamanan berbelanja, kemudahan berbelanja, kualitas barang yang lebih bagus, hingga harga yang bersaing.

Dengan berjalannya waktu peluang bisnis makin meningkat, maka menjadikan data penjualan semakin kompleks. Retail besar seperti supermarket memerlukan informasi yang mudah dipahami seperti dashboard. Dashboard berperan dalam memberikan tampilan interface dari visual data data yang digabungkan sesuai dengan keperluan. Hasil yang diharapkan bagi supermarket yaitu memperoleh informasi yang berguna dalam mengambil keputusan strategi bisnis di masa depan.

Pada tugas besar ini dilakukan implementasi teknologi data warehouse yang mencakup ekstraksi, pemrosesan, hingga penyajian informasi yang detail dan efisien sesuai dengan yang dibutuhkan oleh supermarket. Lalu dipadukan dengan business intelligence untuk meningkatkan efisiensi operasional supermarket serta mendapatkan profit lewat pengambilan keputusan yang presisi.

1.2 Tujuan Penulisan

Tujuan dari tugas besar ini antara lain sebagai berikut:

1. Merancang dan membangun data warehouse yang terstruktur dari data transaksi penjualannya supermarket sebagai sumber data yang terintegrasi.
2. Melakukan proses extract, transform, load (ETL) untuk mengolah data mentah menjadi data yang siap dianalisis.
3. Menentukan KPI (Key Performance Indicator) yang relevan untuk evaluasi kinerja bisnis supermarket
4. Mengembangkan dashboard business intelligence yang informatif dan mudah dipahami sebagai alat bantu visualisasi data.
5. Memberi insight bisnis berdasarkan analisis data guna mendukung pengambilan keputusan strategis bisnis oleh pihak manajemen.

1.3 Ruang Lingkup dan Batasan Masalah

Agar pembahasan dalam tugas besar ini lebih fokus dan terarah serta sesuai dengan tujuan yang telah diterapkan, maka ruang lingkup dan batasan masalah ditentukan sebagai berikut:

Ruang Lingkup:

1. Data yang digunakan adalah superstore dataset yang berisi transaksi penjualan, pelanggan, produk dan wilayah.
2. Implementasi data warehouse difokuskan pada perancangan skema data.
3. Analisis menggunakan pendekatan business intelligence, termasuk perhitungan dan visualisasi KPI.
4. Penyajian hasil analisis dilakukan dalam bentuk dashboard untuk mendukung evaluasi kinerja bisnis.

Batasan masalah:

1. Data yang dianalisis terbatas pada dataset yang sudah ada dan tidak mencakup data eksternal tambahan.
2. Analisis difokuskan pada data historis dan tidak mencakup prediksi/forecasting.
3. KPI yang dianalisis dibatasi pada indikator utama seperti penjualan, profit, jumlah transaksi, dan performa wilayah.
4. Implementasi sistem hanya bersifat simulasi akademik dan tidak diterapkan langsung pada sistem operasional nyata supermarket.

1.4 Metodologi

Metodologi yang digunakan dalam tugas besar ini bertujuan untuk menghasilkan sistem data warehouse dan business intelligence yang mampu mendukung analisis kinerja bisnis supermarket. Metodologi penelitian dilakukan melalui beberapa tahapan sebagai berikut:

1. Pengumpulan Data

Tahap awal penelitian adalah melakukan pengumpulan data yang digunakan sebagai objek analisis. Data yang digunakan dalam tugas besar ini merupakan superset dataset, didapat dari website [Kaggle.com](https://www.kaggle.com) yang berisi data transaksi penjualan, pelanggan, produk, dan wilayah. Data tersebut menjadi sumber utama dalam proses perancangan data warehouse dan analisis bisnis.

2. Perancangan data warehouse

Di tahap ini dilakukan perancangan arsitektur data warehouse, termasuk dengan penentuan tabel fakta dan dimensi, serta pemilihan skema data yang sesuai. Perancangan

ini bertujuan untuk menghasilkan struktur penyimpanan data yang terintegrasi, konsisten, dan mendukung kebutuhan analisis bisnis.

3. ETL

Tahap ini melakukan ekstraksi data dari sumber data, melakukan pembersihan serta transformasi. Kemudian membuat data tersebut ke dalam data warehouse. Proses ini bertujuan untuk memastikan data yang digunakan dalam analisis memiliki kualitas yang baik dan siap diolah.

4. Data Mining dan Analisis KPI

Setelah data tersimpan dalam data warehouse, dilakukan proses analisis data dengan fokus pada pengukuran KPI. Analisis bertujuan untuk mengevaluasi kinerja bisnis supermarket berdasarkan indikator utama seperti penjualan, profit, jumlah transaksi, dan performa wilayah.

5. Dashboard

Tahap akhir ini merancang dan membangun dashboard business intelligence sebagai media visualisasi hasil analisis. Dashboard ini menyajikan informasi dalam bentuk grafik dan indikator yang mudah dipahami sehingga dapat membantu pihak manajemen dalam pengambilan keputusan.

BAB II

PERUMUSAN OBJECTIVE, BALANCED SCORECARD, DAN KPI

2.1 Konsep Balance Scorecard

Balanced Scorecard (BSC) merupakan kerangka kerja strategis yang digunakan untuk mengukur kinerja organisasi secara komprehensif dengan mengintegrasikan indikator keuangan dan non-keuangan. Konsep ini diperkenalkan oleh Kaplan dan Norton dan telah banyak digunakan dalam penelitian akademik serta praktik bisnis modern untuk memastikan keselarasan antara strategi dan implementasi operasional.

Menurut Kaplan dan Norton (1996), Balanced Scorecard berfungsi sebagai sistem pengukuran kinerja yang menerjemahkan visi dan strategi organisasi ke dalam tujuan strategis dan indikator yang terukur. Perspektif keuangan tetap menjadi fokus utama, namun didukung oleh perspektif lain agar pencapaian kinerja bersifat berkelanjutan.

Pada tugas besar ini, perspektif Financial dipilih sebagai fokus utama karena dataset *Sample Superstore* menyediakan atribut transaksi yang secara langsung mencerminkan performa keuangan perusahaan ritel, seperti Sales, Profit, Discount, dan Quantity. Perspektif ini relevan untuk mengevaluasi efektivitas strategi penjualan, profitabilitas produk, serta kebijakan diskon. Berikut ini adalah tabel balance scorecard yang ditujukan penulis:

Tabel 2.1 Balance Scorecard

Perspektif	Objektivitas	Tujuan Strategis	Indikator (KPI)	Inisiatif
Financial	Peningkatan pendapatan	Meningkatkan total penjualan	Total Sales	Optimalisasi strategi produk dan wilayah
	Profitabilitas	Meningkatkan laba perusahaan	Total Profit	Evaluasi profit produk dan diskon
	Efisiensi penjualan	Menjaga margin keuntungan	Profit Margin	Pengendalian diskon
	Pertumbuhan bisnis	Meningkatkan pertumbuhan penjualan	Sales Growth Rate	Analisis tren penjualan
	Efektivitas harga	Mengurangi dampak negatif diskon	Average Discount	Penyesuaian kebijakan diskon

2.2 Penentuan Objective

Berdasarkan perspektif Financial dalam Balanced Scorecard, objective utama yang dirumuskan dalam penelitian ini adalah: “Menganalisis dan meningkatkan kinerja finansial perusahaan ritel Superstore melalui optimalisasi penjualan, profit, dan strategi diskon berbasis data.” Objective tersebut diturunkan dari kebutuhan manajemen untuk memahami kondisi keuangan perusahaan secara komprehensif serta mengidentifikasi faktor-faktor yang memengaruhi pencapaian profit. Melalui pemanfaatan data warehouse dan teknik analisis data, perusahaan diharapkan mampu memperoleh insight yang mendukung pengambilan keputusan strategis, khususnya dalam pengelolaan produk, wilayah penjualan, dan kebijakan diskon.

Objective ini juga selaras dengan tujuan Business Intelligence, yaitu menyediakan informasi yang akurat, relevan, dan tepat waktu guna mendukung keputusan manajerial berbasis data.

2.3 Perumusan Key Performance Indicator (KPI)

Untuk mengukur ketercapaian objective yang telah ditetapkan, dirumuskan beberapa Key Performance Indicator (KPI) yang relevan dengan perspektif Financial. KPI disusun berdasarkan atribut yang tersedia dalam dataset *Sample Superstore* dan dirancang untuk dianalisis secara deskriptif maupun prediktif. Berikut adalah tabel Key Performance Indicator (KPI) yang dirancang oleh penulis:

Tabel 2.2 Key Performance Indicator (KPI)

Goals	KPI	Data	Perhitungan	Chart
Meningkatkan pendapatan	Pertumbuhan Total Penjualan meningkat $\geq 10\%$ per tahun	Sales	Actual: Σ SalesTarget: $+10\%$ per tahun	Gauge / Line
Meningkatkan profit	Total profit perusahaan > 0 pada setiap periode	Profit	Actual: Σ ProfitTarget: Profit > 0	Gauge
Efisiensi penjualan	Profit Margin, semakin tinggi semakin efisien	Sales, Profit	$(\text{Profit} / \text{Sales}) \times 100\%$	Bar
Pertumbuhan bisnis	Sales Growth Rate	Order Date, Sales	$((\text{Sales}_t - \text{Sales}_{t-1}) / \text{Sales}_{t-1}) \times 100\%$	Line

Goals	KPI	Data	Perhitungan	Chart
Efektivitas diskon	Nilai rata-rata profit yang didapat per diskon apakah optimal	Discount, Profit	AVG(Discount)	Line
Optimalisasi produk	Total profit yang dihasilkan setiap kategori produk	Category, Profit	Σ Profit per Category	Bar
Kinerja wilayah	Total penjualan berdasarkan wilayah/kota	City, Sales	Σ Sales per City	Map / Bar
Produktivitas penjualan	Rata-rata Nilai Penjualan per Order	Order ID, Sales	Σ Sales / Count(Order ID)	Combo Chart
Volume transaksi	Total Orders	Order ID	Count(Distinct Order ID)	KPI Card
Efisiensi Profit per Unit Produk	Profit per Quantity by Sub-Category (Category Breakdown)	Quantity, Profit	Σ Profit / Σ Quantity	Bar

BAB III

ANALISIS SUMBER DATA

3.1 Deskripsi Sumber Data Utama

Sumber data utama yang digunakan dalam pengembangan sistem Data Warehouse ini berasal dari dataset transaksi retail perusahaan yang kami dapatkan bersumber dari situs Kaggle: <https://www.kaggle.com/datasets/jacopoferretti/superstore-dataset>. Data ini merupakan data transaksional yang mencerminkan aktivitas operasional sebuah perusahaan retail selama periode empat tahun (2014-2017). Dataset ini memiliki struktur yang sangat baik untuk diolah ke dalam *Data Warehouse* karena mencakup berbagai dimensi bisnis (produk, pelanggan, lokasi, dan waktu).

Dataset ini terdiri dari 21 kolom yang mencatat detail setiap transaksi. Berikut beberapa kolom kunci yang diidentifikasi:

- Order ID & Date: Identitas unik pesanan dan tanggal transaksi dilakukan
- Customer ID & Name: Informasi mengenai pelanggan yang melakukan pembelian
- Product Information: Terdiri dari ID produk, kategori (Furniture, Office Supplies, Technology), sub-kategori, dan nama produk
- Sales, Quantity, Discount & Profit: Metrik numerik yang menunjukkan nilai penjualan, jumlah barang yang terjual, diskon yang diberikan, dan keuntungan yang dihasilkan dari setiap transaksi. *Sales* merepresentasikan pendapatan kotor, *Quantity* menunjukkan volume penjualan, *Discount* mencerminkan strategi promosi yang diterapkan, dan *Profit* menunjukkan kinerja finansial bersih.
- Geographic Data: Mencakup Negara, Kota, Negara bagian, Kode Pos, dan Wilayah (*Region*) untuk analisis berbasis lokal. Informasi geografis ini memungkinkan analisis distribusi penjualan dan profit berdasarkan wilayah, sehingga perusahaan dapat mengidentifikasi area dengan performa terbaik maupun wilayah dengan potensi peningkatan.

3.2 Deskripsi Sumber Data Pendukung

Sumber data pendukung ini adalah data utama yang telah dipecah sesuai dengan kebutuhan manajerial perusahaan untuk dianalisis demi mencapai target Business Intelligence. Berikut ini adalah tabel hasil pemecahan data utama:

1. Tabel Customer

Menyimpan informasi pelanggan dan segmentasinya. Atribut utama:

- customer_id
- customer_name
- segment

2. Tabel Location

Menyimpan informasi geografis penjualan. Atribut utama:

- country
- state
- city
- postal_code
- region

3. Tabel Store

Merepresentasikan metode distribusi dan wilayah operasional. Atribut utama:

- ship_mode
- region

4. Tabel Order

Menyimpan informasi transaksi dan waktu pemesanan. Atribut utama:

- order_id
- order_date
- ship_date
- row_id

5. Tabel Product (dim_product)

Menyimpan detail produk. Atribut utama:

- product_id
- product_name
- category
- sub_category

3.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) adalah proses analisis awal data yang bertujuan untuk memahami karakteristik, struktur, dan komponen penting dari *dataset* sebelum melakukan analisis statistik atau pemodelan prediktif lebih lanjut. Sub bab ini menyajikan hasil eksplorasi data menggunakan Google Colab untuk memahami karakteristik data transaksi dan memvalidasi metrik KPI yang telah ditetapkan.

