

INTELIJENSIA BISNIS

DAN DATA ANALITIK

Pendekatan dan metode-metode untuk Intelijensia Bisnis dan Data Analitik di bidang Teknik Industri sangat luas, sehingga referensi yang digunakan untuk memenuhi pembelajaran berasal dari berbagai referensi. Kelebihannya adalah metode-metode yang disampaikan di setiap bab mewakili beberapa metode yang berbeda-beda sesuai dengan materi yang telah disusun. Buku ini juga dilengkapi dengan contoh soal dan jawaban dan penggunaan software Python, Power Business Intelligence untuk memudahkan pemahaman pembaca terhadap keseluruhan bahasan dalam buku ini. Selain itu juga di setiap bab diberikan penjelasan langkah-langkah penggunaan rumus dan software yang lebih mudah dipahami oleh pembaca.

Pembagian bab-bab dalam buku ini antara lain: Bab I Pengertian Intelijensia Bisnis, Bab II OLAP (Online Analytical Process), Bab III Data Warehouse, Bab IV Database, Bab V Data Warehouse dengan Pentaho, Bab VI Multi Dimensional Modeling, Bab VII Data Analitik, Bab VIII Visualisasi Data, Bab IX Evaluasi Sistem Intelijensia Bisnis.



PENERBIT WAWASAN ILMU
Anggota IKAPI (215/JTE/2021)

Email : redaksi@wawasanilmu.co.id
WA : 0895354692310
FB : Penerbit Wawasan Ilmu
IG : @penerbitwawasanilmu
Twitter : @katalogwawasanilmu
Web : www.wawasanilmu.co.id

TEKNIK

Dr. Ir. Rina Fitriana, S.T., M.M., IPM. dkk.

INTELIJENSIA BISNIS DAN DATA ANALITIK



INTELIJENSIA BISNIS

DAN DATA ANALITIK

Dr. Ir. Rina Fitriana, S.T., M.M., IPM. | Dr. Dedy Sugiarto, S.Si., M.M., M.Kom.
Miwan Kurniawan Hidayat, S.Kom., M.Kom., M.T. | Yusri Eli Hotman Turnip, S.T., M.T.



Intelijensia Bisnis dan Data Analitik

Undang-Undang Republik Indonesia Nomor 28 Tahun 2014 tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Fungsi dan sifat hak cipta Pasal 4

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap:

- a. Penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;
- b. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;
- c. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan
- d. Penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

Sanksi Pelanggaran Pasal 113

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).

Dr. Ir. Rina Fitriana, S.T., M.M., IPM.
Dr. Dedy Sugiarto, S.Si., M.M., M.Kom.
Miwan Kurniawan Hidayat, S.T., M.Kom., M.T.
Yusri Eli Hotman Turnip, S.T., M.T.

Intelijensia Bisnis dan Data Analitik



Intelijensia Bisnis dan Data Analitik

Edisi Pertama
Copyright©2026
Terbit: Februari, 2026
Cetakan ke-1

Ukuran: 15,5 cm x 23 cm; Halaman: xxvi + 309

wi.2026.0779

Penulis:

- Dr. Ir. Rina Fitriana, S.T., M.M., IPM.
- Dr. Dedy Sugiarto, S.Si., M.M., M.Kom.
- Miwan Kurniawan Hidayat, S.T., M.Kom., M.T.
- Yusri Eli Hotman Turnip, S.T., M.T.

Editor Isi : Wahyu Kurniawadi
Cover : Ardyan Arya
Tata letak : Joko Abi P.

Penerbit
Wawasan Ilmu

Anggota IKAPI (215/JTE/2021)
Leler RT 002 RW 006 Desa Kaliwedi Kec. Kebasen Kab. Banyumas Jawa Tengah 53172
Email : redaksi@wawasanilmu.co.id
Web : <https://wawasanilmu.co.id/>
Telp/Pesan : 0895-3596-92310

ISBN :

All Right Reserved

Hak Distribusi/Pemasaran pada Penerbit
Hak Cipta pada Penulis
Hak Cipta Dilindungi Undang-Undang

Dilarang memperbanyak sebagian atau seluruh isi buku ini dalam bentuk apapun, baik secara elektronis maupun mekanis, termasuk memfotokopi, merekam atau dengan sistem penyimpanan lainnya, tanpa izin tertulis dari penerbit.

PRAKATA

Saat ini masih sedikit sekali buku dalam bahasa Indonesia yang membahas metode-metode Intelijensia Bisnis dan Data Analitik dengan metode yang berbeda-beda.

Pendekatan dan metode-metode untuk Intelijensia Bisnis dan Data Analitik di bidang Teknik Industri sangat luas, sehingga referensi yang digunakan untuk memenuhi pembelajaran sesuai Rencana Pembelajaran Semester berasal dari berbagai referensi. Kelebihan buku ajar ini adalah metode-metode yang disampaikan di setiap bab buku ini mewakili beberapa metode yang berbeda-beda sesuai dengan materi dalam Rencana Pembelajaran Semester yang telah disusun. Buku ini juga dilengkapi dengan contoh soal dan jawaban dan penggunaan software Python, Power Business Intelligence untuk memudahkan pemahaman pembaca terhadap keseluruhan bahasan dalam buku ini. Selain itu juga di setiap bab diberikan penjelasan langkah-langkah penggunaan rumus dan software yang lebih mudah dipahami oleh pembaca.

Buku ajar ini diharapkan dapat memudahkan dosen di dalam mengajar mata kuliah Business intelligence dan Intelijensia Bisnis dan Data Analitik. Buku ini diharapkan memudahkan mahasiswa untuk dapat mempelajari Business Intelligence, Intelijensia Bisnis dan Data Analitik. Mata kuliah ini merupakan ilmu yang mendukung industri 4.0, mencakup banyak metode yang dipelajari khususnya di Prodi Teknik Industri.

Mata kuliah pilihan Business Intelligence dan mata kuliah wajib Data Analitik pada kurikulum Teknik Industri diberikan untuk mahasiswa S1 dan mata kuliah Intelijensia Bisnis dan Data Analitik adalah mata kuliah pilihan pada Program Magister Teknik Industri. Buku ajar ini diharapkan dapat memudahkan dosen di dalam mengajar Intelijensia Bisnis dan Data Analitik dan memudahkan mahasiswa untuk dapat mempelajari Intelijensia Bisnis dan Data Analitik. Bab buku ajar ini dibagi sesuai dengan Rencana Pembelajaran Semester (RPS) yang disusun Koordinator mata kuliah Business Intelligence dan Intelijensia Bisnis dan Data

Analitik.

Pembagian bab-bab dalam buku ajar ini adalah:

BAB I. Pengertian Intelijensia Bisnis

BAB II. OLAP (Online Analytical Process)

BAB III. Data Warehouse

BAB IV. Database

BAB V. *Data Warehouse dengan Pentaho*

BAB VI. Multi Dimensional Modeling

BAB VII. Data Analitik

BAB VIII. Visualisasi Data

BAB IX. Evaluasi Sistem Intelijensia Bisnis

Dosen yang mengajar mata kuliah Intelijensia Bisnis dapat menggunakan buku ajar dengan efektif karena berbasis Capaian Pembelajaran Lulusan dan Capaian Pembelajaran Mata Kuliah.

Buku lain yang dapat digunakan sebagai sarana pendamping agar buku ini dapat lebih mudah dipahami adalah Data Mining dan Aplikasinya. Contoh kasus di industri manufaktur dan jasa.

Dengan selesainya penulisan buku ajar ini penulis mengucapkan terima kasih kepada semua pihak yang telah membantu dan mendukung, serta Penerbit yang telah membantu menerbitkan dan memasarkan buku ini.

Buku ini masih jauh dari sempurna, oleh karena itu kami menerima saran dan perbaikan untuk penyempurnaan buku ini.

Jakarta, 1 Februari 2026

Dr. Ir. Rina Fitriana, S.T., M.M., IPM.

Dr. Dedy Sugiarto, S.Si., M.M., M.Kom.

Miwan Kurniawan Hidayat, S.T., M.Kom., M.T.

Yusri Eli Hotman Turnip, S.T., M.T.

DAFTAR ISI

PRAKATA.....	v
DAFTAR ISI.....	vii
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xv

BAB I

Pengertian Intelijensia Bisnis.....	1
A. Tujuan dan Capaian Pembelajaran	1
B. Pengertian Intelijensia Bisnis	2
C. Sejarah Intelijensia Bisnis	2
D. Komponen Intelijensia Bisnis	5
E. Keuntungan Intelijensia Bisnis	8
F. Penelitian intelijensia Bisnis.....	9
G. Rangkuman.....	14
H. Soal Latihan.....	15

BAB II

OLAP (Online Analytical Process)	17
A. Tujuan dan Capaian Pembelajaran	17
B. Pengertian OLAP (<i>Online Analytical Process</i>).....	18
C. Manfaat OLAP (<i>Online Analytical Process</i>).	19
D. Pembagian OLAP (<i>Online Analytical Process</i>).	20

E. Operasi-operasi OLAP.....	20
F. Perbedaan OLAP dan OLTP (<i>Online Transactional Processing</i>)	21
G. Arsitektur OLAP	23
H. Cara Kerja OLAP	25
I. Rangkuman	27
J. Soal Latihan.....	27

BAB III

Data Warehouse.....	29
A. Tujuan dan Capaian Pembelajaran	29
B. Pengertian Data Warehouse.....	30
C. Arsitektur Data Warehouse	35
D. Data Mart.....	40
E. Rangkuman	45
F. Soal Latihan.....	46

BAB IV

Database.....	49
A. Tujuan dan Capaian Pembelajaran	49
B. Pengertian Database.....	50
C. Database Management System.....	52
D. Model Data	57
E. Pembuatan Database.....	63
F. Deskripsi Kasus	67

G. Implementasi.....	79
H. Rangkuman	89
I. Soal Latihan.....	90

BAB V

Data Warehouse Menggunakan Pentaho

A. Tujuan dan Capaian Pembelajaran	93
B. Data Warehouse Menggunakan Pentaho.....	94
C. Komponen Aplikasi Kettle	95
D. Transformasi	99
E. Cara Kerja Spoon.....	103
F. Rangkuman	156
G. Soal Latihan.....	157

BAB VI

Multi Dimensional Modeling

A. Tujuan dan Capaian Pembelajaran	161
B. Multi Dimensional Modeling	162
C. Rangkuman	184
D. Soal Latihan.....	184

BAB VII

Data Analitik.....

A. Tujuan dan Capaian Pembelajaran	185
B. Analitika Data	186

C. Klasifikasi	186
D. Clustering	197
E. Asosiasi	214
F. Support Vector Machine	217
G. Rangkuman	220
H. Soal Latihan	223

BAB VIII

Visualisasi Data.....	225
A. Tujuan dan Capaian Pembelajaran	225
B. Format Excel.....	226
C. Penggunaan Power Business Intelligence.....	232
D. Studi Kasus.....	263
E. Rangkuman	282
F. Soal Latihan.....	282

BAB IX

Evaluasi Sistem Intelijensia Bisnis	283
A. Tujuan dan Capaian Pembelajaran	283
B. Evaluasi Sistem Intelijensia Bisnis	284
C. Validasi.....	287
D. Verifikasi	296
E. Rangkuman	298
F. Soal Latihan.....	298

DAFTAR PUSTAKA..... 299

PROFIL PENULIS 307

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

DAFTAR TABEL

Tabel 1.	Perbedaan OLAP dan OLTP (Vercelis,2009)	22
Tabel 2.	Perbedaan Basis Data Operasional dan Data Warehouses (Singh, 2013).....	32
Tabel 3.	Tipe Data SQL	70
Tabel 4.	Fitur Antarmuka Kettle.....	96
Tabel 5.	Informasi Database retail_db	150
Tabel 6.	Diagram Input Output Pemodelan CRM.....	169
Tabel 7.	Transformasi data	170
Tabel 8.	Davies Bouldin Index (DBI)	174
Tabel 9.	Jumlah Elemen Cluster	175
Tabel 10.	Nilai Centroid.....	176
Tabel 11.	<i>Cluster</i> Wilayah Produksi Daging Ayam.....	177
Tabel 12.	Kategori RFM	178
Tabel 13.	Kategori RFM	180
Tabel 14.	Ranking CLV	183
Tabel 15.	Data Latih.....	190
Tabel 16.	Perhitungan Entropy dan Gain pada Node 1	191
Tabel 17.	Data Node 1.2	192
Tabel 18.	Perhitungan Entropy dan Gain pada Node 1.2	193
Tabel 19.	Data Node 1.2.2	194
Tabel 20.	Perhitungan Entropy dan Gain pada Node 1.2.2.....	194
Tabel 21.	Perbedaan metode clustering dan classification (Pengantar Machine Learning, 2018)	200
Tabel 22.	Data luas hutan yang telah mengalami <i>preprocessing</i>	208

Tabel 23.	Indeks DB untuk jumlah cluster 2-4.....	211
Tabel 24.	Tabel jarak titik centroid antara cluster 1 dan cluster 2	211
Tabel 25.	Pembagian kabupaten dengan K-Means.....	213
Tabel 26.	Tabel Data	226
Tabel 27.	Contoh Tabel yang memiliki judul kolom	226
Tabel 28.	Tidak ada merged cell.....	227
Tabel 29.	Tidak ada duplikat judul kolom.....	227
Tabel 30.	Contoh data tidak kosong	227
Tabel 31.	Perbandingan Metode	257
Tabel 32.	Persoalan saat ini dan rencana di masa depan.....	262
Tabel 33.	Sampel data harga dari PT. Food Station dalam format excel	272
Tabel 34.	Daftar Pertanyaan Kuesioner SUS.....	290
Tabel 35.	Jawaban Kuesioner SUS.....	292
Tabel 36.	Perhitungan Nilai SUS	292
Tabel 37.	Kuesioner SUS	293
Tabel 38.	Skala Interpretasi SUS	294
Tabel 39.	Jawaban Kuesioner	295
Tabel 40.	Perhitungan Nilai.....	295
Tabel 41.	Hasil uji verifikasi sistem.....	297