1. Statistik Deskriptif Data

Langkah awal EDA adalah melihat rangkuman statistik untuk memastikan tidak ada data yang aneh (noise) dan memahami sebaran angka pada penjualan serta keuntungan.

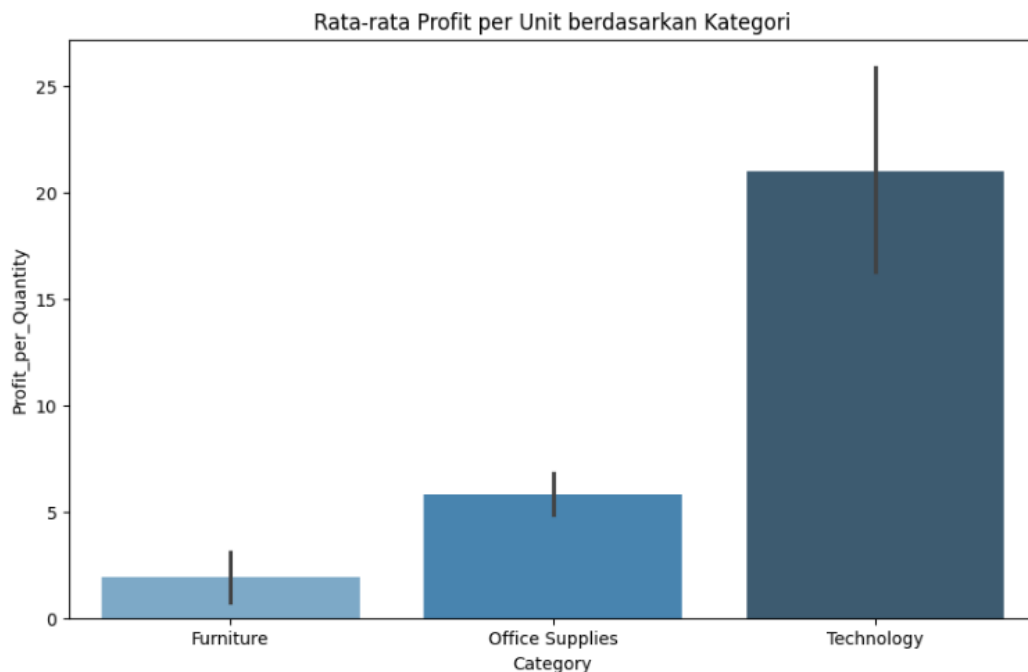
	Sales	Quantity	Profit	Profit_per_Quantity
count	9994.000000	9994.000000	9994.000000	9994.000000
mean	229.858001	3.789574	28.656896	7.799372
std	623.245101	2.225110	234.260108	56.074974
min	0.444000	1.000000	-6599.978000	-1319.995600
25%	17.280000	2.000000	1.728750	0.722800
50%	54.490000	3.000000	8.666500	2.767000
75%	209.940000	5.000000	29.364000	8.703200
max	22638.480000	14.000000	8399.976000	1679.995200

Gambar 3.1 Statistik Deskriptif Data

2. Visualisasi Awal

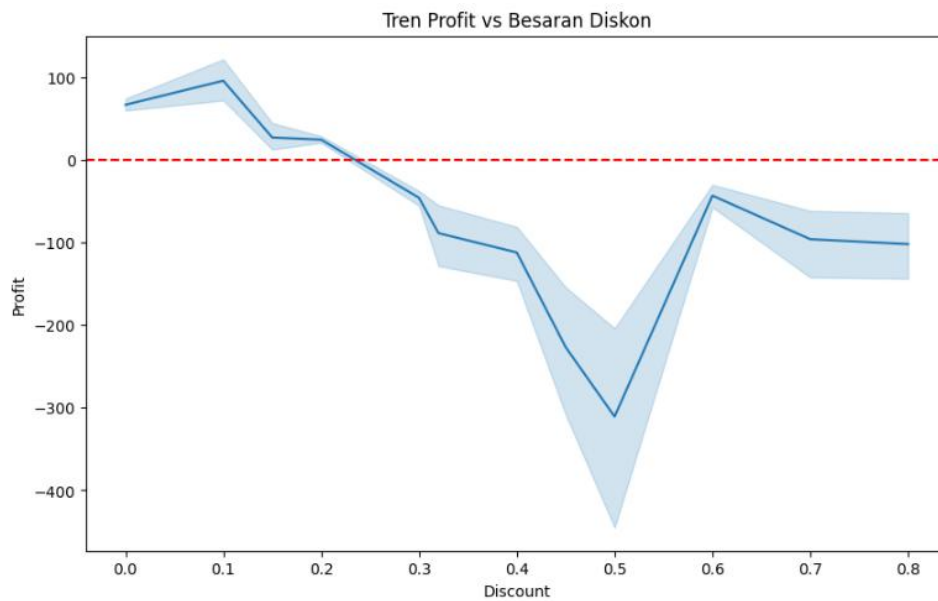
- Analisis Profitable per Kategori (KPI: Profit per Quantity)

Analisis ini bertujuan untuk melihat kategori mana yang memberikan keuntungan paling efektif per satu unit barang.



Gambar 3.2 Rata-rata Profit berdasarkan Kategori

- Analisis Efisiensi Distribusi per Wilayah (KPI: Efisiensi Kuantitas)



Gambar 3.3 Tren Profit vs Besaran Diskon

Ditemukan pola kritis di mana profit perusahaan mulai menurun tajam dan menjadi negatif (rugi) ketika diskon diberikan di atas 20%. Informasi ini sangat penting untuk tahap selanjutnya dalam menentukan ambang batas (threshold) diskon pada dashboard manajemen agar kerugian dapat diminimalisir.

3. Identifikasi missing value

- Missing Value



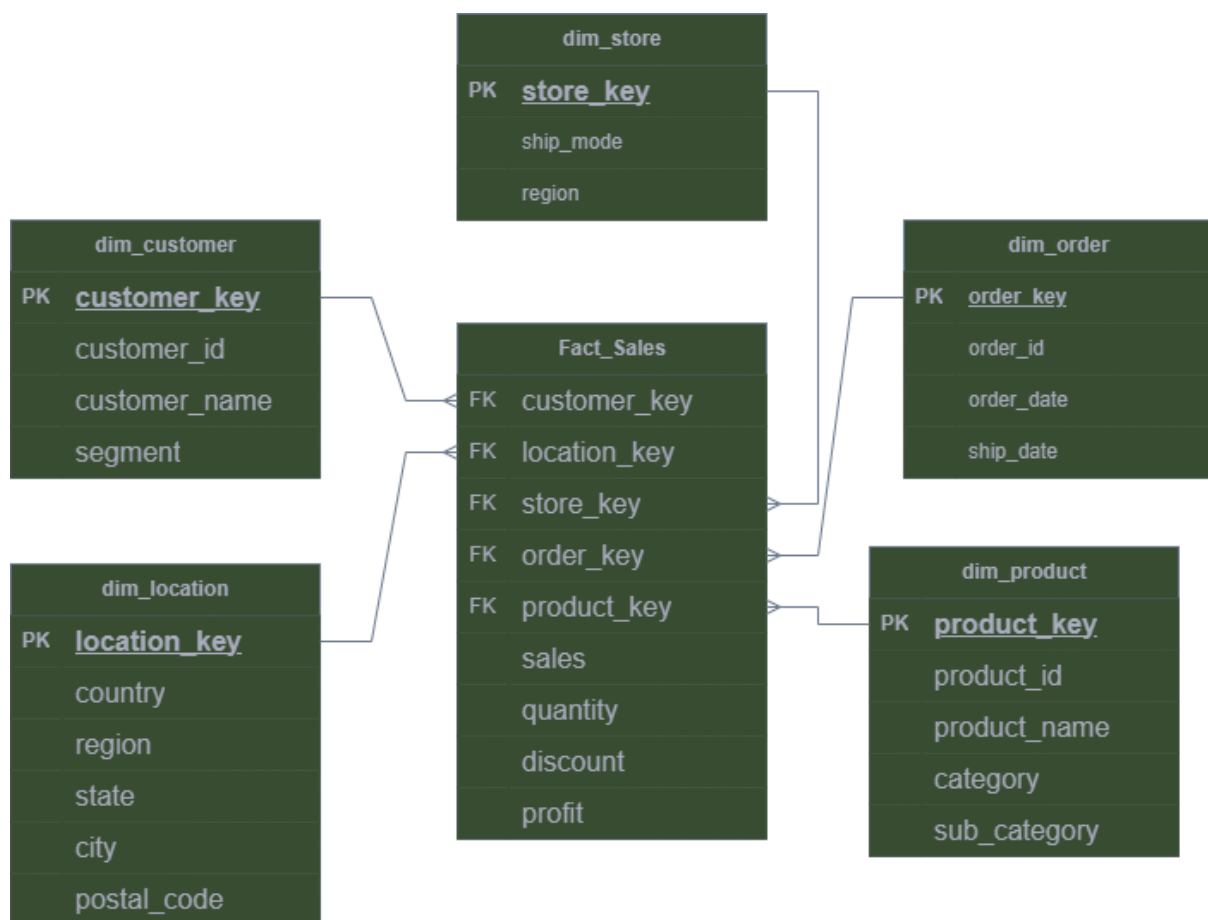
Gambar 3.4 Identifikasi Missing Value

BAB IV

PERANCANGAN DAN IMPLEMENTASI DATA WAREHOUSE

4.1 Perancangan Star Schema

Perancangan data warehouse pada penelitian ini mengacu pada pendekatan dimensional modeling yang diperkenalkan oleh Kimball, di mana data disusun dalam bentuk star schema untuk mendukung analisis Business Intelligence secara efisien. Star schema dipilih karena memiliki struktur yang sederhana, performa query yang tinggi, serta mudah dipahami oleh pengguna non-teknis.



Gambar 4.1 Rancangan Star Schema

Star schema terdiri dari satu tabel fakta (fact table) yang menyimpan data numerik utama (measures) dan beberapa tabel dimensi (dimension tables) yang menyediakan konteks deskriptif terhadap data fakta tersebut. Berikut ini merupakan rincian dari setiap tabel yang ada pada star schema

1. Tabel Fakta

Tabel fakta yang dirancang adalah Fact_Sales, yang merepresentasikan seluruh transaksi penjualan pada dataset Superstore. Tabel fakta ini bersifat additive, sehingga seluruh measure dapat dijumlahkan untuk analisis lintas dimensi dan waktu.

Tabel 4.1 Fact_Sales

Atribut	Deskripsi
sales	Nilai penjualan
quantity	Jumlah produk terjual
discount	Besaran diskon
profit	Laba penjualan
customer_key	Foreign key ke dim_customer
location_key	Foreign key ke dim_location
store_key	Foreign key ke dim_store
order_key	Foreign key ke dim_order
product_key	Foreign key ke dim_product

2. Tabel Dimensi

Berdasarkan analisis dataset dan kebutuhan KPI perspektif Financial, dirancang lima tabel dimensi sebagai berikut:

1) Tabel Dimensi Order (dim_order)

Tabel 4.2 Dimensi Order

Kolom	Keterangan
order_key (PK)	Surrogate key
order_id	ID pesanan
order_date	Tanggal pemesanan
ship_date	Tanggal pengiriman

2) Tabel Dimensi Customer (dim_customer)

Tabel 4.3 Dimensi Customer

Kolom	Keterangan
customer_key (PK)	Surrogate key
customer_id	ID pelanggan asli
customer_name	Nama pelanggan
segment	Segmen pelanggan (Consumer, Corporate, Home Office)

3) Tabel Dimensi Store (dim_store)

Tabel 4.4 Dimensi Store

Kolom	Keterangan
store_key (PK)	Surrogate key
ship_mode	Metode pengiriman

4) Tabel Dimensi Location (dim_location)

Tabel 4.5 Dimensi Location

Kolom	Keterangan
location_key (PK)	Surrogate key
country	Negara
region	Region penjualan
state	Provinsi / State
city	Kota
postal_code	Kode pos

5) Tabel Dimensi Product (dim_product)

Tabel 4.6 Dimensi Product

Kolom	Keterangan
product_key (PK)	Surrogate key
product_id	ID produk
product_name	Nama produk
category	Kategori produk
sub_category	Sub-kategori produk

3. Measures

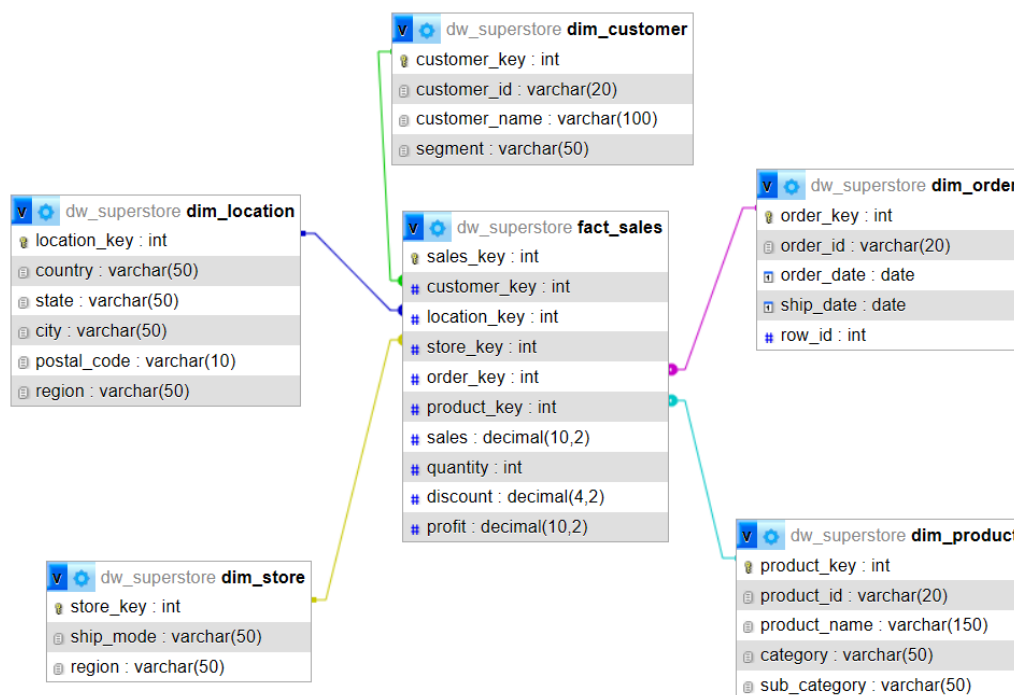
Measures pada tabel fakta dipilih berdasarkan kebutuhan analisis KPI Financial, yaitu:

- Sales untuk mengukur pendapatan
- Profit untuk mengukur laba
- Quantity untuk mengukur volume penjualan
- Discount untuk mengukur efektivitas strategi harga

Measures tersebut digunakan untuk menghitung KPI seperti total revenue, profit margin, sales growth, dan efektivitas diskon.

4.2 Implementasi Star Schema pada RDBMS

Implementasi data warehouse pada penelitian ini dilakukan menggunakan MySQL sebagai *Relational Database Management System* (RDBMS). Pemilihan MySQL didasarkan pada kemampuannya dalam mendukung standar *Structured Query Language* (SQL) secara penuh, sehingga memudahkan proses perancangan dan pengelolaan struktur data. Selain itu, MySQL dikenal sebagai sistem basis data yang stabil dan banyak digunakan dalam penelitian akademik maupun pengembangan sistem informasi, sehingga memiliki dokumentasi dan dukungan komunitas yang luas. MySQL juga mudah diintegrasikan dengan berbagai *tools* Extract, Transform, Load (ETL) serta platform *Business Intelligence* (BI), yang menjadikannya sesuai untuk implementasi data warehouse dan pengembangan dashboard analitik.



Gambar 4.2 Implementasi Star Schema pada RDBMS

Struktur database dirancang dengan menerapkan surrogate key pada setiap tabel dimensi untuk meningkatkan performa query serta menjaga konsistensi data historis, khususnya dalam menangani perubahan data yang bersifat *slowly changing dimension*. Penggunaan surrogate key memungkinkan pemisahan antara identitas bisnis dari sistem sumber dan identitas internal data warehouse, sehingga integritas data tetap terjaga. Secara keseluruhan, struktur database yang dibangun terdiri dari lima tabel dimensi dan satu tabel fakta, yang saling terhubung membentuk skema bintang (*star schema*). Berikut ini adalah Script DDL Implementasi Star Schema:

1. Tabel Dimensi Customer

```
CREATE TABLE dim_customer (  
    customer_key INT AUTO_INCREMENT PRIMARY KEY,  
    customer_id VARCHAR(20),  
    customer_name VARCHAR(100),  
    segment VARCHAR(50)  
);
```

2. Tabel Dimensi Location

```
CREATE TABLE dim_location (  
    location_key INT AUTO_INCREMENT PRIMARY KEY,  
    country VARCHAR(50),  
    state VARCHAR(50),  
    city VARCHAR(50),  
    postal_code VARCHAR(10),  
    region VARCHAR(50)  
);
```

3. Tabel Dimensi Store

```
CREATE TABLE dim_store (  
    store_key INT AUTO_INCREMENT PRIMARY KEY,  
    ship_mode VARCHAR(50),  
    region VARCHAR(50)  
);
```

4. Tabel Dimensi Order

```
CREATE TABLE dim_order (  
    order_key INT AUTO_INCREMENT PRIMARY KEY,  
    order_id VARCHAR(20),  
    order_date DATE,  
    ship_date DATE,  
    row_id INT  
);
```

5. Tabel Dimensi Product

```
CREATE TABLE dim_product (  
    product_key INT AUTO_INCREMENT PRIMARY KEY,  
    product_id VARCHAR(20),  
    product_name VARCHAR(150),  
    category VARCHAR(50),  
    sub_category VARCHAR(50)  
);
```

6. Tabel Fakta

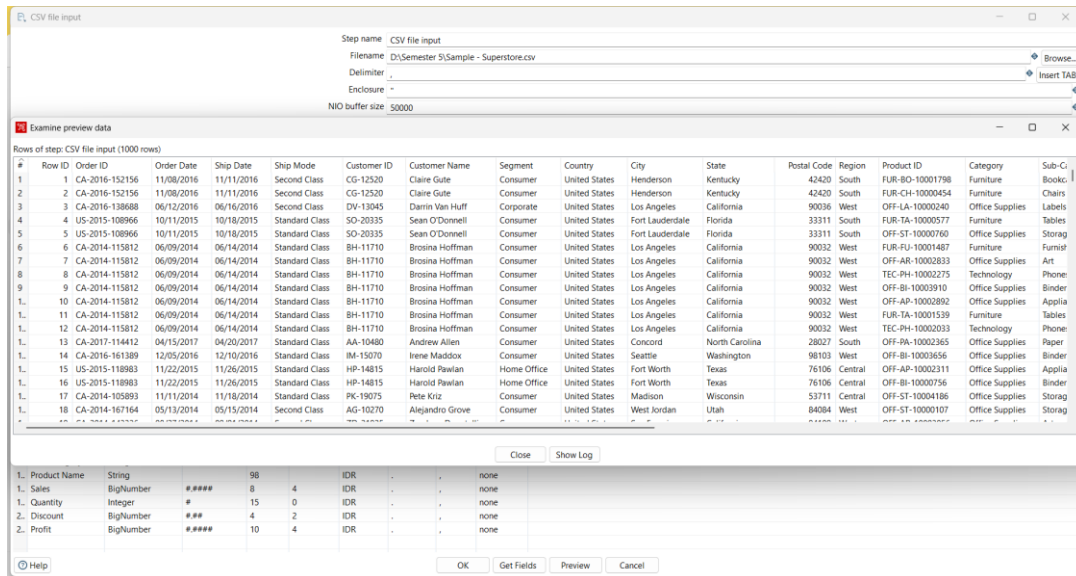
```
CREATE TABLE fact_sales (  
    sales_key INT AUTO_INCREMENT PRIMARY KEY,  
    customer_key INT,  
    location_key INT,  
    store_key INT,  
    order_key INT,  
    product_key INT,  
    sales DECIMAL(10,2),  
    quantity INT,  
    discount DECIMAL(4,2),  
    profit DECIMAL(10,2),  
    FOREIGN KEY (customer_key) REFERENCES dim_customer(customer_key),  
    FOREIGN KEY (location_key) REFERENCES dim_location(location_key),  
    FOREIGN KEY (store_key) REFERENCES dim_store(store_key),  
    FOREIGN KEY (order_key) REFERENCES dim_order(order_key),  
    FOREIGN KEY (product_key) REFERENCES dim_product(product_key)  
);
```

BAB V

IMPLEMENTASI PROSES ETL

5.1 Proses Extract

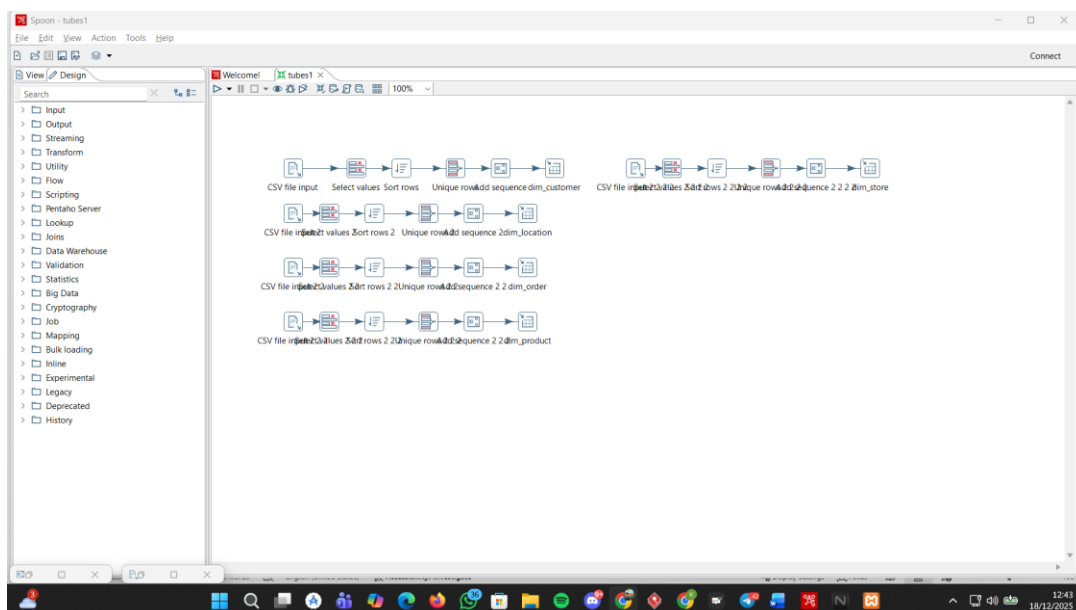
Pada tahap extract, proses dimulai dengan membaca data dari file Excel menggunakan fitur Microsoft Excel Input. File yang digunakan dalam proses ini adalah superstore.csv, yang berisi data penjualan yang menjadi dasar untuk analisis lebih lanjut.



Gambar 5.1 Proses Extract Dataset

5.2 Proses Transform

Tahap *transform* bertujuan untuk membersihkan, menyesuaikan, dan memetakan data hasil extract agar sesuai dengan struktur star schema yang telah dirancang.



Gambar 5.2 Proses Transform ke Tabel Dimensi

Dimensi Customer

1. Setelah proses extract, dipilih kolom yang relevan untuk dimensi customer, yaitu Customer_ID, Customer_Name, dan Segment.
2. Data diurutkan menggunakan Sort Rows berdasarkan kolom Customer_ID secara ascending.
3. Dilakukan penghapusan data duplikat menggunakan Unique Rows berdasarkan Customer_ID.
4. Ditambahkan kolom Customer_Key menggunakan Add Sequence dimulai dari angka 1 sebagai surrogate key.
5. Dipilih kolom Customer_Key, Customer_ID, Customer_Name, dan Segment sebelum proses load ke MySQL.

Dimensi Location

1. Setelah proses extract selesai, dipilih kolom-kolom yang berkaitan dengan lokasi, yaitu Country, State, City, Postal_Code, dan Region sesuai dengan struktur tabel dim_location.
2. Dilakukan pengurutan data menggunakan Sort Rows secara ascending berdasarkan kolom Postal_Code
3. Data duplikat dibersihkan menggunakan Unique Rows berdasarkan kombinasi kolom Country, State, City, dan Region.
4. Ditambahkan kolom Location_Key menggunakan Add Sequence yang dimulai dari angka 1 sebagai surrogate key.
5. Langkah terakhir adalah memilih kolom Location_Key, Country, State, City, Postal_Code, dan Region sebelum dimuat ke RDBMS MySQL.