DAFTAR GAMBAR

Gambar 1.	Hubungan beberapa sistem informasi dengan BI (Negash, 2004).....	4
Gambar 2.	Kemajuan Teknologi BI dan Penerapannya (Chang et al., 2006).....	4
Gambar 3.	Arsitektur Business Intelligence (Sharda et al., 2018)	5
Gambar 4.	Dasar Pemahaman Tentang BI (Ranjan, 2009) ...	7
Gambar 5.	Kerangka Sistem Intelijensia Bisnis di Gudang Farmasi (Syarli,2018).....	13
Gambar 6.	Alur Sistem Intelijensia Bisnis di Gudang Farmasi (Syarli,2018).....	14
Gambar 7.	OLAP <i>Cube</i>	26
Gambar 8.	Struktur OLTP Database penjualan R Bakery....	26
Gambar 9.	Arsitektur Data Warehouse Umum.....	36
Gambar 10.	Ilustrasi Data Mart Dependent.....	42
Gambar 11.	Ilustrasi Data Mart Independent.....	43
Gambar 12.	Ilustrasi Data Mart Hybrid	43
Gambar 13.	Ilustrasi Database Relasional.....	53
Gambar 14.	Ilustrasi Entitas	58
Gambar 15.	Contoh Entitas Product	59
Gambar 16.	Contoh Relasi Sold-in	59
Gambar 17.	Relasi 1:1	60
Gambar 18.	Relasi 1:M	60

Gambar 19.	Relasi M:M	61
Gambar 20.	Notasi Crow's Foot (Edraw Content Team, 2020)	62
Gambar 21.	Notasi UML (Edraw Content Team, 2020)	62
Gambar 22.	Contoh ER Diagram.....	63
Gambar 23.	ER Diagram Sistem Penjualan.....	68
Gambar 24.	Diagram Logis Sistem Penjualan	69
Gambar 25.	Halaman Utama DrawSQL.....	72
Gambar 26.	Mengisi Nama Diagram dan Tipe Database.....	73
Gambar 27.	Jendela Antarmuka DrawSQL	73
Gambar 28.	Desain Tabel Customer di DrawSQL	74
Gambar 29.	Integritas Referensial di DrawSQL.....	75
Gambar 30.	Diagram Fisik Sistem Penjualan	76
Gambar 31.	Menu Ekspor Skema Menjadi Skrip SQL	76
Gambar 32.	Ekspor Skema Menjadi Skrip SQL.....	77
Gambar 33.	Menjalankan Server MySQL Menggunakan Paket XAMPP	79
Gambar 34.	Membuat Koneksi Server MySQL pada SQLyog	80
Gambar 35.	Jendela Antarmuka SQLyog.....	80
Gambar 36.	Menu Create Database	81
Gambar 37.	Jendela Create Database.....	81
Gambar 38.	Database Selector	82
Gambar 39.	Menu Execute SQL Script	82
Gambar 40.	Jendela Eksekusi SQL Script.....	83

Gambar 41.	Menulis Perintah SQL Insert	83
Gambar 42.	Jendela Tabel Data.....	85
Gambar 43.	Pilih Database yang Diekspor	86
Gambar 44.	Menu Backup Database.....	86
Gambar 45.	Jendela Pengaturan Ekspor.....	87
Gambar 46.	Pilih Database Tujuan.....	88
Gambar 47.	Menu Impor Database	88
Gambar 48.	Jendela Impor Database	88
Gambar 49.	Diagram Fisik Penyewaan Kendaraan.....	92
Gambar 50.	Jendela Antarmuka Kettle.....	96
Gambar 51.	Ilustrasi Transformation.....	102
Gambar 52.	Menu Hover Step	102
Gambar 53.	Ilustrasi Job.....	103
Gambar 54.	Menu Membuat Job	105
Gambar 55.	Menambahkan Step Start	105
Gambar 56.	Menambahkan Step Display Msgbox Info	105
Gambar 57.	Menu New Hop.....	106
Gambar 58.	Menu Hover New Hop	106
Gambar 59.	Hop Antara Step Start dan Step Display Msgbox Info	106
Gambar 60.	Konfigurasi Step Display Msgbox Info.....	107
Gambar 61.	Simpan File Job	107
Gambar 62.	Tombol Run	108
Gambar 63.	Jendela Run Option.....	108

Gambar 64.	Menampilkan Kotak Pesan Unconditional	109
Gambar 65.	Menambahkan Step JavaScript	109
Gambar 66.	Menggabungkan Step JavaScript.....	109
Gambar 67.	Jendela Dialog Split Hop.....	110
Gambar 68.	Step Intermediate JavaScript	110
Gambar 69.	Editor JavaScript.....	111
Gambar 70.	Mengubah Properti MsgBox.....	111
Gambar 71.	Konfigurasi Display MsgBox Info Baru	112
Gambar 72.	Job Kotak Pesan Conditional.....	112
Gambar 73.	Menu Evaluation	113
Gambar 74.	Menampilkan Kotak Pesan Conditional.....	113
Gambar 75.	Panel Execution Results	113
Gambar 76.	Sumber Data Sistem Penjualan	114
Gambar 77.	Model Dimensional Sistem Penjualan	115
Gambar 78.	Membuat Database retail_dw	116
Gambar 79.	Membuat Koneksi Database.....	117
Gambar 80.	Konfigurasi Koneksi Data Warehouse	117
Gambar 81.	Menu Share Koneksi Data Warehouse.....	118
Gambar 82.	Konfigurasi Koneksi Sumber Data	119
Gambar 83.	Step dan Hop DimCustomer	120
Gambar 84.	Konfigurasi DimCustomer Step Tabel Input.....	120
Gambar 85.	Konfigurasi DimCustomer Step Add Sequence	121
Gambar 86.	Konfigurasi DimCustomer Step Select Values...	121

Gambar 87.	Konfigurasi DimCustomer Step Table Output ..	122
Gambar 88.	SQL Create Tabel Dimcustomer	123
Gambar 89.	Transformation DimCustomer Berhasil Dijalankan	123
Gambar 90.	Data Tabel DimCustomer.....	124
Gambar 91.	Step dan Hop DimProduct	124
Gambar 92.	Konfigurasi DimProduct Step Tabel Input.....	125
Gambar 93.	Konfigurasi DimProduct Step Add Sequence....	126
Gambar 94.	Konfigurasi DimProduct Step Select Values	127
Gambar 95.	Konfigurasi DimProduct Step Table Output.....	127
Gambar 96.	SQL Create Tabel Dimproduct	128
Gambar 97.	Transformation DimProduct Berhasil Dijalankan	128
Gambar 98.	Data Tabel Dimproduct.....	129
Gambar 99.	Step dan Hop DimStoreBranch.....	129
Gambar 100.	Konfigurasi DimStoreBranch Step Tabel Input .	130
Gambar 101.	Konfigurasi DimStoreBranch Step Add Sequence	131
Gambar 102.	Konfigurasi DimStoreBranch Step Select Values.....	131
Gambar 103.	Konfigurasi DimStoreBranch Step Table Output.....	132
Gambar 104.	SQL Create Tabel Dimstorebranch	133
Gambar 105.	Transformation DimStoreBranch Berhasil Dijalankan	133
Gambar 106.	Data Tabel Dimstorebranch	134

Gambar 107.	Step dan Hop DimSaleDate.....	134
Gambar 108.	Konfigurasi DimSaleDate Step Tabel Input	135
Gambar 109.	Konfigurasi DimSaleDate Step Add Sequence ..	136
Gambar 110.	Konfigurasi DimSaleDate Step Select Values.....	137
Gambar 111.	Konfigurasi DimSaleDate Step Table Output	138
Gambar 112.	SQL Create Tabel Dimsaledate.....	139
Gambar 113.	Transformation DimSaleDate Berhasil Dijalankan	139
Gambar 114.	Data Tabel Dimsaledate.....	140
Gambar 115.	Step dan Hop FactSale.....	140
Gambar 116.	Konfigurasi FactSale Step Tabel Input	141
Gambar 117.	Konfigurasi FactSale Step Tabel Input 2	142
Gambar 118.	Konfigurasi FactSale Step Stream Lookup	142
Gambar 119.	Konfigurasi FactSale Step Tabel Input 3	143
Gambar 120.	Konfigurasi FactSale Step Stream Lookup 2	143
Gambar 121.	Konfigurasi FactSale Step Tabel Input 4	144
Gambar 122.	Konfigurasi FactSale Step Stream Lookup 3	144
Gambar 123.	Konfigurasi FactSale Step Tabel Input 5	145
Gambar 124.	Konfigurasi FactSale Step Stream Lookup 4	145
Gambar 125.	Konfigurasi FactSale Step Add Sequence	146
Gambar 126.	Konfigurasi FactSale Step Select Values.....	147
Gambar 127.	Konfigurasi FactSale Step Table Output	148
Gambar 128.	SQL Create Tabel Factsale	149
Gambar 129.	Transformation FactSale Berhasil Dijalankan....	149

Gambar 130.	Data Tabel Factsale.....	150
Gambar 131.	Step dan Hop JobSale	151
Gambar 132.	Konfigurasi JobSale Step Transformation	151
Gambar 133.	Konfigurasi JobSale Step Transformation 2	152
Gambar 134.	Konfigurasi JobSale Step Transformation 3	153
Gambar 135.	Konfigurasi JobSale Step Transformation 4	154
Gambar 136.	Konfigurasi JobSale Step Transformation 5	155
Gambar 137.	Job JobSale Berhasil Dijalankan	155
Gambar 138.	Model dimensional score peneliti.....	164
Gambar 139.	OLAP (Fitriana,2018)	166
Gambar 140.	Contoh Arsitektur Sistem Intelijensia Bisnis (Fitriana et.al.,2019).....	167
Gambar 141.	Proses <i>Data Mining</i>	172
Gambar 142.	Import Data Pada RapidMiner.....	173
Gambar 143.	Visualisasi Data Jumlah Produksi	173
Gambar 144.	Rancangan RapidMiner Studio.....	174
Gambar 145.	Grafik Jumlah Elemen Cluster	175
Gambar 146.	<i>Cube RFM</i>	178
Gambar 147.	<i>Cube RFM</i> tiga dimensi.....	179
Gambar 148.	<i>Cube RFM</i>	181
Gambar 149.	<i>Cube RFM</i> tiga dimensi.....	181
Gambar 150.	Metodologi klasifikasi	187
Gambar 151.	Struktur model Decision Tree	188
Gambar 152.	Ilustrasi pengelompokan Menggunakan Metode Classification	190

Gambar 153.	Pohon Keputusan pada Node I.....	192
Gambar 154.	Pohon keputusan Node 1.2.....	194
Gambar 155.	Pohon Keputusan.....	195
Gambar 156.	Pohon Keputusan.....	197
Gambar 157.	Proses <i>knowledge discovery</i>	198
Gambar 158.	Visualisasi Hasil Clustering Algoritme SOM dengan Cluster = 3	204
Gambar 159.	Grafik Elbow method untuk clustering pasien diabetes	205
Gambar 160.	Hasil Clustering Pasien Diabetes	205
Gambar 161.	Model analisa <i>K-Means</i> menggunakan Rapidminer Studio.....	210
Gambar 162.	Diagram luas hutan masing-masing kabupaten dan clustering yang dihasilkan	212
Gambar 163.	Peta pembagian kabupaten berdasarkan clustering yang dilakukan	213
Gambar 164.	Disain relasi antar table dalam database OLAP.....	234
Gambar 165.	Relasi antar tabel di Power BI	235
Gambar 166.	Hubungkan Power BI dengan MySQL Database	237
Gambar 167.	Tampilan Power BI.....	238
Gambar 168.	Visualisasi Clustered Column Chart.....	239
Gambar 169.	Pilih Kolom yang akan divisualisasi	239
Gambar 170.	Tab Visualisasi	240
Gambar 171.	Tambahkan data field	241
Gambar 172.	Hasil Clustered Chart.....	242

Gambar 173.	Query Database	242
Gambar 174.	Tampilan rata-rata harga di Power BI.....	243
Gambar 175.	Visualisasi Linechart.....	243
Gambar 176.	Memasukkan kolom Visualisasi	244
Gambar 177.	Filter untuk Visual Line Chart	245
Gambar 178.	Hasil Visual Line chart	245
Gambar 179.	Visualisasi pada Power BI.....	246
Gambar 180.	Fields pada Power BI.....	246
Gambar 181.	Hasil Visual Table Power BI	247
Gambar 182.	Membuka Python script.....	248
Gambar 183.	Pindahkan tab pada bagian Fields	248
Gambar 184.	Tombol Run script.....	249
Gambar 185.	Hasil Python script membuat boxplot	250
Gambar 186.	Memasukan Data ke Values	250
Gambar 187.	Kode Visual Scatter Plot Python Script Editor di Power BI.....	252
Gambar 188.	Hasil Visual Scatter Plot	252
Gambar 189.	Memasukan Data ke Values	253
Gambar 190.	Kode Visual Pie Chart Python Script Editor di Power BI	254
Gambar 191.	Hasil Visual Pie Chart.....	254
Gambar 192.	Membuka Donut chart	255
Gambar 193.	Memilih tab Fields untuk bagian Value	255
Gambar 194.	Memilih tab Fields untuk bagian Value	256
Gambar 195.	Hasil Visual Donut Chart.....	257

Gambar 196.	Improvement Implementation	258
Gambar 197.	Display PowerBI di Genba.....	259
Gambar 198.	Tampilan Power BI.....	259
Gambar 199.	Verifying the Effects.....	260
Gambar 200.	Before after improvement	261
Gambar 201.	Keuntungan dalam SQDC (Safety, Quality, Delivery, Cost)	262
Gambar 202.	<i>Dashboard Cost Power BI</i>	264
Gambar 203.	<i>Cost Material Consumable</i>	264
Gambar 204.	<i>Dashboard Cost Power BI</i>	266
Gambar 205.	<i>Cost Material Consumable</i>	267
Gambar 206.	<i>Stacked Bar Chart Dashboard</i>	268
Gambar 207.	<i>Clustered Bar Chart Dashboard</i>	269
Gambar 208.	<i>Donut Chart Dashboard</i>	270
Gambar 209.	<i>Line Chart Dashboard</i>	270
Gambar 210.	<i>Table Dashboard</i>	271
Gambar 211.	<i>Clustering Dashboard</i>	271
Gambar 212.	Diagram bintang fact_harga	273
Gambar 213.	Diagram bintang fact_rice_income.....	274
Gambar 214.	Diagram bintang fact_rice_outcome	274
Gambar 215.	Diagram bintang fact_rice_income_outcome ...	275
Gambar 216.	Kustomisasi sumber data di Google Data Studio	276
Gambar 217.	Perbandingan tampilan grafik batang di Google Data Studio (sisi kiri) dan sumber data (sisi kanan)	277

Gambar 218.	Tampilan grafik deret waktu di Google Data Studio	278
Gambar 219.	Data harga beras tanggal 4 April 2018 yang bersumber dari PT. Food Station	278
Gambar 220.	Perbandingan tren harga sebelum dan sesudah operasi pasar beras medium tanggal 22 November 2018.....	279
Gambar 221.	Diagram lingkaran daerah asal pasokan beras..	279
Gambar 222.	Dashboard penjualan.....	280
Gambar 223.	Dashboard stok – pesanan - produksi.....	281
Gambar 224.	Grafik Peringkat Persentil Nilai SUS.....	292
Gambar 225.	Grafik Peringkat Persentil Nilai SUS	294

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

BAB I

Pengertian Intelijensia Bisnis

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu menjelaskan dan mengetahui pengertian intelijensia bisnis, Sejarah intelijensia bisnis, komponen intelijensia bisnis dan keuntungan intelijensia bisnis. Setelah membaca buku ini diharapkan mahasiswa mengetahui pengertian intelijensia Bisnis, sejarah intelijensia bisnis, komponen intelijensia bisnis dan keuntungan intelijensia bisnis

Materi yang diberikan :

1. Pengertian Intelijensia Bisnis.
2. Sejarah Intelijensia Bisnis.
3. Komponen Intelijensia Bisnis.
4. Keuntungan Intelijensia Bisnis.
5. Penelitian Intelijensia Bisnis

Capaian Pembelajaran:

Capaian pembelajaran mata kuliah pada bab ini adalah mahasiswa mampu mengetahui pengertian inteligensia bisnis, sejarah inteligensia bisnis, komponen inteligensia bisnis, keuntungan inteligensia bisnis. Capaian pembelajaran pada bab ini adalah mampu menguasai prinsip dan teknik perancangan sistem terintegrasi dengan pendekatan sistem.

B. Pengertian Intelijensia Bisnis

Intelijensia Bisnis adalah kategori aplikasi dan teknologi yang luas untuk mengumpulkan, menyediakan akses, dan menganalisis data dengan tujuan membantu pengguna perusahaan membuat keputusan bisnis yang lebih baik. Integrasi data adalah Kombinasi proses teknis dan bisnis yang digunakan untuk menggabungkan data dari sumber berbeda menjadi informasi yang bermakna dan berharga. *Business Intelligence* (BI) didefinisikan sebagai model matematika dan analisa metodologi eksploitasi data yang tersedia untuk mendapatkan pengetahuan dan informasi yang bermanfaat untuk proses pengambilan keputusan yang kompleks. [Akbar,2018].

Lingkungan umum inteligensia bisnis adalah aplikasi pihak ketiga, sumber data, sistem legacy, sistem tertutup (tidak ada akses langsung), spreadsheet dan dokumen, pemformatan yang beragam.

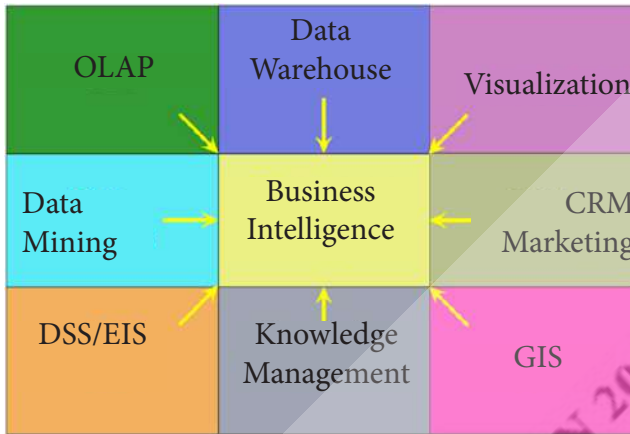
C. Sejarah Intelijensia Bisnis

Intelijensia Bisnis (*Business Intelligence*) adalah kategori aplikasi dan teknologi yang luas untuk mengumpulkan, menyediakan akses, dan menganalisis data dengan tujuan membantu pengguna perusahaan membuat keputusan bisnis yang lebih baik.

Pada tahun 1960-an, aplikasi komputer pertama dikembangkan untuk tujuan ilmiah dan pemrosesan transaksi. Laporan yang merangkum data transaksi yang diproses memberikan informasi untuk mendukung keputusan, namun sangat terbatas. Sepanjang akhir tahun 1960-an dan awal hingga pertengahan tahun 1970-an, berbagai aplikasi pendukung keputusan dikembangkan. *Decision*

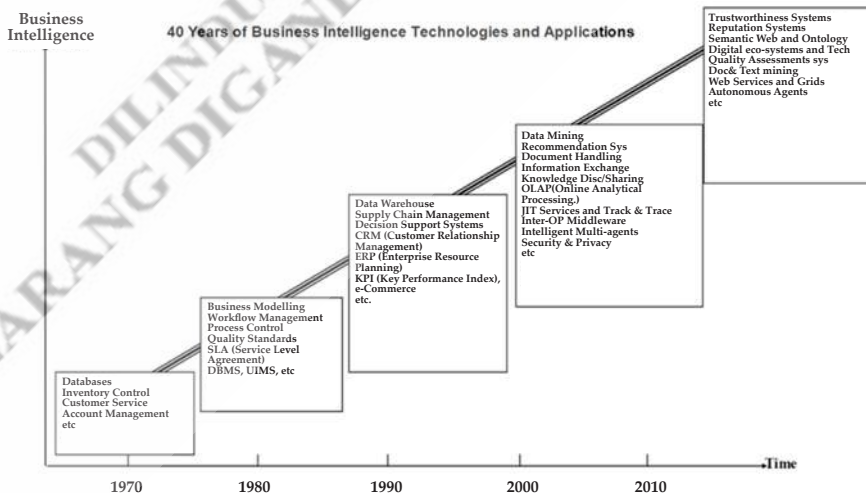
support systems (DSS) mulai digunakan untuk menggambarkan aplikasi ini dan juga menjadi nama bidang yang sedang berkembang ini. Seiring waktu, lingkungan pendukung keputusan berubah. Aplikasi tambahan bermunculan, seperti *executive information systems* (EIS), *group decision support systems* (GDSS), dan *geographic information systems* (GIS), dengan karakteristik dan nama uniknya masing-masing. *Data warehouse* menyimpan sejumlah besar data pendukung keputusan dan merupakan kunci pelaporan, *online analytical processing* (OLAP), dan aplikasi *dashboard*. Langkah cepat dibuat dengan visualisasi data. Selain data numerik, data lain juga masuk ke dalam dunia pendukung keputusan, seperti dokumen dan halaman web. Serta internet menjadi platform penyampaian DSS. Seperti yang telah dijelaskan, DSS telah berubah selama bertahun-tahun, dengan banyak jenis aplikasi baru (misalnya OLAP dan *dashboard*). Pada tahun 1989, Howard Dresner menciptakan istilah *business intelligence* (BI) yang sangat populer di industri dan digunakan untuk menggambarkan semua aplikasi pendukung keputusan. *Business intelligence* (BI) adalah aplikasi, teknologi, dan proses untuk mengumpulkan, menyimpan, mengakses, dan menganalisis data untuk membantu pengguna bisnis membuat keputusan yang lebih baik (Watson, 2009). *Business intelligence* (BI) sebagai pendukung dalam pengambilan keputusan operasional dan strategis yang dapat terhubung dengan sistem informasi lainnya, yaitu *Customer Relationship Management* (CRM), *Visualization*, *Decision Support Systems* (DSS)/*Executive Information Systems* (EIS), *Data Warehouse*, *Knowledge Management*, *Data Mining*, *On-Line Analytical Processing* (OLAP), *Geographic Information Systems* (GIS) (Rehman et al., 2022). Gambar 1 mengilustrasikan keterkaitan beberapa sistem informasi sebagai pendukung BI.

Business Intelligence dapat menjadi bagian dari bagian CRM dan pemasaran dan menjadi solusi untuk memenuhi kebutuhan perusahaan yang tidak memiliki bagian jasa keuangan (Liu L.,2010)



Gambar 1. Hubungan beberapa sistem informasi dengan BI (Negash, 2004)

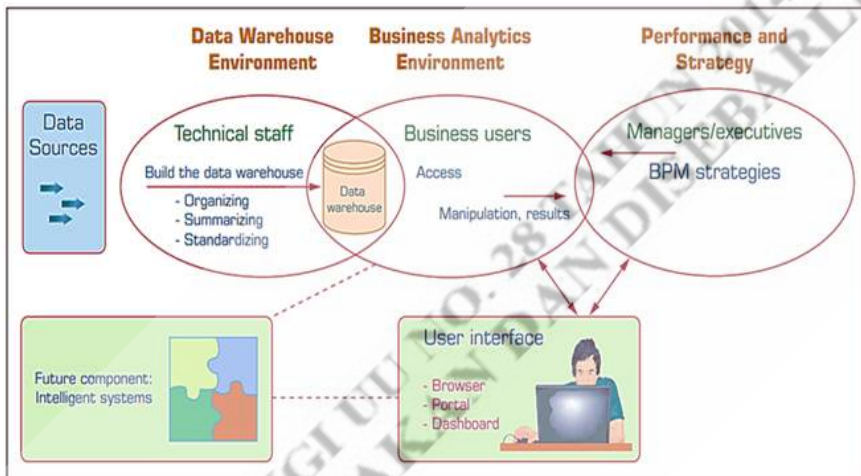
Thomsen (2003) mengemukakan bahwa Terminologi *business intelligence* (BI) telah mengambil alih peran yang sebelumnya ditempati oleh sistem pendukung keputusan (*decision support systems*), sistem informasi eksekutif (*executive information systems*), dan sistem informasi manajemen (*management information systems*) (Siswono, 2013). Gambar 2 menunjukkan bahwa konsep *business intelligence* (BI) telah mengalami perkembangan selama empat dekade terakhir dan akan terus berkembang di masa depan.



Gambar 2. Kemajuan Teknologi BI dan Penerapannya (Chang et al., 2006)

D. Komponen Intelijensia Bisnis

Sistem *business intelligence* (BI) terdiri dari empat bagian utama yaitu sumber data dan *data warehouse*; *business analytics*, alat untuk mengolah, menambang, dan menganalisis data di *data warehouse*; *business performance management* (BPM) untuk pemantauan dan analisis kemampuan kerja; serta antarmuka pemakai (misalnya *data visualization*) (Sharda et al., 2018). Hubungan antar komponen BI diilustrasikan pada Gambar 3.



Gambar 3. Arsitektur Business Intelligence (Sharda et al., 2018) - Sumber Data dan *Data Warehouse*

Sumber data (*data sources*) dapat berupa basis data operasional, data historis, data eksternal misalnya dari perusahaan riset pasar atau dari internet, atau informasi dari lingkungan gudang data yang sudah ada. Sumber data dapat berupa basis data relasional atau struktur data lain yang mendukung lini aplikasi bisnis. Sumber data juga dapat berada di berbagai platform dan dapat berisi informasi terstruktur, seperti tabel atau *spreadsheet*, atau informasi tidak terstruktur, seperti file teks biasa atau gambar dan informasi multimedia lainnya. *Data warehouse* adalah suatu sistem penyimpanan data yang digunakan dalam bidang *business intelligence* (BI). *Data warehouse* adalah tempat di mana data dari berbagai sumber yang berbeda dikumpulkan, disimpan, dan

diorganisir secara terstruktur agar dapat diakses dan dianalisis dengan lebih efisien. *Data Warehouse* memungkinkan perusahaan untuk menggabungkan data operasional dari berbagai departemen atau sistem internal, serta data eksternal, menjadi satu sumber data yang terpusat. Dengan demikian, memfasilitasi analisis yang mendalam, pelaporan yang lebih baik, dan pengambilan keputusan yang lebih terinformasi bagi para pemangku kepentingan bisnis.

1. *Business Analytics*

Business analytics dalam konteks *business intelligence* (BI) merujuk pada proses analisis data untuk mendapatkan wawasan yang mendalam tentang kinerja bisnis, tren pasar, dan peluang potensial. Ini melibatkan penggunaan teknik statistik, pemodelan prediktif, dan analisis data lainnya untuk mengidentifikasi pola, tren, dan hubungan dalam data bisnis. *Business analytics* memungkinkan perusahaan untuk membuat keputusan yang lebih baik, memahami pelanggan dengan lebih baik, meningkatkan efisiensi operasional, dan mengoptimalkan strategi bisnis berdasarkan pemahaman yang didapat dari data.