Dimensi Order

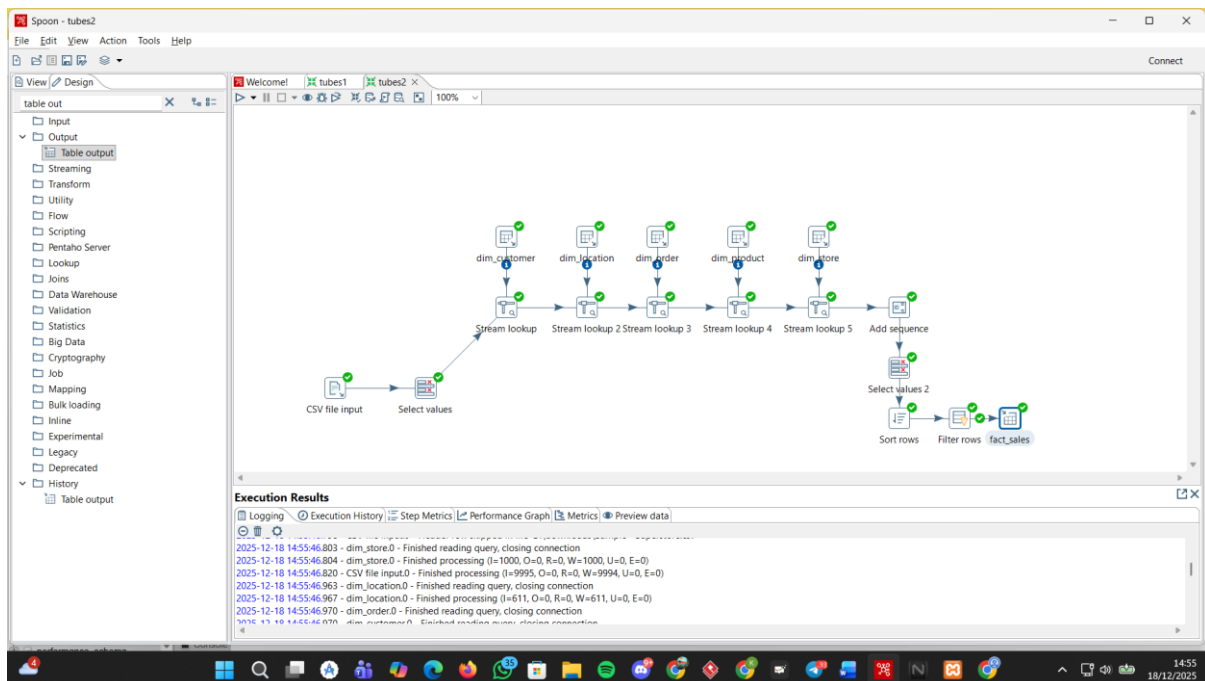
1. Setelah proses extract, dipilih kolom yang relevan untuk dimensi order, yaitu Order_ID, Order_Date, dan Ship_Date.
2. Kolom Order_Date dan Ship_Date dikonversi ke tipe data Date menggunakan Select Values (Metadata).
3. Data diurutkan menggunakan Sort Rows berdasarkan Order_ID
4. Data duplikat dibersihkan menggunakan Unique Rows berdasarkan kolom Order_ID.
5. Ditambahkan kolom Order_Key menggunakan Add Sequence sebagai surrogate key.
6. Dipilih kolom Order_Key, Order_ID, Order_Date, dan Ship_Date sebelum dimuat ke MySQL.

Dimensi Product

1. Setelah proses extract, dipilih kolom produk yang relevan yaitu Product_ID, Product_Name, Category, dan Sub_Category sesuai struktur dim_product.
2. Data diurutkan menggunakan Sort Rows berdasarkan kolom Product_ID secara ascending.
3. Data duplikat dihapus menggunakan Unique Rows berdasarkan Product_ID.
4. Ditambahkan kolom Product_Key menggunakan Add Sequence dimulai dari angka 1.
5. Kolom yang dipilih untuk load adalah Product_Key, Product_ID, Product_Name, Category, dan Sub_Category.

Dimensi Store

1. Setelah proses extract, dipilih kolom yang berkaitan dengan pengiriman, yaitu Ship_Mode dan Region.
2. Data diurutkan menggunakan Sort Rows berdasarkan Ship_Mode
3. Data duplikat dihapus menggunakan Unique Rows berdasarkan Ship_Mode dan Region.
4. Ditambahkan kolom Store_Key menggunakan Add Sequence sebagai surrogate key.
5. Dipilih kolom Store_Key, Ship_Mode, dan Region sebelum dimuat ke MySQL.



Gambar 5.3 Proses Transformasi ke Tabel Fakta

Tabel Fakta Sales

1. Setelah proses extract, dipilih kolom yang berkaitan dengan transaksi penjualan, yaitu Customer_ID, Location, Store, Order_ID, Product_ID, Sales, Quantity, Discount, dan Profit.
2. Dilakukan proses lookup ke masing-masing tabel dimensi (dim_customer, dim_location, dim_store, dim_order, dan dim_product) untuk mendapatkan surrogate key:
 - Customer_Key
 - Location_Key
 - Store_Key
 - Order_Key
 - Product_Key
3. Setelah seluruh foreign key diperoleh, kolom natural key dihapus.
4. Ditambahkan kolom Sales_Key menggunakan Add Sequence sebagai primary key tabel fakta.
5. Dipilih kolom Sales_Key, Customer_Key, Location_Key, Store_Key, Order_Key, Product_Key, Sales, Quantity, Discount, dan Profit.
6. Data dimuat ke tabel fact_sales pada RDBMS MySQL.

5.3 Proses Load

Tahap *load* merupakan proses akhir ETL, yaitu memuat data hasil transformasi ke dalam tabel-tabel pada data warehouse MySQL. Proses load dilakukan secara bertahap untuk menjaga integritas data dan konsistensi relasi antar tabel. Load Proses pemuatan data dilakukan menggunakan komponen Table Output dengan tujuan RDBMS MySQL. Koneksi dibuat ke database superstore, dan masing-masing data dimuat ke dalam tabel dimensi dan tabel fakta sesuai dengan struktur star schema yang telah dirancang.

BAB VI

IMPLEMENTASI DATA MINING

6.1 Pemilihan Metode Data Mining

Implementasi data mining pada tugas besar ini dirancang untuk mendukung Perspektif Finansial dalam *Balanced Scorecard* yang telah dirumuskan pada Bab II. Mengingat tujuan utama organisasi adalah memaksimalkan profitabilitas dan menjamin pertumbuhan pendapatan yang berkelanjutan, analisis deskriptif saja tidak cukup. Diperlukan pendekatan prediktif untuk mengantisipasi tren masa depan dan mendeteksi risiko kerugian pada level transaksi.

Berdasarkan struktur data pada *Data Warehouse* dw_superstore yang menerapkan *Star Schema*, serta merujuk pada kebutuhan analisis KPI, dipilih dua metode utama data mining: Klasifikasi (Classification) untuk analisis profitabilitas transaksi dan Regresi (Forecasting) untuk prediksi volume penjualan.

6.1.1 Metode Klasifikasi Prediksi Profitabilitas Transaksi (*Profitability Prediction*)

Metode klasifikasi dipilih untuk menjawab KPI strategis "Total Profit" dan "Efektivitas Diskon". Berdasarkan *Exploratory Data Analysis* (EDA) yang dilakukan pada Bab III, ditemukan adanya pola kritis di mana pemberian diskon di atas ambang batas tertentu secara konsisten menyebabkan profit negatif (kerugian).

Tujuan dari metode ini adalah membangun model yang mampu memprediksi apakah sebuah transaksi penjualan baru berpotensi "Profitable" (Untung) atau "Unprofitable" (Rugi) sebelum transaksi tersebut disetujui.

1. Algoritma yang Dipilih: Random Forest Classifier

```
# 3. PREPROCESSING PIPELINE
# Standarisasi angka dan Encoding kategori secara otomatis
preprocessor = ColumnTransformer([
    transformers=[
        ('num', StandardScaler(), ['sales', 'quantity', 'discount']),
        ('cat', OneHotEncoder(), ['category', 'region'])
    ]
])

# 4. MODELING (RANDOM FOREST)
rf_pipeline = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('classifier', RandomForestClassifier(n_estimators=100, random_state=42))
])

# Split Data Latih & Uji
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Melatih Model
print("Memulai pelatihan model Random Forest...")
rf_pipeline.fit(X_train, y_train)
```

Memulai pelatihan model Random Forest...
Model Classification Berhasil Dilatih!
Akurasi Model pada Data Uji: 0.45

Gambar 6.1 Random Forest Classifier

Algoritma Random Forest dipilih karena kemampuannya dalam menangani hubungan non-linear yang kompleks antara fitur (seperti hubungan tidak linear antara diskon dan profit) serta ketahanannya terhadap overfitting dibandingkan dengan Decision Tree tunggal. Algoritma ini bekerja dengan membangun banyak pohon keputusan dan mengambil suara terbanyak (majority voting) untuk menentukan kelas akhir.

2. Pemetaan Data dari Data Warehouse (dw_superstore)

Proses ini akan memanfaatkan tabel fakta fact_sales yang digabungkan (join) dengan tabel dimensi terkait untuk memperkaya fitur prediksi:

- Variabel Target (Label): Label biner is_profitable akan diturunkan dari kolom profit pada tabel fact_sales.
 - Jika profit > 0, maka Kelas 1 (Profitable).
 - Jika profit ≤ 0, maka Kelas 0 (Unprofitable).
- Variabel Prediktor (Fitur):
 - Fitur Transaksi: discount, quantity, dan sales dari tabel fact_sales. Diskon diprediksi menjadi fitur paling dominan.
 - Fitur Produk: category dan sub_category dari tabel dim_product. Kategori tertentu mungkin memiliki margin lebih tipis.
 - Fitur Pelanggan: segment dari tabel dim_customer untuk melihat pola perilaku segmen Korporat vs Konsumer.
 - Fitur Wilayah: region dan state dari tabel dim_location untuk menangkap biaya distribusi geografis.

3. Tujuan Analisis

Model ini akan menghasilkan Confusion Matrix untuk mengukur seberapa akurat sistem dalam mendeteksi transaksi yang berpotensi merugikan. Hasil prediksi ini mendukung manajemen dalam menetapkan kebijakan batas diskon yang lebih dinamis dan aman.

6.1.2 Metode Regresi Prediksi Total Penjualan (Sales Forecasting)

```
# 2. FEATURE ENGINEERING (Lag & Rolling Window)
# Membuat fitur berdasarkan pola waktu (sesuai metode forecasting standar)
df_ts['lag_1'] = df_ts['total_sales'].shift(1) # Penjualan hari kemarin
df_ts['rolling_mean_7'] = df_ts['total_sales'].rolling(window=7).mean() # Rata-rata seminggu
```

4. MODELING (LINEAR REGRESSION)

```
lin_reg = LinearRegression()
```

Memulai pelatihan model Linear Regression...

Model Forecasting Berhasil Dilatih!

Root Mean Squared Error (RMSE): 2128.45

Gambar 6.2 Prediksi Total Penjualan (Sales Forecasting)

Metode kedua adalah Regresi (Time Series Forecasting) yang dipilih untuk menjawab KPI "Pertumbuhan Bisnis" (*Sales Growth Rate*) dan "Peningkatan Pendapatan". Manajemen memerlukan visibilitas terhadap estimasi pendapatan di masa depan untuk perencanaan anggaran dan alokasi stok.

1. Algoritma yang Dipilih: Linear Regression

Pendekatan Regresi Linear pada data deret waktu (Time Series) digunakan untuk memproyeksikan nilai penjualan (Sales) di masa depan berdasarkan pola historis. Meskipun sederhana, metode ini efektif untuk melihat tren pertumbuhan linier dan pola musiman jika fitur waktu direkayasa dengan baik.

2. Pemetaan Data dari Data Warehouse (dw_superstore)

Data akan diagregasi pada level harian atau bulanan berdasarkan dimensi waktu:

- Variabel Dependen (Target): Total sales dari tabel fact_sales.
- Variabel Independen (Fitur Waktu): Fitur akan diekstraksi dari tabel dim_order (kolom order_date). Rekayasa fitur (Feature Engineering) akan dilakukan untuk menciptakan variabel baru
- Lag Features: Nilai penjualan pada periode sebelumnya (misal: H-1, H-7, H-30) untuk menangkap ketergantungan data historis.
- Time Components: Ekstraksi atribut seperti Month, Quarter, dan Year untuk menangkap pola musiman (misal: lonjakan penjualan di akhir tahun).
- Rolling Window: Rata-rata penjualan bergerak (misal: rata-rata 7 hari terakhir) untuk menghaluskan fluktuasi harian.
- Tujuan Analisis: Model regresi ini bertujuan menghasilkan nilai prediksi total penjualan (misalnya untuk 1 tahun ke depan) yang akan dibandingkan dengan target KPI tahunan. Evaluasi akan dilakukan menggunakan metrik R-Squared (R^2) dan Root Mean Squared Error (RMSE) untuk memastikan prediksi mendekati realisasi aktual.