2. *Business Performance Management* (BPM)

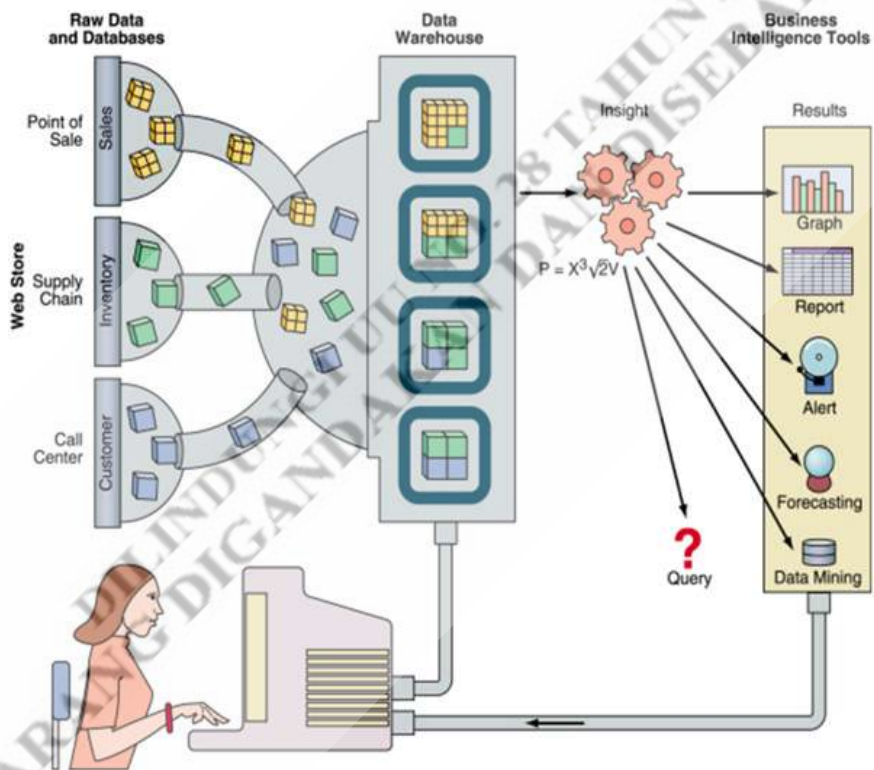
BPM adalah pendekatan sistematis yang digunakan oleh perusahaan untuk mengukur, mengelola, dan meningkatkan kinerja bisnis dengan memanfaatkan berbagai metrik dan KPI (*Key Performance Indicators*) untuk evaluasi kinerja, serta melibatkan proses perencanaan strategis, penetapan target, pelaporan kinerja, dan tindakan perbaikan berkelanjutan dengan tujuan mencapai hasil bisnis yang optimal.

3. Antarmuka Pemakai (*user interface*)

User interface (UI) dalam *business intelligence* (BI) adalah tampilan grafis atau antarmuka yang digunakan oleh pengguna untuk berinteraksi dengan data dan fungsionalitas BI. UI bertujuan untuk menyajikan informasi yang relevan dan mudah dimengerti kepada pengguna, sehingga mereka dapat dengan mudah menjelajahi data, membuat laporan, dan melakukan analisis tanpa memerlukan pengetahuan teknis yang mendalam. UI dalam BI sering kali mencakup *dashboard*

interaktif, laporan yang dapat disesuaikan, fitur navigasi yang intuitif, dan alat visualisasi data untuk membantu pengguna memahami dan menginterpretasikan informasi dengan lebih baik.

Gambar 4 menyajikan pemahaman tentang BI. Gambar tersebut juga mengungkapkan bagaimana data dari sumber yang berbeda dapat diekstraksi dan disimpan untuk diambil dan analisis. Fungsi dan laporan BI dasar ditunjukkan pada Gambar 4.



Gambar 4. Dasar Pemahaman Tentang BI (Ranjan, 2009)

E. Keuntungan Intelijensia Bisnis

Manfaat dari penerapan sistem intelijensia bisnis adalah berkontribusi pada efektivitas dan efisiensi operasional perusahaan serta membantu dalam mencapai tujuan bisnis yang lebih baik. Penerapan sistem intelijensia bisnis dapat meningkatkan efisiensi dalam proses pengambilan keputusan. Berikut ini adalah manfaat penggunaan sistem intelijensia bisnis:

1. Mengakses dan menganalisis laporan secara mandiri tanpa bergantung pada pihak lain.
2. Mengidentifikasi area di sistem yang mungkin mengalami pemborosan sumber daya, sehingga tindakan perbaikan dapat diambil.
3. Membantu mengungkap kekuatan dan kelemahan perusahaan melalui analisis data yang mendalam.
4. Dengan akses ke informasi *real-time* dan visualisasi data yang baik, pengambilan keputusan dapat dilakukan lebih efisien.
5. Memungkinkan analisis data secara *real-time* dengan kemampuan navigasi yang cepat.
6. Memfasilitasi berbagi informasi antara pengguna dan memudahkan akses ke data yang relevan.
7. Menyediakan visualisasi data dalam bentuk grafik, tabel, dan *dashboard* yang memudahkan interpretasi dan pengambilan keputusan.

Perusahaan-perusahaan telah menyadari pentingnya intelijen bisnis bagi banyak orang, diantaranya yaitu (Ranjan, 2009):

1. Dengan alat BI yang unggul, karyawan dapat dengan mudah mengubah pengetahuan bisnis melalui kecerdasan analitis untuk menyelesaikan banyak masalah bisnis.
2. Dengan BI, perusahaan dapat mengidentifikasi pelanggan yang paling menguntungkan dan alasan yang mendasari loyalitas pelanggan tersebut, serta mengidentifikasi pelanggan masa depan yang potensial.

3. Analisis data *click-stream* untuk meningkatkan strategi *e-commerce*.
4. Deteksi *warranty-reported problems* dengan cepat untuk meminimalkan dampak kekurangan desain produk.
5. Menemukan kegiatan kriminal pencucian uang.
6. Menganalisis potensi pertumbuhan profitabilitas pelanggan dan mengurangi risiko melalui penilaian kredit keuangan yang lebih akurat dari pelanggan.
7. Menentukan kombinasi produk dan lini layanan apa yang cenderung dibeli pelanggan dan kapan waktu pembeliannya.
8. Menetapkan tarif yang lebih menguntungkan untuk premi asuransi.
9. Mengurangi waktu henti peralatan dengan menerapkan pemeliharaan prediktif.
10. Menggunakan *attrition and churn analysis* untuk memahami dan mengelola tingkat kehilangan pelanggan, karena pelanggan berpindah menjadi pelanggan perusahaan lain (pesaing).
11. Mendeteksi dan mencegah perilaku penipuan, seperti dari lonjakan penggunaan saat kartu kredit atau telepon yang dicuri.

F. Penelitian intelijensia Bisnis

Beberapa penelitian intelijensia bisnis yang paling populer adalah pendekatan tunggal sistem intelijensia bisnis, membahas tentang teori, metode, model, arsitektur, tools, sistem dan studi kasus penerapan intelijensia bisnis (Fitriana et.al.,2011). Intelijensia bisnis dari suara pelanggan, *Even Driven Architecture*, arsitektur berorientasi proses dan berorientasi layanan. Ada banyak perangkat lunak yang digunakan dalam penelitian Sistem Intelijen Bisnis seperti SharePoint Server, Microsoft SQL Server, Power Business Intelligence Microsoft dan produk BI, dan terakhir, menjelaskan cara memberikan solusi BI dengan BI stack.

Beberapa manfaat penggunaan *business intelligence*

adalah membantu melihat laporan secara mandiri; membantu mengidentifikasi pemborosan dalam sistem; membantu mengidentifikasi kekuatan dan kelemahan perusahaan; meningkatkan proses pengambilan keputusan; mengaktifkan analisis secara *real-time* dengan navigasi cepat; memudahkan berbagi dan mengakses informasi; memberikan jawaban yang sangat cepat untuk pertanyaan dan masalah bisnis; menyediakan visualisasi data sehingga sangat mudah dibaca, dipahami, dan ditafsirkan (M. Joshi and A. Dubbawar, 2021)

1. Integrasi antara sistem intelijensia bisnis dan sistem rantai pasok

Intelijensia bisnis, teknologi dasar intelijensia bisnis, dan isi dari rantai pasok terintegrasi dan berfokus pada analisis penerapan intelijensia bisnis dalam rantai pasok terintegrasi untuk memberikan dasar bagi perusahaan untuk mengimplementasikan intelijensia bisnis (Luhao, 2010). Intelijensia bisnis rantai pasok memperkenalkan kekuatan pendorong untuk penerapannya dan menjelaskan arsitektur BI rantai pasokan. Sistem pengukuran kinerja rantai pasokan global berdasarkan model referensi proses dijelaskan. Teknologi mutakhir utama seperti arsitektur berorientasi layanan, pemantauan aktivitas bisnis, portal web, penambangan data, dan perannya dalam sistem BI juga dibahas. Terakhir, tren dan teknologi utama BI yang akan mempengaruhi sistem masa depan akan dijelaskan. (Stefanovic, N. and Stefanovic, D, 2009)

2. Integrasi antara intelijensia bisnis dan sistem manajemen hubungan pelanggan

Sistem manajemen hubungan pelanggan dan intelijensia bisnis memberikan pendekatan holistik kepada pelanggan yang mencakup perbaikan dalam profil pelanggan, deteksi nilai pelanggan yang lebih sederhana, pengukuran keberhasilan perusahaan dalam memuaskan pelanggannya, dan menciptakan manajemen hubungan pelanggan yang komprehensif (Habul, A, 2010). Infrastruktur konseptual dan teknologi diusulkan dan diintegrasikan ke dalam sistem manajemen hubungan pelanggan yang terkait dengan konsep dan teknologi intelijensia bisnis

yang digunakan untuk memperoleh pengetahuan tentang siswa dan untuk mendukung proses pengambilan keputusan (Piedade, M.B *et. al.*,2010). Dalam studi mendalam terhadap organisasi di Amerika Utara dan Eropa, lebih dari 60% organisasi yang disurvei IDC mengatakan mereka akan membelanjakan sebagian anggarannya untuk intelijensia bisnis dalam 12 bulan ke depan. Intelijensia bisnis dapat mengambil manfaat dari manajemen hubungan pelanggan dalam hal pemasaran dan memperluas solusi untuk memenuhi kebutuhan perusahaan yang tidak memiliki banyak uang di industri jasa keuangan (Luhao L., 2010). Intelijen bisnis elektronik bertujuan untuk mengembangkan spektrum peluang bisnis yang luar biasa dan penerapan intelijen bisnis oleh pengguna sangatlah penting dan proposisi yang relevan dibuat (Yujun B. *et al.* 2009)

3. Integrated between Intelijensia Bisnis dan Data Mining

Metodologi penambangan data yang disebut Business Intelligencedriven Data Mining (BIdDM) menggabungkan penambangan data berbasis pengetahuan dan penambangan data berbasis metode, dan mengisi kesenjangan antara pengetahuan intelijen bisnis dan berbagai metode penambangan data yang ada di e-Business. BIdDM berisi dua proses: proses konstruksi kerangka empat lapis dan proses penambangan data. Sebuah metodologi ditetapkan dalam menyiapkan kerangka empat lapis, yang merupakan bagian penting dalam BIdDM. Studi kasus e-Shop B2C disediakan untuk menggambarkan penggunaan BidDM Conferences (Zhang H. 2009). Intelijen bisnis adalah informasi tentang kinerja masa lalu perusahaan yang digunakan untuk membantu memprediksi kinerja perusahaan di masa depan. Hal ini dapat mengungkapkan tren-tren yang muncul yang dapat memberikan keuntungan bagi perusahaan. Penambangan data memungkinkan pengguna menyaring sejumlah besar informasi yang tersedia di gudang data; dari proses penyaringan inilah permata intelijen bisnis dapat ditemukan (Pillai, Jyothi.2011). Penjelajah intelijen bisnis apakah mengoptimalkan hasil pencarian atau tidak, tulisan ini memilih tiga objek penelitian, Google, Quintura, Clusty, dan melakukan analisis varians dalam hal efisiensi, efektivitas dan

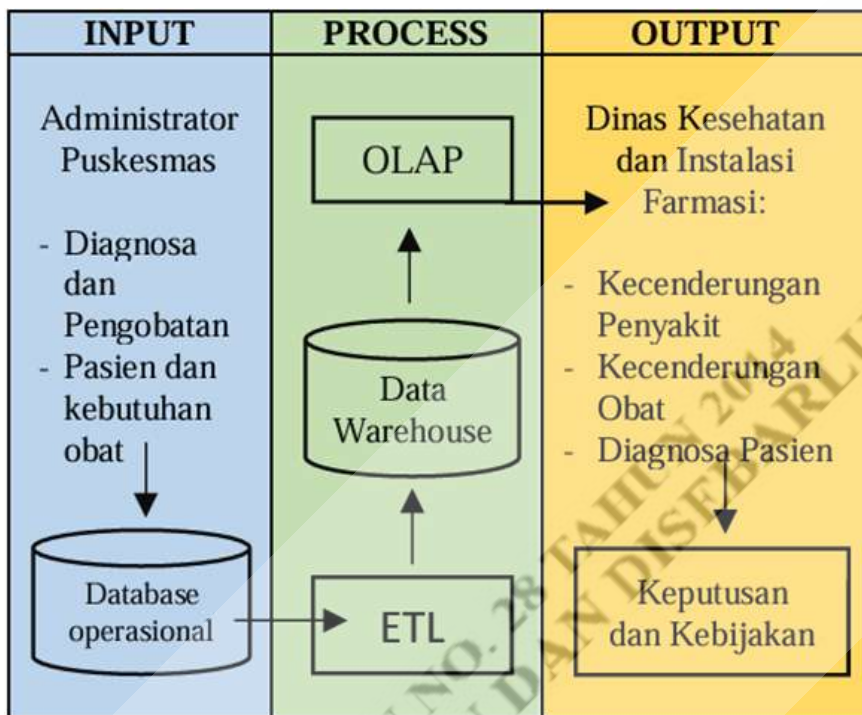
kegunaan. Hasilnya menunjukkan bahwa teknik visualisasi dan pengelompokan menawarkan implikasi praktis bagi pengguna mesin pencari (Jiang, 2009).

4. Integrasi Intelijensia Bisnis dan Artificial Intelligence

Penerapan intelijensia bisnis pada jaringan saraf tiruan dalam menganalisis heterogenitas konsumen dalam konteks perilaku makan di luar di Taiwan. Kumpulan data untuk penelitian ini dikumpulkan melalui survei terhadap 800 konsumen Taiwan. Hasil analisis data menunjukkan bahwa algoritma ekstraksi aturan jaringan saraf mampu menemukan segmen konsumen yang berbeda dan memprediksi konsumen dalam setiap segmen dengan akurasi yang baik (Yoichi, 2010). Pendekatan hybrid fuzzy-Delphi-AHP untuk mengusulkan kerangka kerja yang lebih komprehensif dengan elemen bisnis tertentu, dan juga menunjukkan enam indeks kinerja bagi perusahaan untuk menyesuaikan strategi bisnis. Untuk mengurangi risiko bisnis dalam mengembangkan pasar internasional, penggunaan model aliansi merupakan strategi utama bagi perusahaan layanan informasi. Di sisi lain, perusahaan harus menangani informasi bisnis yang lebih akurat untuk mendukung kecerdasan bisnis mereka.

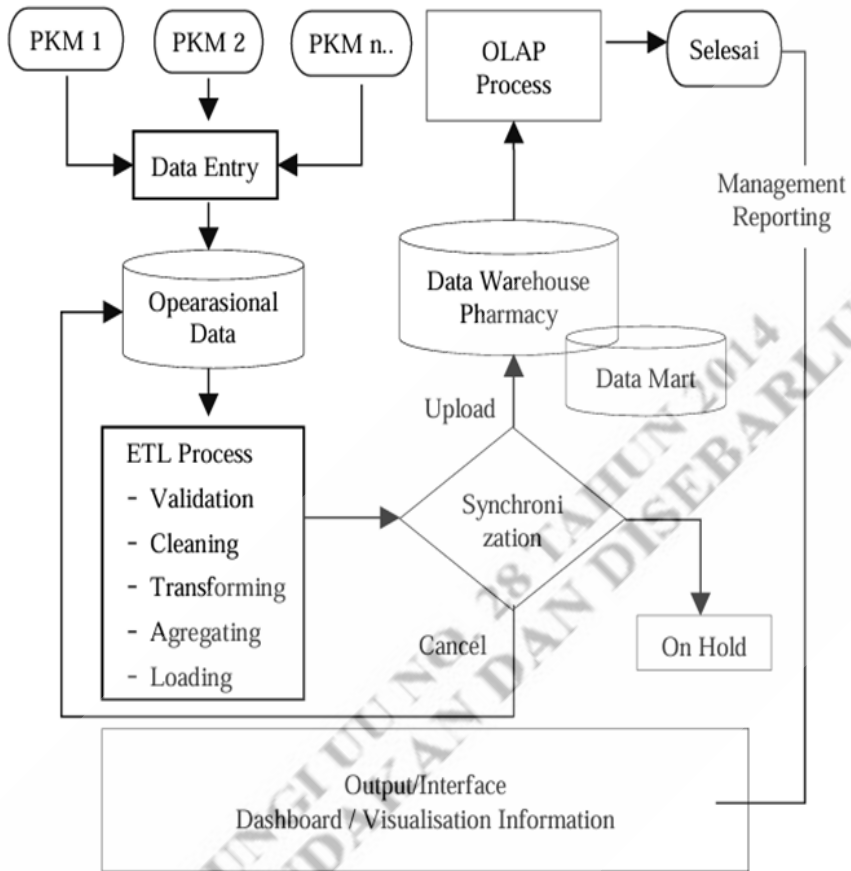
5. Implementation Intelijensia Bisnis

Strategi implementasi intelijensia bisnis bisa menggunakan implementasi sumber daya eksternal dan pengembangan atau outsourcing. Atau bisa menggunakan implementasi sumber daya internal dan pengembangan intelijen bisnis (in-house). Sebagian besar perangkat lunak Intelijensia Bisnis diberikan sebagai solusi lengkap, meskipun keragaman fungsi yang luas tidak memungkinkan siapa pun menjadi pemimpin dalam semua hal.



Gambar 5. Kerangka Sistem Intelijensia Bisnis di Gudang Farmasi (Syarli,2018)

Sistem yang dibangun menyediakan informasi mengenai fluktuasi permintaan obat pada 11 puskesmas dan kecenderungan penyakit. Administrasi puskesmas membuat database operasional mengenai diagnosa dan pengobatan pasien dan kebutuhan obat. Pada tahap proses dilakukan Ekstrak Transform Load dan pembuatan Data Warehouse dan Olap. Pada tahap output Dinas Kesehatan dan Instalasi Farmasi mengambil keputusan dan kebijakan berdasarkan kecenderungan penyakit, kecenderungan obat, diagnosa pasien.



Gambar 6. Alur Sistem Intelijensia Bisnis di Gudang Farmasi (Syarli,2018)

G. Rangkuman

Informasi dalam suatu perusahaan disimpan dalam berbagai format, termasuk sistem lama dan file teks/spreadsheet/dokumen. Intelijensia bisnis akan dapat mengekstraksi dan mengintegrasikan informasi ke dalam gudang data yang memungkinkan analisis dan penggunaan lebih lanjut seperti mendukung proses pengambilan keputusan eksekutif. Memilih platform BI itu sulit, dalam beberapa kasus praktik terbaiknya adalah memadupadankan berbagai produk untuk memanfaatkan potensinya secara maksimal. Keterbatasan waktu dan anggaran antara lain akan menentukan apakah yang

terbaik adalah memilih metode in-house atau outsourcing dalam mengimplementasikan Business Intelligence.

H. Soal Latihan

1. Beri penjelasan contoh Intelijensia Bisnis di industri manufaktur.
2. Jelaskan keuntungan Business Intelligence.
3. Jelaskan komponen Business Intelligence.

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

BAB II

OLAP (*Online Analytical Process*)

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu menguasai konsep teoretis dan aplikasi OLAP (*Online Analytical Process*). Setelah membaca buku ini diharapkan mahasiswa mampu menguasai pengertian, manfaat, pembagian dan operasi operasi OLAP (*Online Analytical Process*).

Materi yang diberikan :

1. Pengertian OLAP (*Online Analytical Process*).
2. Manfaat OLAP (*Online Analytical Process*).
3. Pembagian OLAP (*Online Analytical Process*).
4. Operasi-operasi OLAP.
5. Perbedaan OLAP dan OLTP (*Online Transactional Processing*).

6. Arsitektur OLAP.
7. Cara Kerja OLAP.

Capaian Pembelajaran:

Capaian pembelajaran mata kuliah pada bab ini adalah mahasiswa mampu menguasai konsep teoretis dan aplikasi OLAP (*Online Analytical Process*). Capaian pembelajaran pada bab ini adalah mampu menguasai prinsip dan teknik perancangan sistem terintegrasi dengan pendekatan sistem

B. Pengertian OLAP (*Online Analytical Process*)

Proses analitik *online* (OLAP) adalah teknologi perangkat lunak untuk menganalisis data bisnis. Penyimpanan data organisasi dikumpulkan dan disimpan di banyak sumber data, contohnya situs web, aplikasi, smart meter, dan sistem internal. OLAP menggabungkan dan mengelompokkan data ini ke dalam kategori untuk memberikan wawasan yang dapat ditindaklanjuti untuk perencanaan strategis. Misalnya, pengecer menyimpan data tentang semua produk yang dijualnya, seperti warna, ukuran, biaya, dan lokasi. Pengecer juga mengumpulkan data pembelian pelanggan, seperti nama barang yang dipesan dan total nilai penjualan, dalam sistem yang berbeda. OLAP menggabungkan kumpulan data untuk menjawab pertanyaan seperti produk warna mana yang lebih populer atau bagaimana penempatan produk berdampak pada penjualan.

OLAP digunakan untuk menganalisis data dan informasi yang nantinya akan digunakan sebagai *Decision Support System* (DSS). Beberapa aktivitas yang dapat dilakukan melalui OLAP antara lain seperti : melakukan proses query, meminta laporan, mendukung analisis statistik, melakukan analisis interaktif, dan membangun aplikasi multimedia.

Penggunaan alat intelijen bisnis dan OLAP dalam lingkungan elearning dan menyajikan studi kasus tentang bagaimana menerapkan teknologi ini dalam database sistem e-learning. Studi tersebut menunjukkan bahwa siswa menghabiskan sedikit waktu

dengan perangkat kursus dan lebih memilih untuk menggunakan aktivitas kolaboratif, seperti ruang kelas virtual dan forum daripada hanya melihat materi pembelajaran.. Pentingnya Sistem Intelijen serta arsitektur sistem pendukung interaktif pengambilan keputusan OLAP.

C. Manfaat OLAP (*Online Analytical Process*)

Pemrosesan analitis *online* (OLAP) membantu organisasi memproses dan mendapatkan manfaat dari makin banyaknya informasi digital.

Beberapa manfaat OLAP meliputi hal berikut.

1. Pengambilan keputusan lebih cepat

Bisnis menggunakan OLAP untuk mengambil keputusan yang cepat dan akurat agar tetap kompetitif dalam ekonomi yang serba cepat. Melakukan kueri analitis di beberapa basis data relasional sangat memakan waktu karena sistem komputer mencari di beberapa tabel data. Di sisi lain, sistem OLAP menghitung dan mengintegrasikan data sehingga analisis bisnis dapat menghasilkan laporan lebih cepat jika diperlukan.

2. Dukungan pengguna nonteknis

Sistem OLAP membuat analisis data yang kompleks lebih mudah bagi pengguna bisnis nonteknis. Pengguna bisnis dapat membuat perhitungan analitis yang kompleks dan menghasilkan laporan tanpa perlu mempelajari cara mengoperasikan basis data.

3. Tampilan data terintegrasi

OLAP menyediakan platform terpadu untuk pemasaran, keuangan, produksi, dan unit bisnis lainnya. Manajer dan pengambil keputusan dapat melihat gambaran yang lebih besar dan memecahkan masalah dengan efektif. Mereka dapat melakukan analisis *what-if*, yang menunjukkan dampak keputusan yang diambil oleh satu departemen di area bisnis lainnya.

D. Pembagian OLAP (*Online Analytical Process*)

Berdasarkan struktur basis datanya, OLAP dapat dibagi menjadi 3 bagian antara lain adalah:

1. *Multidimensional Online Analytical Processing* (MOLAP)

MOLAP adalah OLAP yang secara langsung mengarah pada basis data multidimensi. MOLAP memproses data yang telah disimpan dalam *array* multidimensional (semua kombinasi data yang mungkin dicerminkan), masing-masing diletakkan dalam suatu sel yang dapat diakses secara langsung.

2. *Relational Online Analytical Processing* (ROLAP)

ROLAP adalah suatu format pengolahan OLAP yang melakukan analisis data secara dinamis yang disimpan dalam basis data relasional bukan pada basis data multidimensi.

ROLAP merupakan bentuk teknologi dari OLAP yang paling berkembang.

3. *Hybrid Online Analytical Processing* (HOLAP)

HOLAP merupakan kombinasi antara ROLAP dengan MOLAP. HOLAP dikembangkan untuk mengkombinasikan antara kapasitas data pada ROLAP yang besar dengan kemampuan proses pada MOLAP.

E. Operasi-operasi OLAP

Operasi-operasi OLAP dibagi menjadi:

1. Roll up (drill-up): *summarize data by climbing up hierarchy or by dimension reduction.*
2. Drill down (roll down): *reverse of roll-up from higher level summary to lower level summary or detailed data, or introducing new dimensions*
3. Slice and dice: *project and select.*

4. Pivot (rotate): *reorient the cube, visualization, 3D to series of 2D planes.*

5. Other operations

drill across: involving (across) more than one fact table

drill through: through the bottom level of the cube to its back-end relational tables (using SQL)

F. Perbedaan OLAP dan OLTP (*Online Transactional Processing*)

Penambangan data adalah teknologi analitik yang memproses data historis dalam volume besar untuk menemukan pola dan wawasan. Analisis bisnis menggunakan alat penambangan data untuk menemukan hubungan dalam data dan membuat prediksi tren masa depan yang akurat.

1. OLAP dan penambangan data

Pemrosesan analitis *online* (OLAP) adalah teknologi analisis basis data yang melibatkan mengkueri, mengekstraksi, dan mempelajari data yang dirangkum. Di sisi lain, penggalian data melibatkan melihat secara saksama ke dalam informasi yang belum diproses. Misalnya, tenaga pemasaran dapat menggunakan alat penambangan data untuk menganalisis perilaku pengguna dari catatan setiap kunjungan situs web. Mereka kemudian dapat menggunakan perangkat lunak OLAP untuk memeriksa perilaku tersebut dari berbagai sudut, seperti durasi, perangkat, negara, bahasa, dan tipe browser. OLAP (pemrosesan analitis online). Tugas utama sistem gudang data adalah analisis data dan pengambilan keputusan

2. OLTP

Pemrosesan transaksi *online* (OLTP) adalah teknologi data yang menyimpan informasi dengan cepat dan andal dalam basis data. Rekayasawan data menggunakan alat OLTP untuk menyimpan data transaksional, seperti catatan keuangan, langganan layanan, dan umpan balik pelanggan, dalam basis data relasional. Sistem

OLTP melibatkan pembuatan, memperbarui, dan menghapus catatan dalam tabel relasional. OLTP (pemrosesan transaksi online). Tugas utama DBMS relasional tradisional. Operasi sehari-hari: pembelian, inventaris, perbankan, manufaktur, penggajian, registrasi, akuntansi, dll.

3. OLAP dan OLTP

OLTP sangat bagus untuk menangani dan menyimpan beberapa aliran transaksi dalam basis data. Namun, opsi ini tidak dapat melakukan kueri kompleks dari basis data. Oleh karena itu, analis bisnis menggunakan sistem OLAP untuk menganalisis data multidimensi. Misalnya, ilmuwan data menghubungkan untuk melakukan kueri intensif komputasi di data historis.