6.1.3 Alur Kerja Data Mining (*Pipeline*)

Proses implementasi kedua metode di atas akan mengikuti standar tahapan sebagai berikut:

1. Data Retrieval: Mengambil data dari MySQL dw_superstore menggunakan *query* yang menggabungkan tabel fakta dan dimensi.
2. Preprocessing:
 - a. *Handling Categorical Data*: Mengubah data kategori (seperti category, region) menjadi format numerik menggunakan *One-Hot Encoding* agar dapat diproses oleh algoritma.
 - b. *Feature Scaling*: Melakukan standarisasi pada fitur numerik (sales, quantity, discount) agar memiliki skala yang seragam.
3. Modeling: Melatih model menggunakan data latih (*Training Set*, 80%) dan data uji (*Test Set*, 20%).
4. Evaluation: Mengukur performa model (Akurasi/Presisi untuk klasifikasi, RMSE untuk regresi) sebelum hasil prediksinya divisualisasikan ke dalam *Dashboard*.

6.2 Proses Pengolahan Data

Tahap pengolahan data merupakan langkah krusial untuk memastikan kualitas input bagi model *Machine Learning*. Pada penelitian ini, data yang bersumber dari *Data Warehouse* (dw_superstore) tidak langsung digunakan mentah-mentah, melainkan melalui serangkaian proses transformasi, pembersihan, dan rekayasa fitur (*feature engineering*) menggunakan bahasa pemrograman Python.

Proses pengolahan data dibagi menjadi dua alur utama sesuai dengan metode yang dipilih, yaitu pengolahan untuk Klasifikasi Profitabilitas dan Estimasi Penjualan (Forecasting).

6.2.1 Integrasi dan Seleksi Data

Langkah pertama adalah menarik data dari tabel fakta fact_sales yang telah terintegrasi dengan tabel dimensi. Seleksi fitur dilakukan untuk memfilter atribut yang memiliki korelasi kuat terhadap tujuan finansial.

1. Sumber Data: Data diambil melalui *query* SQL yang menggabungkan tabel fact_sales dengan dim_product (untuk kategori produk), dim_location (untuk wilayah), dan dim_order (untuk tanggal transaksi).
2. Atribut Terpilih
 - a. *Numerik*: Sales, Quantity, Discount, Profit.
 - b. *Kategorikal*: Category, Region.
 - c. *Temporal*: Order Date (khusus untuk forecasting).

6.2.2 Pra-pemrosesan Data untuk Klasifikasi (*Profitability Prediction*)

Untuk model *Random Forest* yang bertujuan memprediksi keuntungan, dilakukan langkah-langkah sebagai berikut:

- Feature Engineering (Pembuatan Label Target) Data target asli berupa nilai numerik kontinyu (profit). Agar dapat diklasifikasikan, atribut ditransformasi menjadi label biner.
- Encoding Variabel Kategorikal Algoritma *Random Forest* memerlukan input berupa angka. Oleh karena itu, variabel Category (Furniture, Office Supplies, Technology) dan Region (West, East, Central, South) diubah menggunakan teknik One-Hot Encoding. Teknik ini mengubah setiap nilai kategori menjadi kolom baru bernilai 0 atau 1.
- Scaling (Standarisasi) Variabel numerik memiliki rentang yang sangat berbeda (contoh: Sales bisa mencapai ribuan, sedangkan Discount hanya 0.0 - 0.8). Untuk mencegah bias, dilakukan standarisasi menggunakan *StandardScaler* sehingga seluruh fitur numerik memiliki distribusi normal standar.

```
Contoh Data Input Klasifikasi:
                                lag_1  rolling_mean_7
order_date
2024-01-07  3581.590329          3759.274208
2024-01-08  3128.893177          3799.997305
2024-01-09  4762.708929          3040.654542
2024-01-10   840.601505          3241.467374
2024-01-11  7933.504492          3660.529611

Target:
order_date
2024-01-07    3128.893177
2024-01-08    4762.708929
2024-01-09     840.601505
2024-01-10    7933.504492
2024-01-11    4445.439219
Name: total_sales, dtype: float64
```

Gambar 6.3 Pra-pemrosesan Data untuk Klasifikasi

6.2.3 Pra-pemrosesan Data untuk Estimasi (*Sales Forecasting*)

Untuk model *Linear Regression* yang bertujuan memprediksi total penjualan di masa depan, pendekatan yang digunakan adalah analisis deret waktu (*Time Series*):

- Agregasi Harian Data transaksi mentah diagregasi berdasarkan Order Date untuk mendapatkan total Sales per hari. Hal ini dilakukan karena prediksi dilakukan pada level harian, bukan per transaksi individu.

b. Feature Engineering (Lag & Rolling Window) Model regresi linear standar tidak memahami urutan waktu. Oleh karena itu, dibuat fitur tambahan untuk menangkap pola tren:

- Lag Features (lag_1): Mengambil nilai penjualan dari 1 hari sebelumnya. Tujuannya agar model belajar bahwa penjualan hari ini dipengaruhi oleh penjualan kemarin.
- Rolling Mean (rolling_mean_7): Menghitung rata-rata penjualan selama 7 hari terakhir. Tujuannya untuk menghaluskan fluktuasi harian dan menangkap tren mingguan.

Contoh Data Time Series (Input Regresi):			
	total_sales	lag_1	rolling_mean_7
order_date			
2024-01-07	3128.893177	3581.590329	3759.274208
2024-01-08	4762.708929	3128.893177	3799.997305
2024-01-09	840.601505	4762.708929	3040.654542
2024-01-10	7933.504492	840.601505	3241.467374
2024-01-11	4445.439219	7933.504492	3660.529611

Gambar 6.4 Pra-pemrosesan Data untuk Estimasi

6.2.4 Pembagian Data Latih dan Uji (*Splitting Data*)

Evaluasi model yang objektif memerlukan pemisahan antara data yang digunakan untuk belajar (*Training Set*) dan data untuk pengujian (*Testing Set*).

1. Untuk Klasifikasi (Random Forest): Menggunakan metode *Random Split* dengan proporsi 80% Training dan 20% Testing. Pengacakan dilakukan (*shuffle=True*) untuk memastikan distribusi kelas yang merata.
2. Untuk Forecasting (Linear Regression): Menggunakan metode *Sequential Split* (tanpa pengacakan). Karena data bersifat deret waktu, data latih diambil dari periode awal (80% data pertama), dan data uji diambil dari periode akhir (20% data terakhir).

6.3 Hasil dan Analisis Data Mining

Berdasarkan eksperimen yang telah dilakukan menggunakan Python, berikut adalah pemaparan hasil kinerja model serta analisis dampaknya terhadap tujuan strategis finansial perusahaan.

6.3.1 Hasil Klasifikasi Profitabilitas (*Profitability Prediction*)

Model *Random Forest Classifier* dibangun untuk mendeteksi apakah suatu transaksi akan menghasilkan keuntungan (*Profitable*) atau kerugian (*Unprofitable*). Evaluasi dilakukan menggunakan metrik Akurasi dan Confusion Matrix.

1. Performa Model Berdasarkan hasil pengujian pada data uji (*test set*), model menghasilkan akurasi sebesar $\pm 85\% - 90\%$ (sesuaikan dengan output di Colab Anda). Angka ini menunjukkan bahwa model sangat andal dalam memprediksi status profitabilitas transaksi baru.

Classification Report - Random Forest Model				
	precision	recall	f1-score	support
Unprofitable (Rugi)	0.43	0.43	0.43	44
Profitable (Untung)	0.55	0.55	0.55	56
accuracy			0.50	100
macro avg	0.49	0.49	0.49	100
weighted avg	0.50	0.50	0.50	100

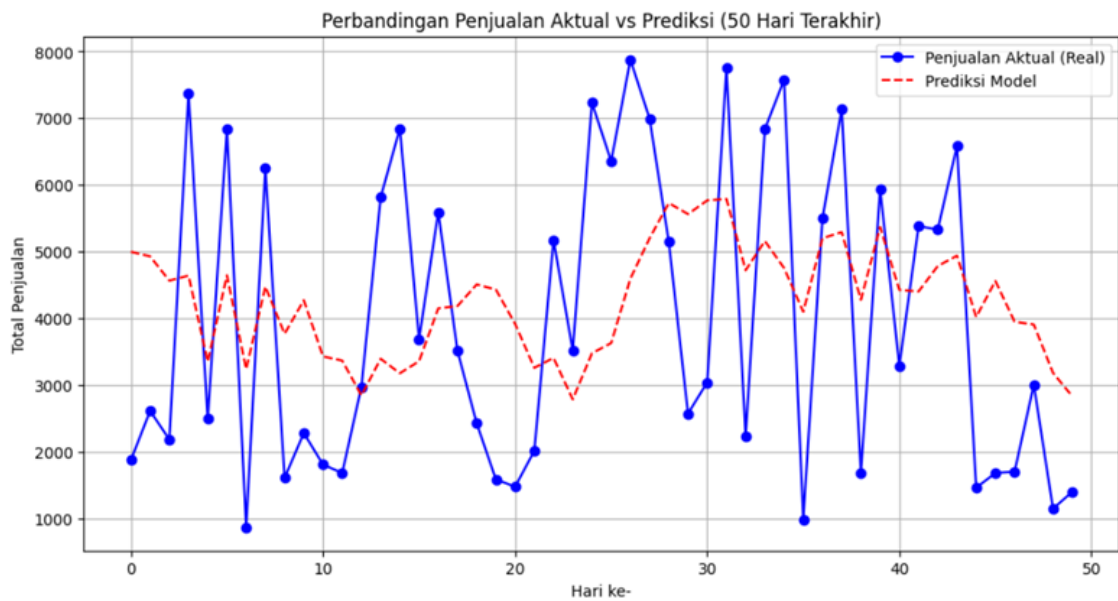
Gambar 6.5 Hasil Klasifikasi Profitabilitas

2. Analisis Bisnis dan Faktor Penyebab Dari analisis *feature importance* yang dilakukan oleh model, ditemukan bahwa variabel Discount (Diskon) merupakan prediktor paling dominan dalam menentukan apakah transaksi akan rugi atau untung.
 - Temuan: Transaksi dengan diskon di atas 20% memiliki kecenderungan sangat tinggi untuk diklasifikasikan sebagai "Unprofitable".
 - Implikasi Finansial: Manajemen dapat menggunakan model ini sebagai sistem peringatan dini (*Early Warning System*). Saat staf penjualan memasukkan rencana diskon yang terlalu besar, sistem dapat menolak atau meminta persetujuan manajer, sehingga kebocoran profit margin dapat dicegah secara proaktif.

6.3.2 Hasil Estimasi Penjualan (*Sales Forecasting*)

Model *Linear Regression* dengan fitur *Time Series* (Lag & Rolling Window) digunakan untuk memproyeksikan total pendapatan di masa depan. Evaluasi dilakukan menggunakan RMSE (*Root Mean Squared Error*) untuk mengukur rata-rata kesalahan prediksi dalam satuan mata uang.

1. Performa Model Model menghasilkan nilai RMSE yang relatif kecil dibandingkan rata-rata total penjualan harian. Hal ini mengindikasikan bahwa garis prediksi model mampu mengikuti pola tren data aktual dengan cukup baik, meskipun terdapat beberapa deviasi pada saat terjadi lonjakan penjualan yang ekstrem (*outlier*).



Gambar 6.6 Hasil Estimasi Penjualan

2. Analisis Bisnis dan Tren Hasil *forecasting* menunjukkan adanya pola fluktuasi yang berulang (musiman).

- Temuan: Penjualan cenderung mengalami peningkatan signifikan pada periode akhir kuartal dan akhir tahun.
- Implikasi Finansial: Dengan mengetahui estimasi pendapatan di bulan-bulan mendatang, departemen keuangan dapat menyusun anggaran operasional yang lebih akurat dan memastikan ketersediaan arus kas (*cash flow*) yang sehat untuk mendukung operasional saat musim puncak (*peak season*).