Tabel 1. Perbedaan OLAP dan OLTP (Vercelis,2009)

Karakteristik	OLTP	OLAP
Volatility	Data dinamis	Data statis
Ketepatan waktu	Data saat ini	Data saat ini dan masa lalu
Dimensi waktu	implisit dan terkini	eksplisit dan variasi
Granularitas	data rinci	data dikumpulkan dan dikonsolidasikan
Memperbarui	terus menerus dan tidak teratur	berkala dan teratur
Aktivitas	Berulang	Tidak dapat diprediksi
Fleksibilitas	Rendah	Tinggi
Performansi	Tinggi, beberapa detik per pertanyaan	Mungkin rendah untuk pertanyaan kompleks

Pengguna	Karyawan	Pekerja berpengetahuan
Fungsi	Operational	Analitikal
Tujuan penggunaan	Transaksi	Kueri yang kompleks dan pendukung keputusan
Prioritas	Performansi Tinggi	Fleksibilitas tinggi
Matriks	Transaksi tinggi	Respon yang efektif
Ukuran	Megabites ke Gigabites	Gigabites ke Terabites

G. Arsitektur OLAP

Data multidimensi disimpan oleh proses analitik *online* (OLAP) dengan menunjukkan informasi dalam lebih dari dua dimensi, atau kategori. Data dua dimensi melibatkan kolom dan baris, tetapi data multidimensi memiliki beberapa karakteristik. Misalnya, data multidimensi untuk penjualan produk mungkin terdiri dari dimensi berikut:

1. Waktu.
2. Lokasi.
3. Tipe produk.

Sistem OLAP multidimensi dibangun rekayasawan data terdiri dari elemen-elemen berikut:

1. Gudang data

Informasi dari berbagai sumber, termasuk aplikasi, *file*, dan basis data dikumpulkan dalam Gudang data. Ini memproses informasi menggunakan berbagai alat sehingga data siap untuk tujuan analitis. Informasi dari basis data relasional yang menyimpan data dalam tabel baris dan kolom dikumpulkan dalam gudang data.

2. ETL (Extract, Transform, Load)

Extract, Transform, Load (ETL) adalah proses basis data yang secara otomatis mengambil, mengubah, dan menyiapkan data ke format yang sesuai untuk tujuan analitis. Gudang data menggunakan ETL untuk mengonversi dan menstandarisasi informasi dari berbagai sumber sebelum membuatnya tersedia untuk alat OLAP.

3. *Server OLAP*

Sistem OLAP diberdayakan oleh mesin yang disebut *Server OLAP*. Server ini menggunakan ETL untuk mentransformasikan informasi dalam basis data relasional dan mempersiapkannya untuk operasional OLAP.

4. *Basis data OLAP*

Basis data terpisah yang terhubung ke gudang data disebut *Basis data OLAP*. Gudang data menggunakan analisis OLAP untuk pengolahan data dan menggunakan basis data OLAP. Pembuatan model data OLAP menggunakan basis data OLAP.

5. *Kubus OLAP*

Sebuah kubus data adalah model yang mewakili larik multidimensi informasi. Meski lebih mudah untuk memvisualisasikannya sebagai model data tiga dimensi, sebagian besar kubus data memiliki lebih dari tiga dimensi. *Kubus OLAP*, atau *hypercube*, adalah istilah untuk kubus data dalam sistem OLAP. *Kubus OLAP* kaku karena Anda tidak dapat mengubah dimensi dan data yang mendasarinya setelah Anda memodelnya. Misalnya, jika Anda menambahkan dimensi gudang ke kubus dengan dimensi produk, lokasi, dan waktu, Anda harus memodel ulang seluruh kubus.

6. *Alat analitik OLAP*

Analisis bisnis menggunakan alat OLAP untuk berinteraksi dengan kubus OLAP. Mereka melakukan operasi seperti *slicing*, *dicing*, dan *pivoting* untuk mendapatkan wawasan yang lebih mendalam tentang informasi spesifik dalam kubus OLAP.

H. Cara Kerja OLAP

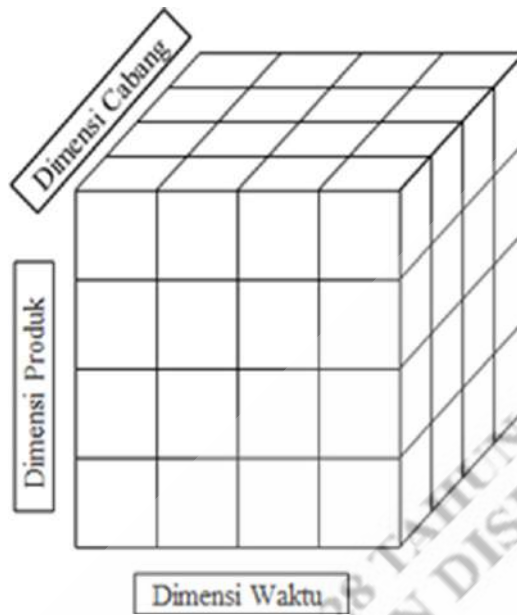
Online analytical processing (sistem pemrosesan analitis *online*/OLAP) bekerja dengan mengumpulkan, menata, mengagregatkan, dan menganalisis data menggunakan langkah-langkah berikut:

1. Server OLAP mengumpulkan data dari berbagai sumber data, termasuk basis data relasional dan gudang data.
2. Kemudian, alat *extract, transform, and load* (ETL) membersihkan, mengagregatkan, menghitung di awal, dan menyimpan data dalam kubus OLAP sesuai jumlah dimensi yang ditentukan.
3. Analisis bisnis menggunakan alat OLAP untuk mengkueri dan menghasilkan laporan dari data multidimensional di kubus OLAP.

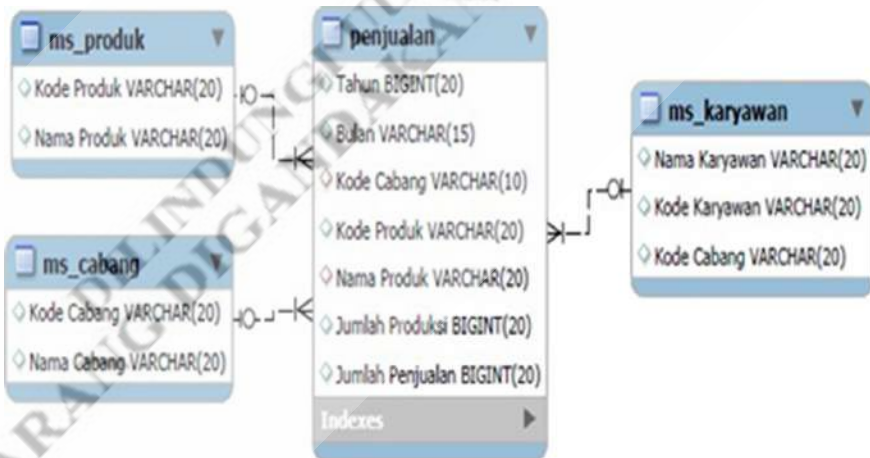
OLAP menggunakan *Multidimensional Expressions* (Eksprei Multidimensional/MDX) untuk kueri kubus OLAP. MDX adalah kueri, seperti SQL, yang menyediakan satu set instruksi untuk memanipulasi basis data.

Studi Kasus OLAP di pabrik roti (Fitriana et.al.,2018)

Tahap pengolahan cube terdiri dari beberapa langkah. Tahap pertama adalah membuat schema. Schema berisi database apa saja yang digunakan sehingga akan menampilkan hasil yang dapat menampilkan bentuk kubus. Dalam proses pembuatan OLAP Cube terdiri dari proses pembuatan struktur database yaitu Struktur OLTP dan struktur OLAP. Pembuatan OLTP ini merupakan system yang memproses transaksi di pabrik roti setiap harinya, sedangkan OLAP adalah suatu proses yang digunakan untuk melakukan analisis data penjualan pabrik roti. OLAP biasanya digunakan untuk pengambilan keputusan. Pada kasus ini OLAP dapat memberikan keputusan berupa produk mana yang paling laku terjual dan di cabang mana paling banyak terjualnya roti. Struktur OLTP dan OLAP database penjualan R Bakery dapat dilihat di Gambar 7.



Gambar 7. OLAP Cube



Gambar 8. Struktur OLTP Database penjualan R Bakery

Struktur OLTP Database penjualan R Bakery Langkah selanjutnya adalah pembuatan schema. Adapun database yang digunakan dalam schema ini adalah dimensi produk,

dimensi waktu, dan dimensi cabang. Penggabungan Dimensi OLAP Dari schema tersebut kemudian dilanjutkan dengan mengaplikasikan ke software Pentaho Analysis. Pentaho adalah software yang digunakan untuk pembuatan data warehouse dan business. CLV menghasilkan output berupa ranking dari pedagang di R Bakery. Usulan yang diberikan adalah dengan memberikan penghargaan kepada pedagang ranking 1 hingga 3 dan memberikan semangat dan motivasi kepada pedagang yang ranking dibawah 3. Usulan ini diberikan agar pada pedagang lebih semangat dalam memasarkan produk roti R Bakery.

I. Rangkuman

Proses analitik *online* (OLAP) adalah teknologi perangkat lunak untuk menganalisis data bisnis. Operasi operasi OLAP antara lain adalah Roll up (drill-up), Drill down (roll down), Slice and dice, Pivot (rotate). *Online analytical processing* (sistem pemrosesan analitis *online*/OLAP) bekerja dengan mengumpulkan, menata, mengagregatkan, dan menganalisis data. Berdasarkan struktur basis datanya, OLAP dapat dibagi menjadi 3 bagian yaitu *Multidimensional Online Analytical Processing* (MOLAP), *Relational Online Analytical Processing* (ROLAP) dan *Hybrid Online Analytical Processing* (HOLAP).

J. Soal Latihan

1. Beri penjelasan pengertian OLAP (Online Analytical Processing) dan contoh OLAP di industri
2. Jelaskan contoh operasi operasi OLAP.
3. Jelaskan beda OLAP dan OLTP.
4. Jelaskan cara kerja OLAP.
5. Jelaskan pembagian OLAP berdasarkan struktur basis data.
6. Jelaskan penerapan OLAP pada Jurnal dengan judul Implementasi Teknologi OLAP Pada Sistem Pengolahan Data

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

BAB III

Data Warehouse

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu menguasai prinsip dan teknik perancangan *Data Warehouse*.

Setelah membaca buku ini diharapkan mahasiswa mengetahui pengertian data warehouse, perbedaan data operasional dan data warehouse, karakteristik data warehouse, proses pengembangan data warehouse, arsitektur data warehouse, data mart.

Materi yang diberikan :

1. Pengertian *Data Warehouse*.
2. Arsitektur *Data Warehouse*.
3. *Data Mart*.

Capaian Pembelajaran:

Capaian pembelajaran mata kuliah pada bab ini adalah mahasiswa mampu menguasai pengertian *Data Warehouse*,

Arsitektur Data Warehouse, Data Mart. Capaian pembelajaran pada bab ini adalah mampu menguasai prinsip dan teknik perancangan sistem terintegrasi dengan pendekatan sistem.

B. Pengertian Data Warehouse

Dalam era digital yang terus maju, data menjadi aset yang sangat bernilai bagi setiap organisasi. Data digunakan untuk pengambilan keputusan, perumusan strategi, dan pemahaman tren pasar. Dengan semakin banyaknya data yang tersedia, kebutuhan akan sistem penyimpanan dan pengelolaan data yang efisien menjadi semakin penting. Business intelligence meliputi berbagai metodologi, proses, arsitektur, dan teknologi yang mengubah data mentah menjadi informasi yang berarti dan berguna untuk pengambilan keputusan. Business intelligence (BI) dan sistem pendukung keputusan membantu manajer di berbagai tingkatan organisasi dalam menganalisis informasi strategis. Sistem ini mengumpulkan sejumlah besar data dan mereduksinya menjadi bentuk yang dapat digunakan untuk menganalisis perilaku organisasi. Transformasi data melibatkan serangkaian tugas yang mengambil data dari sumbernya dan, melalui proses ekstraksi, transformasi, integrasi, dan pembersihan, menyimpannya dalam repositori umum yang disebut data warehouse. data warehouse dikembangkan dan diterapkan sebagai bagian penting dari sistem pendukung keputusan untuk menyediakan infrastruktur yang memungkinkan pengguna mendapatkan respons yang efisien dan akurat terhadap pertanyaan yang kompleks. Data warehouse hadir sebagai solusi untuk mengelola data dalam skala besar dan memungkinkan analisis yang mendalam. Buku ini bertujuan untuk membantu pembaca memahami prinsip dan teknik perancangan data warehouse yang krusial untuk pengelolaan data yang efektif dalam organisasi.

Beragam sistem dan alat dapat digunakan untuk mengakses, menganalisis, dan memanfaatkan data yang ada dalam data warehouse. Sejak awal pengembangan data warehouse, mekanisme yang umum untuk tugas-tugas ini adalah Online Analytical Processing (OLAP). Sistem OLAP memungkinkan pengguna untuk melakukan query secara interaktif dan mengagregasi data dalam

data warehouse secara otomatis. Dengan demikian, pengambil keputusan dapat dengan mudah mengakses informasi yang dibutuhkan dan menganalisisnya pada berbagai tingkat detail. Data mining juga telah digunakan sejak untuk menemukan dan mengekstrak pengetahuan yang tersembunyi dalam data warehouse. Banyak teknik business intelligence baru telah dikembangkan dan digunakan untuk mendukung pengambilan keputusan. Oleh karena itu, pasar business intelligence bergerak ke arah penyediaan alat analisis canggih yang dikenal secara umum sebagai analitik data. Teknik business intelligence yang digunakan untuk memanfaatkan data warehouse yaitu (Vaisman & Zimányi, 2014):

1. Pelaporan, seperti dashboard dan peringatan.
2. Manajemen kinerja, seperti metrik, indikator kinerja utama, dan scorecards.
3. Analitik, seperti OLAP, data mining, time series analysis, text mining, web analytics, and advanced data visualization.

Data warehouse adalah sistem yang dirancang khusus untuk keperluan query dan analisis, bukan untuk pemrosesan transaksi sehari-hari. Sistem ini mengumpulkan data dari berbagai sumber dalam organisasi untuk menyediakan basis data terpadu yang mudah diakses untuk analisis bisnis. Proses pengembangan data warehouse diringkaskan dalam langkah-langkah dasar berikut (Singh, 2013):

1. **Modeling**, langkah ini melibatkan pemahaman mendalam tentang proses bisnis, kebutuhan informasi yang diperlukan untuk proses tersebut, dan keputusan yang saat ini dibuat dalam proses tersebut.
2. **Building**, menentukan kebutuhan alat yang tepat untuk jenis dukungan keputusan yang diperlukan dalam proses bisnis yang ditargetkan; merancang model data yang membantu menjelaskan lebih lanjut kebutuhan informasi; memecah masalah menjadi spesifikasi data dan penyimpanan data yang aktual, yang pada akhirnya akan membentuk data mart atau data warehouse yang lebih luas.

3. **Deploying**, mengimplementasikan secara awal dalam keseluruhan proses, termasuk menentukan jenis data yang akan disimpan dan berbagai alat business intelligence yang akan digunakan; memulai dengan pelatihan pengguna. Tahap deploy secara khusus mencakup periode di mana pengguna mengeksplorasi repositori (untuk memahami data yang ada dan yang seharusnya ada) serta versi awal dari data warehouse. Ini dapat menyebabkan evolusi data warehouse, seperti penambahan data, perpanjangan periode historis, atau kembali ke tahap building untuk memperluas cakupan data warehouse dengan model data.

Data warehouse adalah sistem komputer yang didedikasikan untuk menyimpan dan menganalisis data untuk mengungkapkan tren, pola, dan korelasi yang memberikan informasi dan wawasan. Secara tradisional, organisasi telah menggunakan gudang data untuk menyimpan dan mengintegrasikan data yang dikumpulkan dari sumber internal mereka (biasanya database transaksional), termasuk pemasaran, penjualan, produksi, dan keuangan (Kraynak & Baum, 2020). *Data warehouse* muncul ketika perusahaan menyadari bahwa menganalisis data langsung dari database transaksional memperlambat analisis data tersebut. Oleh karena itu, semua data itu diduplikasi di *data warehouse* untuk analisis.

Tabel 2. Perbedaan Basis Data Operasional dan Data Warehouses (Singh, 2013)

Fitur	Basis Data Operasional	Data Warehouse
Pengguna	Ribuan	Ratusan
Beban kerja	Transaksi yang sudah ditetapkan	Query analisis khusus
Akses	Hingga ratusan catatan, mode tulis dan baca	Hingga jutaan catatan, umumnya dalam mode baca saja
Tujuan	Bergantung pada aplikasi	Dukungan pengambilan keputusan
Data	Terperinci, baik numerik maupun alfanumerik	Dijumlahkan, terutama numerik

Integrasi data	Berdasarkan aplikasi	Berdasarkan subjek
Kualitas	Dalam hal integritas	Dalam hal konsistensi
Cakupan waktu	Data saat ini saja	Data saat ini dan historis
Pembaruan	Kontinu	Periode
Model	Ternormalisasi	Denormalisasi, multidimensional
Optimasi	Untuk akses OLTP ke bagian basis data	Untuk akses OLAP ke sebagian besar basis data

Data warehouse dapat diartikan sebagai gudang data. Secara sederhana, data warehouse adalah kumpulan data yang dihasilkan untuk mendukung pengambilan keputusan; Ini juga merupakan repositori data terkini dan historis yang berpotensi menarik bagi manajer di seluruh organisasi (Sharda et al., 2018). Data biasanya disusun agar tersedia dalam bentuk yang siap untuk kegiatan pemrosesan analitik. Bill Inmon mendefinisikan data warehouse sebagai basisdata yang memiliki karakter: subject oriented, integrated, time Variant, non-volatile (Sharda et al., 2018).

1. **Subject oriented.** Data diatur berdasarkan subjek tertentu, seperti penjualan, produk, atau pelanggan, dan hanya mencakup informasi yang relevan untuk mendukung pengambilan keputusan. Dengan orientasi subjek, pengguna dapat mengetahui tidak hanya bagaimana kinerja bisnis mereka, tetapi juga alasan di baliknya. Data warehouse berbeda dari database operasional, yang biasanya berfokus pada produk dan disesuaikan untuk menangani transaksi yang memperbarui database. Orientasi subjek memberikan pandangan yang lebih komprehensif tentang organisasi.
2. **Integrated.** Integrasi sangat erat kaitannya dengan subject oriented. Data warehouse harus menggabungkan data dari berbagai sumber ke dalam format yang konsisten. Untuk melakukannya, mereka harus mengatasi konflik penamaan

dan perbedaan dalam satuan pengukuran. Data warehouse dianggap sepenuhnya terintegrasi.

3. **Time variant (time series).** Data warehouse menyimpan data historis. Data tidak selalu menunjukkan status saat ini (kecuali dalam sistem real-time). Mereka mendeteksi tren, penyimpangan, dan hubungan jangka panjang untuk peramalan dan perbandingan, yang membantu dalam pengambilan keputusan. Setiap data warehouse memiliki aspek temporal. Waktu adalah dimensi penting yang harus didukung oleh semua data warehouse. Data untuk analisis dari berbagai sumber mencakup beberapa titik waktu (misalnya, tampilan harian, mingguan, bulanan).
4. **Non-volatile.** Setelah data dimasukkan ke dalam data warehouse, data tersebut tidak dapat diubah atau diperbarui oleh pengguna. Data yang sudah usang dibuang, dan perubahan dicatat sebagai data baru.

Data Warehousing menggambarkan proses ekstraksi, pemberian, transformasi, pengendalian, dan pemuatan operasional yang menjaga data dalam sebuah data warehouse. Tujuan utama pembuatan data warehouse adalah untuk menyatukan data yang beragam ke dalam sebuah tempat penyimpanan dimana pengguna dapat dengan mudah antara lain untuk menjalankan query (pencarian data), menghasilkan laporan, dan melakukan analisis. Salah satu keuntungan yang diperoleh dari keberadaan data warehouse adalah dapat meningkatkan efektivitas pembuatan keputusan.

Untuk membangun data warehouse yang baik diperlukan tool yang mampu mengambil, mengolah dan menyimpan data pada berbagai format dengan baik. Tool ini biasa disebut dengan ETL, singkatan dari Extract, Transform and Load. Salah satu produk ETL adalah Kettle atau Pentaho Data Integration, suatu produk yang bersifat free open source ETL dan dapat berjalan di lingkungan multi-platform. Kettle memiliki lebih dari 140 built-in modules yang dapat digunakan hampir semua format data terpopuler dan biasa digunakan di dunia enterprise saat ini. Dengan kemampuannya

yang sangat baik dan didukung oleh komunitas open source, Kettle menjadi aplikasi favorit di antara para pengembang data warehouse.

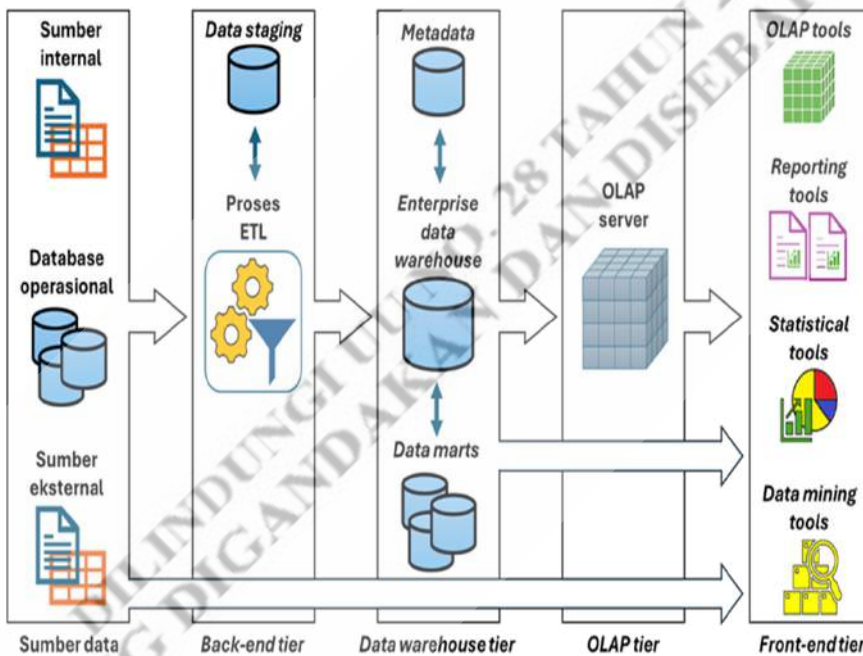
C. Arsitektur Data Warehouse

Sebuah Data Warehouse (DW) terdiri dari dua komponen utama: basis data dukungan keputusan yang terintegrasi dan perangkat lunak terkait yang berfungsi untuk mengumpulkan, membersihkan, mentransformasi, dan menyimpan data dari berbagai sumber operasional dan eksternal. Untuk memenuhi kebutuhan historis, analitis, dan BI, sebuah data warehouse juga bisa mencakup data mart dependen, yaitu salinan subset data dari warehouse tersebut. Secara umum, data warehouse mencakup segala penyimpanan data atau ekstrak yang digunakan untuk mendukung penyajian data bagi tujuan BI. Enterprise Data Warehouse (EDW) adalah data warehouse terpusat yang dirancang untuk memenuhi kebutuhan BI seluruh organisasi. EDW mengikuti model data perusahaan untuk memastikan konsistensi dalam aktivitas dukungan keputusan di seluruh organisasi (DAMA International, 2017).

Arsitektur merupakan salah satu komponen penting dalam perancangan data warehouse. Arsitektur data warehouse yang baik memastikan bahwa data dapat diakses dengan cepat dan efisien. Terdapat berbagai elemen dalam arsitektur data warehouse, termasuk sumber data, proses ETL (*Extract, Transform, Load*), penyimpanan data, dan teknologi OLAP (*Online Analytical Processing*). Elemen-elemen ini bekerja sama untuk menyediakan platform analisis yang tangguh, tepat guna, dan sesuai dengan kebutuhan bisnis. Secara umum arsitektur data warehouse terdiri dari beberapa tingkatan (Vaisman & Zimányi, 2014):

1. **Back-end tier** mencakup extraction, transformation, and loading (ETL) tools yang berfungsi untuk memindahkan data ke dalam data warehouse dari operational databases dan sumber data lainnya, baik dari dalam maupun luar organisasi, serta data staging area yang merupakan database perantara tempat semua proses integrasi dan transformasi data dilakukan sebelum data dimasukkan ke dalam data warehouse.

2. **Data warehouse tier** meliputi enterprise data warehouse dan/atau beberapa data mart serta metadata repository yang menyimpan informasi tentang data warehouse dan isinya.
3. **OLAP tier** terdiri dari OLAP server yang menawarkan pandangan multidimensional terhadap data, tanpa memperhatikan cara data disimpan dalam sistem yang mendasarinya.
4. **Front-end tier** digunakan untuk analisis dan visualisasi data, dan mencakup alat-alat klien seperti OLAP tools, reporting tools, statistical tools, dan data mining tools.



Gambar 9. Arsitektur Data Warehouse Umum

Berikut ini penjelasan rinci berbagai komponen arsitektur yang umum:

1. Back-end tier

Pada back-end tier, proses yang sering disebut sebagai extraction, transformation, and loading dilakukan. Seperti namanya,

proses ini terdiri dari tiga langkah berikut:

- a. **Extraction**, mengumpulkan data dari berbagai sumber yang beragam. Sumber-sumber ini bisa berupa database operasional atau file dengan format yang berbeda, data bisa berasal dari dalam organisasi atau dari luar.
- b. **Transformation**, mengubah data dari format sumber ke format yang sesuai untuk data warehouse. Proses ini mencakup beberapa hal: **cleaning**, yang menghapus kesalahan dan inkonsistensi dalam data serta mengubahnya ke format yang standar; **integration**, yang menyelaraskan data dari berbagai sumber baik pada tingkat skema maupun data; dan **aggregation**, yang merangkum data dari sumber sesuai dengan data warehouse.
- c. **Loading**, memuat data yang sudah diubah ke dalam data warehouse. Ini juga termasuk memperbarui data dari sumber ke data warehouse pada frekuensi tertentu untuk memastikan data tetap terkini untuk proses pengambilan keputusan. Frekuensi penyegaran ini bisa bervariasi dari bulanan hingga beberapa kali sehari atau bahkan mendekati waktu nyata, tergantung pada kebijakan organisasi.

Proses ETL umumnya memerlukan data staging area, yaitu sebuah database di mana data yang diekstrak mengalami serangkaian modifikasi sebelum akhirnya siap dimuat ke dalam data warehouse.

2. Data warehouse tier

Pada data warehouse tier terdapat enterprise data warehouse dan beberapa data mart. Enterprise data warehouse bersifat terpusat dan mencakup seluruh organisasi, data mart adalah data warehouse khusus yang ditujukan untuk area fungsional atau departemen tertentu dalam organisasi. Data mart bisa dianggap sebagai data warehouse kecil yang lokal. Data dalam data mart dapat berasal dari enterprise data warehouse atau dikumpulkan langsung dari sumber data.

Komponen lain dari tingkatan data warehouse adalah metadata repository. Metadata didefinisikan sebagai "data tentang data." Metadata biasanya dibagi menjadi metadata teknis dan metadata bisnis. Metadata bisnis menjelaskan makna (atau semantik) dari data serta aturan, kebijakan, dan batasan organisasi yang terkait dengan data tersebut. Sementara itu, metadata teknis menjelaskan bagaimana data diatur dan disimpan dalam sistem komputer serta aplikasi dan proses yang memanipulasi data tersebut.