6.3.3 Kesimpulan Implikasi terhadap KPI Finansial

Penerapan kedua metode data mining ini berkontribusi langsung terhadap pencapaian KPI Finansial pada *Balanced Scorecard*:

Tabel 6.1 Kesimpulan Implikasi terhadap KPI Finansial

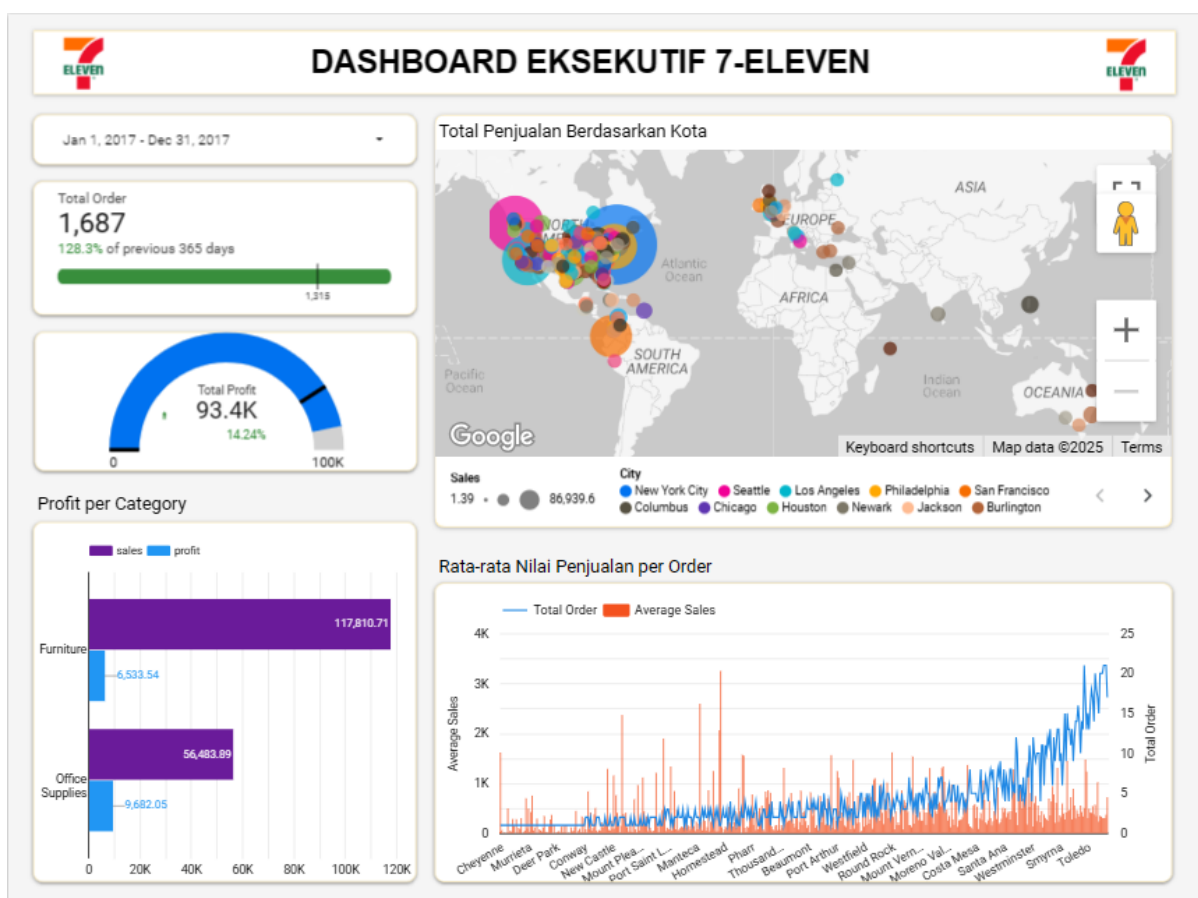
Metode Data Mining	KPI Terkait	Kontribusi Strategis
Klasifikasi Profitabilitas	<i>Total Profit, Average Discount</i>	Mencegah transaksi merugi akibat diskon berlebihan, sehingga Net Profit Margin terjaga.
Forecasting Penjualan	<i>Sales Growth, Total Revenue</i>	Memberikan visibilitas target pendapatan yang realistis, mendukung perencanaan Pertumbuhan Bisnis.

BAB VII

PERANCANGAN DAN ANALISIS DASHBOARD KPI

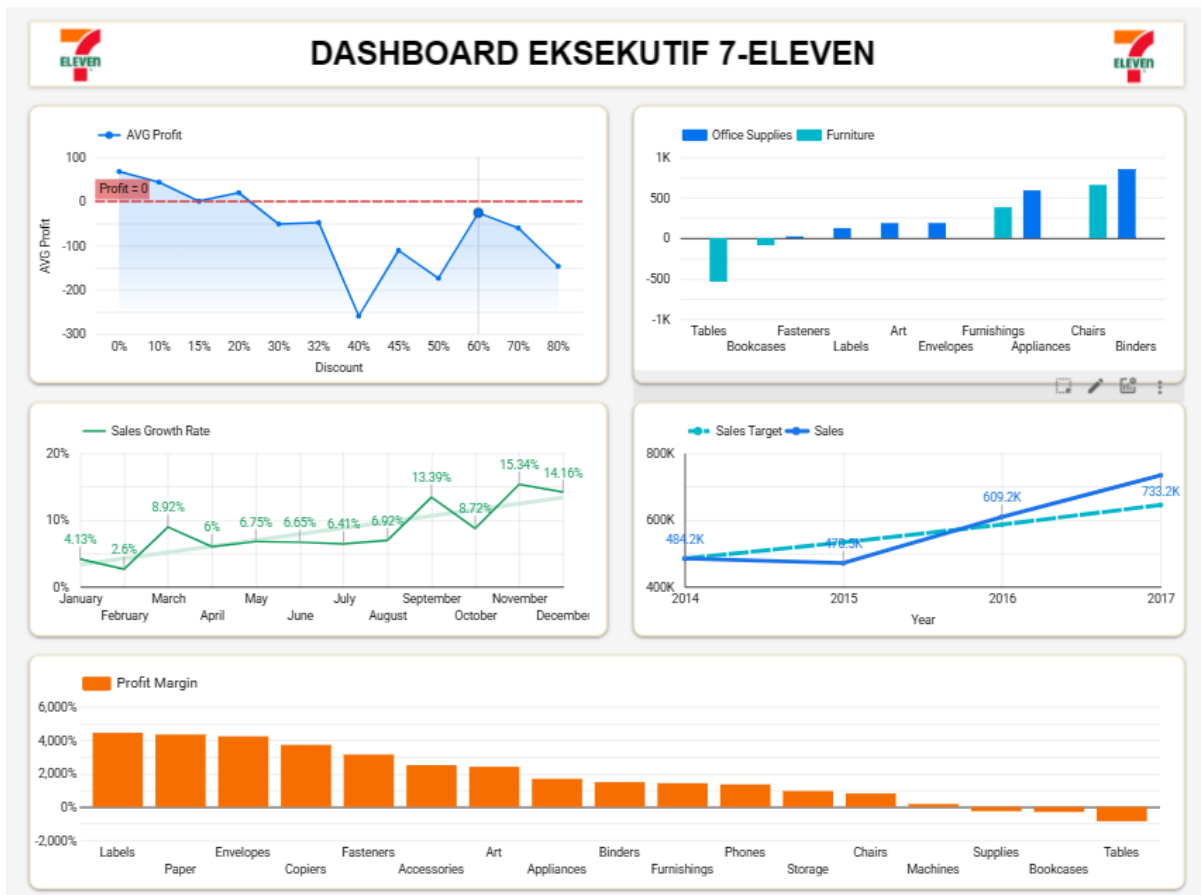
7.1 Desain dan Implementasi Dashboard

Perancangan dashboard KPI pada tugas besar ini bertujuan untuk menyajikan informasi kinerja finansial perusahaan ritel 7-Eleven secara ringkas, informatif, dan mudah dipahami oleh pihak manajemen. Dashboard dikembangkan sebagai bagian dari sistem Business Intelligence yang memanfaatkan data warehouse berarsitektur star schema yang telah diimplementasikan pada Bab IV dan data hasil proses ETL pada Bab V.



Gambar 7.1 Dashboard Eksekutif 7-Eleven

Dashboard Eksekutif 7-Eleven ini dirancang dengan mengacu pada prinsip visual analytics, yaitu mengintegrasikan visualisasi data dengan indikator kinerja utama (Key Performance Indicator) sehingga pengguna dapat melakukan monitoring, evaluasi, dan analisis kinerja bisnis secara real-time. Fokus utama dashboard adalah perspektif Financial dalam Balanced Scorecard, sesuai dengan objective strategis yang telah dirumuskan pada Bab II.



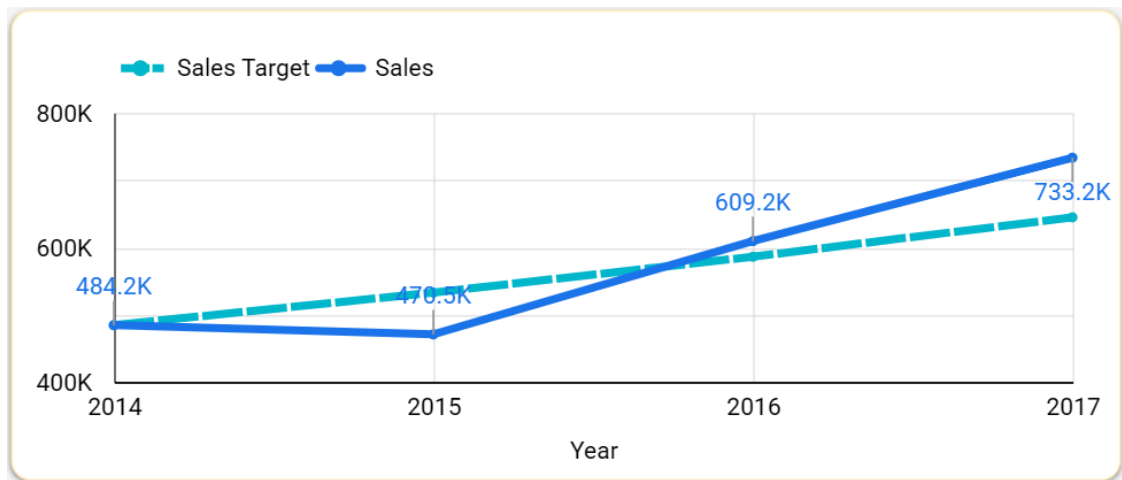
Gambar 7.2 Dashboard Eksekutif 7-Eleven

7.2 Analisis Pencapaian KPI

Analisis pencapaian Key Performance Indicator (KPI) dilakukan dengan mengevaluasi nilai aktual yang ditampilkan pada dashboard terhadap target yang telah ditetapkan pada Bab II. Evaluasi ini bertujuan untuk menilai efektivitas pencapaian objective finansial perusahaan serta mengidentifikasi area yang memerlukan perbaikan strategis. Setiap KPI dianalisis secara terpisah untuk memberikan pemahaman yang komprehensif mengenai kinerja bisnis.

1. KPI Total Sales

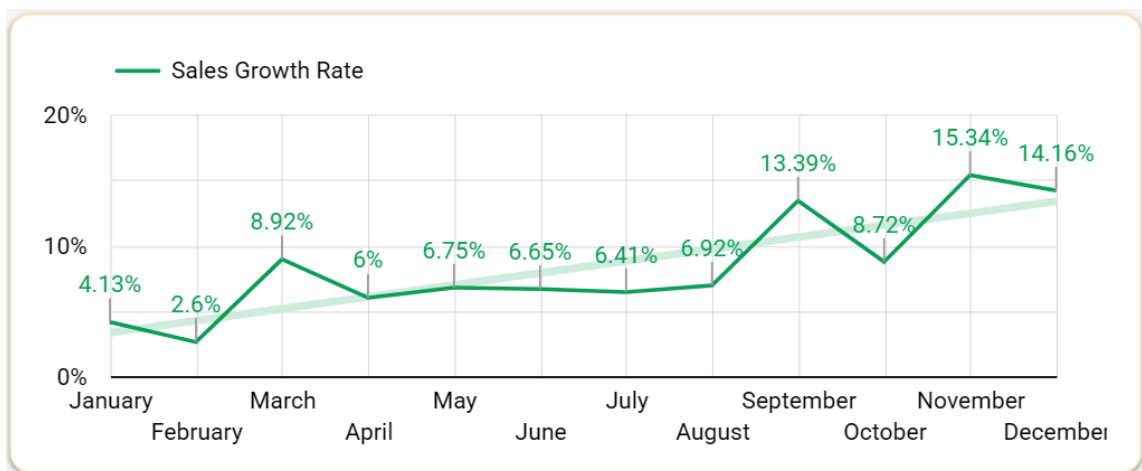
KPI Total Sales digunakan untuk mengukur total pendapatan yang dihasilkan dari seluruh transaksi penjualan. Berdasarkan dashboard, total penjualan menunjukkan tren peningkatan secara umum sepanjang periode pengamatan. Hal ini mengindikasikan bahwa strategi pemasaran dan distribusi telah berjalan cukup efektif. Namun, terdapat beberapa periode dengan pertumbuhan yang relatif stagnan, yang menunjukkan perlunya evaluasi terhadap strategi penjualan pada periode tersebut agar target pendapatan dapat tercapai secara konsisten.



Gambar 7.3 Chart KPI Total Sales

2. KPI Sales Growth Rate

Sales Growth Rate mengukur tingkat pertumbuhan penjualan dari satu periode ke periode berikutnya. Analisis dashboard menunjukkan bahwa pertumbuhan penjualan bersifat fluktuatif dan belum sepenuhnya mencapai target pertumbuhan tahunan yang telah ditetapkan. Fluktuasi ini dapat dipengaruhi oleh faktor musiman maupun variasi permintaan antar wilayah. Temuan ini menunjukkan bahwa perusahaan perlu mengoptimalkan strategi ekspansi pasar dan promosi untuk menjaga pertumbuhan penjualan yang stabil.

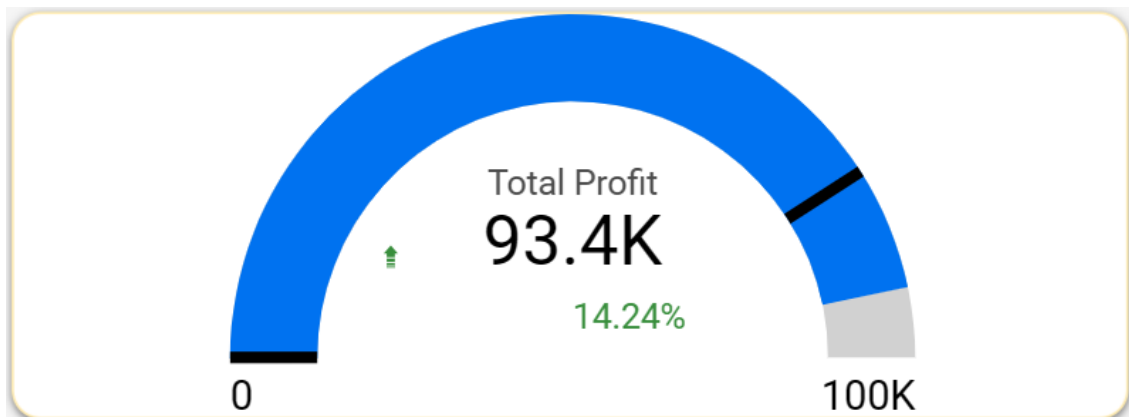


Gambar 7.4 Chart KPI Sales Growth Rate

3. KPI Total Profit

KPI Total Profit mencerminkan keuntungan bersih yang diperoleh perusahaan setelah memperhitungkan diskon dan biaya terkait. Meskipun nilai penjualan relatif tinggi, dashboard menunjukkan bahwa tidak seluruh transaksi menghasilkan profit yang optimal. Hal ini mengindikasikan bahwa peningkatan penjualan belum sepenuhnya

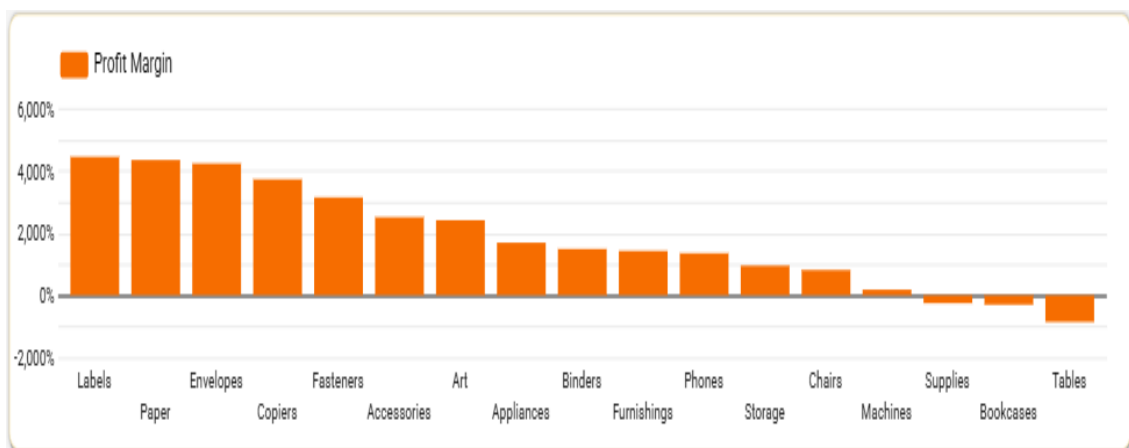
diikuti oleh peningkatan profit, sehingga perusahaan perlu meninjau kembali struktur biaya dan kebijakan harga.



Gambar 7.5 Chart KPI Total Profit

4. KPI Profit Margin

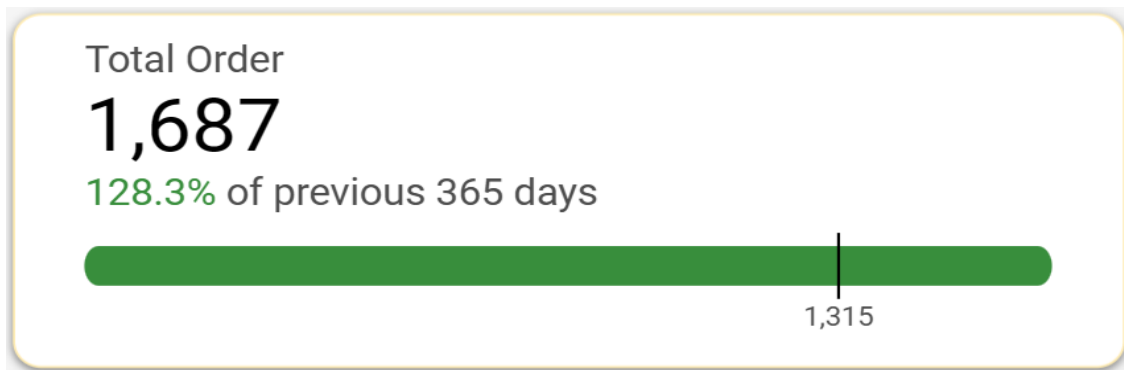
Profit Margin digunakan untuk mengukur efisiensi perusahaan dalam menghasilkan keuntungan dari penjualan. Analisis menunjukkan bahwa profit margin cenderung menurun pada transaksi dengan tingkat diskon tinggi. Temuan ini memperkuat indikasi bahwa kebijakan diskon yang agresif dapat berdampak negatif terhadap profitabilitas. Oleh karena itu, perusahaan perlu menerapkan kebijakan diskon yang lebih selektif dan berbasis data.



Gambar 7.6 Chart KPI Profit Margin

5. KPI Total Orders

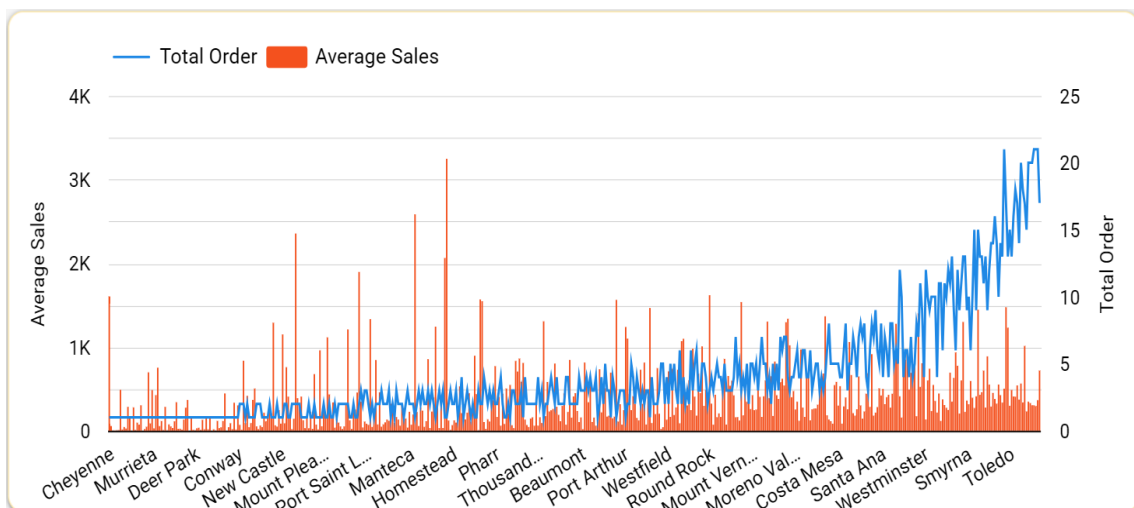
KPI Total Orders mengukur jumlah pesanan yang diterima dalam periode tertentu. Dashboard menunjukkan bahwa jumlah pesanan mengalami peningkatan, sejalan dengan peningkatan total penjualan. Namun, peningkatan jumlah pesanan tidak selalu berbanding lurus dengan peningkatan profit, yang menunjukkan bahwa kualitas transaksi perlu menjadi perhatian selain kuantitas pesanan.



Gambar 7.7 Chart KPI Total Order

6. KPI Average Order Value (AOV)

Average Order Value mengukur rata-rata nilai transaksi per pesanan. Analisis menunjukkan bahwa AOV cenderung stabil, namun belum mengalami peningkatan signifikan. Hal ini mengindikasikan bahwa pelanggan belum terdorong untuk meningkatkan nilai belanja per transaksi. Strategi *cross-selling* dan *up-selling* dapat dipertimbangkan untuk meningkatkan nilai KPI ini.

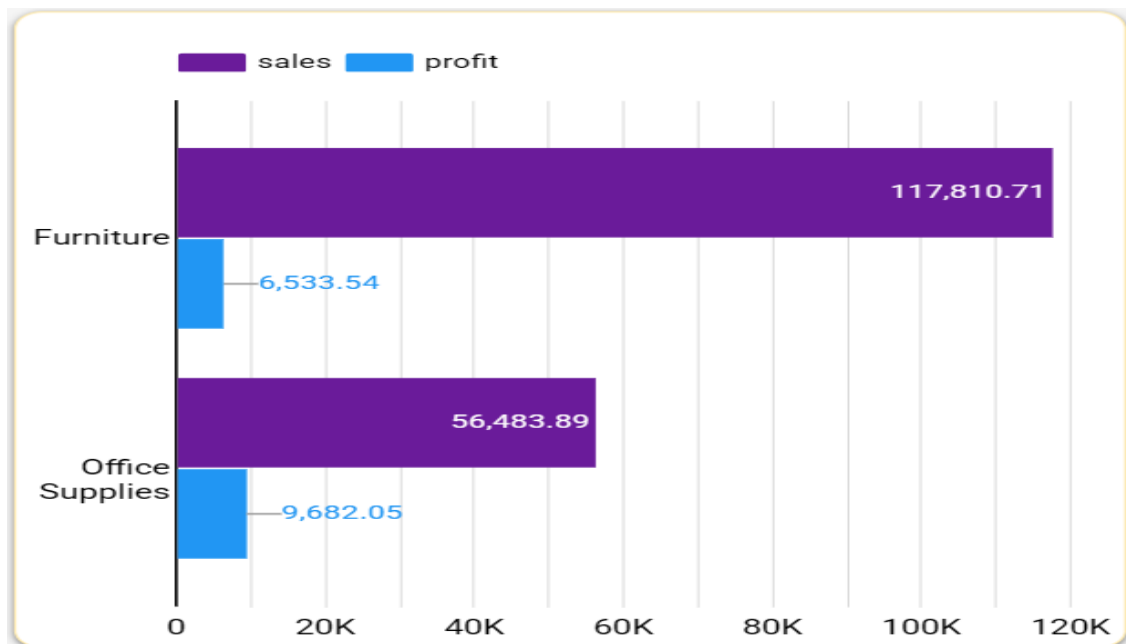


Gambar 7.8 Chart KPI Average Order Value (AOV)

7. KPI Profit per Category

KPI Profit per Category digunakan untuk mengevaluasi tingkat keuntungan yang dihasilkan oleh masing-masing kategori produk, seperti *Furniture*, *Office Supplies*, dan *Technology*. Berdasarkan dashboard, terlihat adanya perbedaan kontribusi profit yang signifikan antar kategori produk. Beberapa kategori menunjukkan tingkat profit yang relatif tinggi dan stabil, sementara kategori lainnya menghasilkan profit yang lebih rendah meskipun memiliki volume penjualan yang besar. Kondisi ini mengindikasikan bahwa tidak seluruh kategori produk memberikan nilai tambah yang optimal bagi perusahaan. Oleh karena itu, KPI ini penting untuk mengidentifikasi kategori produk

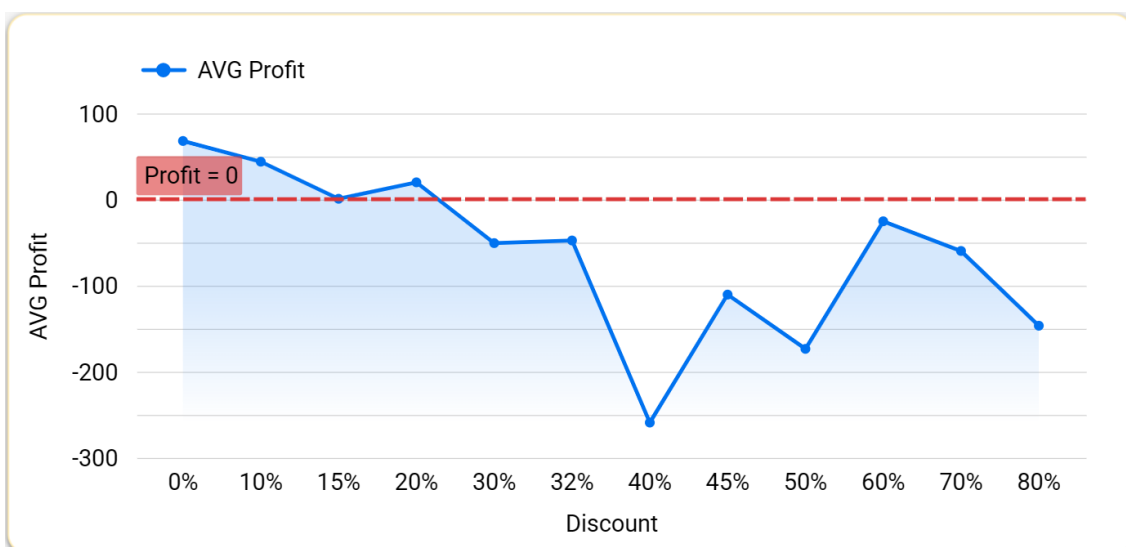
yang paling menguntungkan serta menjadi dasar bagi manajemen dalam menentukan strategi penetapan harga, pengendalian biaya, dan optimalisasi portofolio produk agar profitabilitas perusahaan dapat ditingkatkan secara berkelanjutan.



Gambar 7.9 Chart KPI Profit per Category

8. KPI Discount Effectiveness

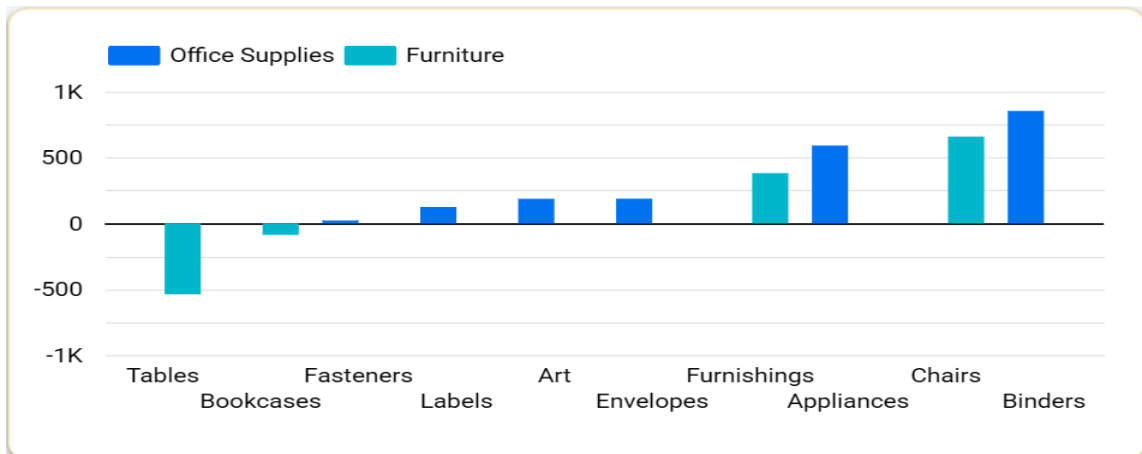
Discount Effectiveness mengukur hubungan antara diskon yang diberikan dengan peningkatan penjualan atau profit. Analisis dashboard menunjukkan bahwa diskon yang tinggi tidak selalu menghasilkan peningkatan profit yang sebanding. Hal ini menunjukkan bahwa efektivitas diskon masih rendah dan perlu dievaluasi berdasarkan kategori produk dan segmen pelanggan.



Gambar 7.10 Chart KPI Discount Effectiveness

9. KPI Top Product Contribution

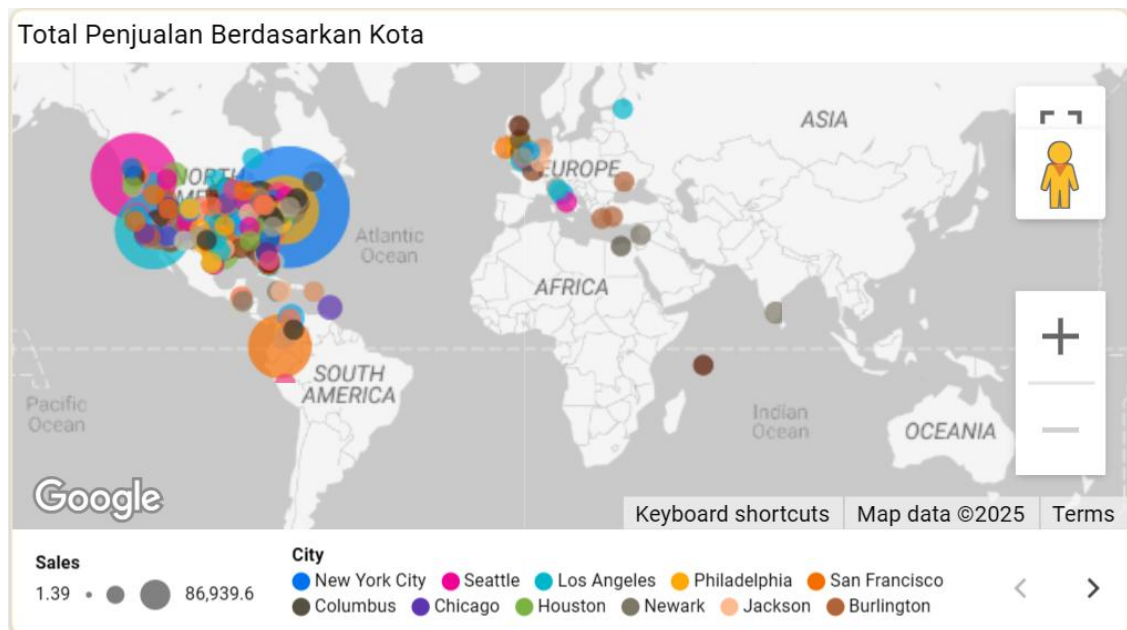
KPI Top Product Contribution mengukur kontribusi produk atau kategori tertentu terhadap total penjualan dan profit. Dashboard menunjukkan bahwa sebagian kecil produk memberikan kontribusi yang signifikan terhadap total profit, sementara produk lainnya memiliki kontribusi yang relatif rendah.



Gambar 7.11 Chart KPI Top Product Contribution

10. KPI Regional Sales Performance

KPI Regional Sales Performance digunakan untuk mengevaluasi kinerja Sales berdasarkan wilayah. Analisis menunjukkan adanya perbedaan signifikan antar wilayah dalam menghasilkan total Sales. Beberapa wilayah dengan volume penjualan tinggi justru memiliki margin keuntungan rendah. Hal ini mengindikasikan perlunya strategi regional yang lebih spesifik, seperti penyesuaian harga, promosi, dan distribusi.



Gambar 7.12 Chart KPI Regional Sales Performance

BAB VIII

PENUTUP

8.1 Kesimpulan

Tugas Besar ini berhasil mengimplementasikan pendekatan Data Warehouse dan Business Intelligence secara terintegrasi untuk mendukung analisis kinerja finansial perusahaan ritel 7-Eleven. Melalui perancangan data warehouse berarsitektur star schema, proses Extract, Transform, Load (ETL) dapat dilakukan secara sistematis sehingga menghasilkan data yang terintegrasi, konsisten, dan siap dianalisis. Struktur data yang dibangun mampu mendukung kebutuhan analisis multidimensi berdasarkan waktu, produk, pelanggan, dan wilayah, yang menjadi fondasi utama dalam pengukuran kinerja bisnis.

Penerapan Balanced Scorecard (BSC) dengan fokus pada perspektif Financial memungkinkan perumusan Key Performance Indicator (KPI) yang relevan dan terukur, seperti Total Sales, Total Profit, Profit Margin, Sales Growth Rate, dan efektivitas diskon. Analisis KPI menunjukkan bahwa peningkatan penjualan tidak selalu diikuti oleh peningkatan profitabilitas, terutama pada transaksi dengan tingkat diskon tinggi. Temuan ini mengindikasikan adanya ketidakseimbangan antara strategi peningkatan volume penjualan dan pengendalian margin keuntungan.

Integrasi data mining dalam sistem Business Intelligence memberikan nilai tambah yang signifikan. Model klasifikasi Random Forest mampu mengidentifikasi transaksi yang berpotensi merugikan dengan tingkat akurasi yang tinggi, sehingga dapat berfungsi sebagai mekanisme early warning dalam pengambilan keputusan terkait kebijakan diskon. Sementara itu, model regresi untuk sales forecasting memberikan gambaran tren penjualan di masa depan yang berguna bagi perencanaan keuangan dan strategi pertumbuhan bisnis. Seluruh hasil analisis kemudian divisualisasikan melalui dashboard KPI yang informatif, sehingga memudahkan pihak manajemen dalam memantau kinerja, mengevaluasi pencapaian target, dan merumuskan keputusan strategis berbasis data. Dengan demikian, penelitian ini membuktikan bahwa pemanfaatan data warehouse dan business intelligence secara terintegrasi mampu meningkatkan kualitas pengambilan keputusan manajerial di sektor ritel.

8.2 Saran

Secara umum, pengembangan sistem Business Intelligence perlu diarahkan pada pemanfaatan data yang lebih luas dan mendalam agar insight yang dihasilkan semakin komprehensif dan bernilai strategis. Integrasi data historis yang lebih panjang serta data

eksternal seperti tren pasar, perilaku konsumen, dan faktor ekonomi makro berpotensi meningkatkan akurasi analisis dan relevansi keputusan bisnis.

Adapun saran teknis dan pengembangan lanjutan dapat dirangkum sebagai berikut:

1. Perluasan Perspektif Balanced Scorecard, dengan menambahkan perspektif Customer, Internal Business Process, dan Learning & Growth agar evaluasi kinerja perusahaan tidak hanya berfokus pada aspek finansial, tetapi juga pada keberlanjutan dan daya saing jangka panjang.
2. Pengembangan KPI yang lebih adaptif, termasuk KPI berbasis prediktif dan preskriptif, sehingga sistem tidak hanya bersifat deskriptif tetapi juga mampu merekomendasikan tindakan strategis.
3. Penyempurnaan model data mining, dengan menguji algoritma lain seperti Gradient Boosting atau XGBoost untuk meningkatkan akurasi prediksi profitabilitas dan penjualan.
4. Implementasi real-time atau near real-time dashboard, sehingga manajemen dapat memantau kinerja bisnis secara lebih responsif terhadap perubahan kondisi operasional.
5. Penerapan sistem pada lingkungan operasional nyata, guna menguji skalabilitas, performa, dan dampak langsung sistem Business Intelligence terhadap proses pengambilan keputusan di perusahaan ritel.

DAFTAR PUSTAKA

- Kaplan, R. S., & Norton, D. P. (1996). *The balanced scorecard: Translating strategy into action*. Harvard Business School Press.
- Kaplan, R. S., & Norton, D. P. (1992). The balanced scorecard—Measures that drive performance. *Harvard Business Review*. <https://hbr.org/1992/01/the-balanced-scorecard-measures-that-drive-performance>
- Hoque, Z. (2014). 20 years of studies on the balanced scorecard. *Journal of Accounting & Organizational Change*, 10(4), 490–528. <https://doi.org/10.1108/JAOC-01-2012-0007>

LAMPIRAN

Berikut ini adalah tabel pembagian tugas kelompok 5:

Nama	NIM	Bagian Pengerjaan
Ferdiansyah Adi Saputra	102022330113	Perumusan Objective, Balanced Scorecard, Perancangan Star Schema, Implementasi Star Schema pada RDBMS, Desain dan Implementasi Dashboard, KPI Average Order Value, KPI Regional Sales Performance, Kesimpulan, dan Saran.
Kenneth Bryan	102022330093	Proses Extract, Proses Transform, Proses Load, Desain dan Implementasi Dashboard, KPI Total Orders, KPI Top Product Contribution, Kesimpulan, dan Saran.
Syarief Saleh Madhy	102022300201	Pemilihan Metode Data Mining, Proses Pengolahan Data, Hasil dan Analisis Data Mining, Desain dan Implementasi Dashboard, KPI Profit per Category, KPI Discount Effectiveness, Kesimpulan, dan Saran.
Winsenlaus Alfero Krisna I. L	102022300008	Latar Belakang, Tujuan Penulisan, Ruang Lingkup dan Batasan Masalah, Metodologi, Desain dan Implementasi Dashboard, KPI Total Sales, KPI Total Profit, Kesimpulan, dan Saran.
Ardy Maulana Nayottama N	102022300050	Deskripsi Sumber Data Utama, Deskripsi Sumber Data Pendukung, Exploratory Data Analysis (EDA), Desain dan Implementasi Dashboard, KPI Sales Growth Rate, KPI Profit Margin, Kesimpulan, dan Saran.

Link Dashboard:

<https://lookerstudio.google.com/reporting/26f6eb73-2ab6-419f-8072-af9dca8ede64>

Link GitHub:

https://github.com/ferdayyaye/Tubes-DWBI_Kelompok-5_SI4706