Dalam konteks data warehouse, metadata teknis bisa mencakup berbagai hal, menggambarkan sistem data warehouse, sistem sumber, dan proses ETL. Secara khusus, metadata repository mungkin berisi informasi berikut:

- a. Metadata yang menggambarkan struktur data warehouse dan data mart, baik pada tingkat konseptual/logis (termasuk fakta, dimensi, hierarki, dan definisi data turunan) maupun pada tingkat fisik (seperti indeks, partisi, dan replikasi). Metadata ini juga mencakup informasi keamanan (otorisasi pengguna dan kontrol akses) dan informasi pemantauan (seperti statistik penggunaan, laporan kesalahan, dan jejak audit).
- b. Metadata yang menggambarkan sumber data, termasuk skema data (pada tingkat konseptual, logis, dan/atau fisik), serta informasi deskriptif seperti kepemilikan, frekuensi pembaruan, batasan hukum, dan metode akses.
- c. Metadata yang menggambarkan proses ETL, termasuk data lineage (penelusuran data warehouse kembali ke data sumber), aturan dan default untuk ekstraksi, pembersihan, dan transformasi data, serta aturan dan algoritma untuk penyegaran dan pembersihan data.

3. OLAP tier

Pada OLAP tier dalam arsitektur terdiri dari OLAP server, yang menyediakan data multidimensional kepada pengguna bisnis dari data warehouses atau data mart. Sebagian besar produk database menawarkan ekstensi OLAP dan alat terkait yang memungkinkan pembuatan dan querying cubes, serta navigasi, analisis, dan

pelaporan. Namun, belum ada bahasa standar untuk mendefinisikan dan memanipulasi data cubes. MDX (Multi Dimensional eXpressions) adalah bahasa query yang digunakan untuk database OLAP. Dukungan dari berbagai vendor OLAP membuat MDX menjadi standar yang berlaku umum untuk melakukan query pada sistem OLAP.

4. Front-end tier

Pada front-end tier mencakup alat-alat klien yang memungkinkan pengguna untuk memanfaatkan konten dari data warehouse. Alat-alat klien yang biasa digunakan meliputi:

- a. OLAP tools memberikan kemampuan eksplorasi dan manipulasi interaktif data warehouse. Alat ini memfasilitasi pembuatan query kompleks yang mungkin melibatkan banyak data.
- b. Reporting tools mendukung pembuatan, pengiriman, dan pengelolaan laporan, baik dalam bentuk laporan kertas maupun laporan interaktif berbasis web. Laporan ini menggunakan query yang telah ditetapkan sebelumnya, yang meminta informasi tertentu dalam format tertentu secara rutin. Teknik pelaporan modern termasuk key performance indicator dan dashboard.
- c. Statistical tools digunakan untuk menganalisis dan memvisualisasikan data cube dengan metode statistik.
- d. Data mining tools memungkinkan pengguna untuk menganalisis data guna menemukan informasi berharga seperti pola dan tren, serta membuat prediksi berdasarkan data saat ini.

Beberapa komponen yang ditampilkan pada gambar arsitektur mungkin tidak selalu ada di lingkungan nyata. Dalam beberapa kasus, hanya terdapat enterprise data warehouse tanpa data mart, atau bahkan enterprise data warehouse tidak ada sama sekali. Membangun enterprise data warehouse merupakan tugas yang kompleks dan membutuhkan banyak waktu serta sumber daya. Sebaliknya, data mart biasanya lebih mudah untuk dibangun

dibandingkan dengan enterprise warehouse. Dalam beberapa kondisi lain, OLAP server mungkin tidak ada dan/atau klien langsung mengakses data warehouse. Hal ini ditunjukkan oleh panah yang menghubungkan tingkatan data warehouse dengan tingkatan front-end. Area staging data mungkin tidak diperlukan jika data dalam sistem sumber sangat mirip dengan data dalam warehouse. Situasi ini biasanya terjadi ketika hanya ada satu sumber data (atau beberapa) dengan kualitas data yang tinggi.

D. Data Mart

Data warehouse dirancang untuk menganalisis data seluruh organisasi. Sebuah data warehouse menyimpan data tentang seluruh organisasi yang mendukung pengguna di tingkat manajemen tinggi dalam membuat keputusan strategis. Namun, keputusan ini juga dapat diambil pada tingkat organisasi yang lebih rendah yang berhubungan dengan area bisnis tertentu, di mana hanya sebagian data dari data warehouse yang dibutuhkan. Seringkali, departemen atau divisi tertentu dalam organisasi hanya membutuhkan sebagian dari data warehouse yang disesuaikan dengan kebutuhan mereka. Misalnya, departemen penjualan mungkin hanya memerlukan data penjualan, sedangkan departemen sumber daya manusia mungkin membutuhkan data demografi dan data karyawan. Data warehouse yang khusus untuk departemen ini disebut data mart. Namun, data mart ini tidak harus bersifat eksklusif untuk satu departemen; data ini bisa saja dibagikan dengan bagian lain dari organisasi yang memerlukannya.

Data mart merupakan bentuk dari data warehouse dengan cakupan yang terbatas, dirancang untuk mendukung pemrosesan analitis, dan biasanya melayani satu departemen, bagian tertentu dari organisasi, atau domain masalah analisis data spesifik. Ini adalah jenis data warehouse yang hanya menyimpan data yang relevan untuk domain masalah tertentu dan dioptimalkan secara fisik untuk kebutuhan analisis serta alat-alat yang mendukungnya. Alasan membangun data mart yaitu:

1. Memberikan akses data yang paling sering dibutuhkan oleh pengguna untuk analisis.

2. Menyajikan data dalam format yang sesuai dengan pandangan bersama sekelompok pengguna di sebuah departemen atau area bisnis.
3. Meningkatkan waktu respons pengguna dengan mengurangi volume data yang perlu diakses.
4. Menyediakan data yang terstruktur sesuai dengan kebutuhan alat akses pengguna, seperti Online Analytical Processing (OLAP) dan alat data mining, yang mungkin memerlukan struktur basis data internal.
5. Data mart umumnya menggunakan data yang lebih sedikit, sehingga proses seperti pembersihan, pemuatan, transformasi, dan integrasi data menjadi lebih mudah. Karena itu, membuat dan mengatur data mart biasanya lebih sederhana dibandingkan dengan membangun data warehouse perusahaan.
6. Biaya untuk membuat data mart biasanya lebih rendah dibandingkan dengan biaya yang diperlukan untuk membangun data warehouse.
7. Pengguna potensial data mart lebih jelas dan lebih mudah dijangkau untuk mendapatkan dukungan dalam proyek data mart dibandingkan dengan proyek data warehouse perusahaan.

Struktur data mart mirip dengan data warehouse tetapi lebih kecil ukurannya. Data mart bisa ditempatkan secara fisik bersama dengan data warehouse atau memiliki platform terpisah. Seperti dalam database operasional, ada dua metode utama untuk merancang data warehouse dan data mart terkait:

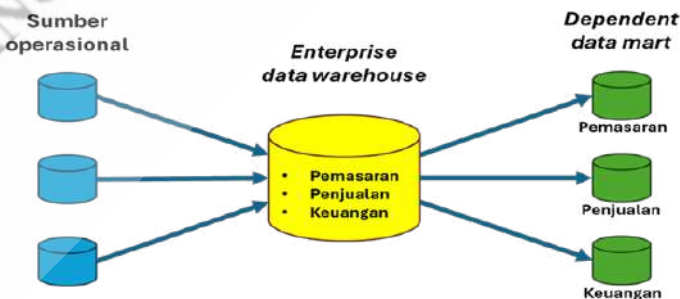
1. **Desain top-down:** Kebutuhan pengguna di berbagai tingkat organisasi dikumpulkan terlebih dahulu sebelum proses desain dimulai, dan satu skema untuk seluruh data warehouse dibuat. Setelah itu, data mart terpisah disesuaikan dengan karakteristik dari setiap area bisnis atau proses.
2. **Desain bottom-up:** Setiap data mart dibuat dengan skema terpisah, mempertimbangkan kebutuhan pengguna yang bertanggung jawab untuk area bisnis atau proses tertentu. Kemudian, skema-skema ini digabungkan menjadi satu skema

global untuk seluruh data warehouse.

Pemilihan antara pendekatan top-down dan bottom-up bergantung pada banyak faktor, seperti keahlian profesional tim pengembang, ukuran data warehouse, motivasi pengguna untuk memiliki data warehouse, dan dukungan finansial, di antara hal-hal lainnya. Pengembangan data warehouse skala perusahaan dengan pendekatan top-down bisa menjadi tantangan besar bagi banyak organisasi dalam hal biaya dan waktu. Ini juga merupakan tugas yang menantang bagi desainer karena ukuran dan kompleksitasnya. Sebaliknya, ukuran data mart yang lebih kecil memungkinkan pengembalian investasi lebih cepat dan memudahkan proses pengembangan. Jika motivasi pengguna rendah, pendekatan bottom-up dapat menghasilkan data mart dengan lebih cepat dan biaya lebih rendah, sehingga pengguna dapat segera menggunakan alat OLAP dan membuat laporan baru; ini dapat meningkatkan penerimaan pengguna dan motivasi untuk memiliki data warehouse

Ada tiga jenis dasar data mart: dependent, independent, dan hybrid. Kategorisasi ini didasarkan pada sumber data yang digunakan untuk mengisi data mart. Data mart dependent mendapatkan data dari data warehouse pusat yang sudah ada. Sebaliknya, data mart independent adalah sistem mandiri yang mengambil data langsung dari sumber operasional atau eksternal, atau keduanya. Data mart hybrid dapat mengambil data dari sistem operasional maupun data warehouse (Singh, 2013).

Data mart dependent memungkinkan penyatuan data organisasi dalam satu data warehouse, memberikan keuntungan dari sentralisasi.



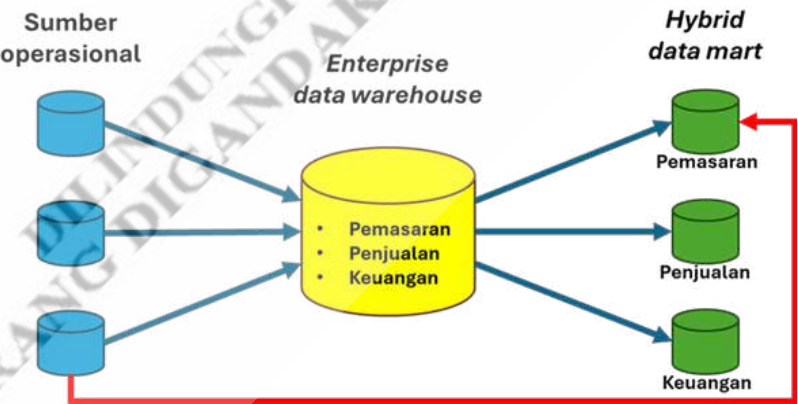
Gambar 10. Ilustrasi Data Mart Dependent

Data mart independent dibangun tanpa memerlukan data warehouse pusat, yang mungkin bermanfaat untuk kelompok kecil dalam organisasi.



Gambar 11. Ilustrasi Data Mart Independent

Data mart hybrid memungkinkan kombinasi data dari berbagai sumber selain data warehouse, yang bermanfaat dalam situasi yang memerlukan integrasi ad hoc, seperti setelah penambahan kelompok atau produk baru dalam organisasi.



Gambar 12. Ilustrasi Data Mart Hybrid

Sebagai contoh pengembangan data mart dalam konteks sebuah perusahaan ritel yaitu sebuah perusahaan ritel ingin meningkatkan kemampuan analisis penjualannya. Mereka memiliki data dari

berbagai sumber, termasuk sistem point of sale (POS), inventaris, dan data pelanggan dari program loyalitas. Langkah-langkah pengembangan data mart yang dilakukan yaitu:

1. Identifikasi Kebutuhan Bisnis: Tim manajemen menentukan bahwa mereka memerlukan analisis yang lebih mendalam tentang tren penjualan, preferensi pelanggan, dan pengelolaan inventaris untuk memperbaiki strategi pemasaran dan pengelolaan stok.
2. Pemilihan Data: Data dari sistem POS, data inventaris, dan informasi program loyalitas pelanggan dipilih untuk dimasukkan ke dalam data mart. Data ini mencakup informasi penjualan harian, data produk, dan profil pelanggan.
3. Desain Skema: Tim pengembang membuat skema data mart yang mencakup tabel fakta untuk penjualan dan tabel dimensi untuk produk, waktu, dan pelanggan. Skema ini dirancang untuk memungkinkan analisis yang cepat dan fleksibel.
4. Ekstraksi, Transformasi, dan Loading (ETL): Proses ETL diterapkan untuk mengambil data dari berbagai sumber, membersihkan dan mengubahnya sesuai kebutuhan, dan memuatnya ke dalam data mart.
5. Pengujian dan Validasi: Setelah data mart dibangun, dilakukan pengujian untuk memastikan integritas data dan memvalidasi hasil analisis. Pengguna akhir dilibatkan untuk memberikan masukan.
6. Implementasi Alat BI: Perusahaan mengimplementasikan alat Business Intelligence (BI) untuk memudahkan analis dan manajer dalam mengakses dan menganalisis data dari data mart. Alat ini memungkinkan pembuatan laporan dan visualisasi data yang interaktif.
7. Pelatihan Pengguna: Pelatihan diberikan kepada pengguna akhir tentang cara menggunakan alat BI dan memanfaatkan data mart untuk analisis dan pelaporan.

Dengan data mart, perusahaan ritel tersebut dapat dengan cepat menganalisis tren penjualan, memahami preferensi pelanggan, dan

mengelola inventaris dengan lebih efisien. Ini membantu mereka dalam merancang strategi pemasaran yang lebih tepat sasaran dan mengurangi biaya operasional terkait manajemen stok.

E. Rangkuman

Data menjadi aset yang sangat berharga dalam organisasi untuk pengambilan keputusan strategis, dengan business intelligence berperan dalam mengubah data mentah menjadi informasi yang berguna.

Data warehouse merupakan bagian penting dari sistem pendukung keputusan, menyediakan infrastruktur untuk analisis data yang mendalam dan respons yang efisien terhadap pertanyaan yang kompleks.

Teknik seperti OLAP dan data mining digunakan untuk memanfaatkan data warehouse, memungkinkan query interaktif dan analisis data yang lebih dalam, serta menemukan pengetahuan tersembunyi.

Data warehouse adalah sistem komputer yang menyimpan dan menganalisis data untuk mengungkap tren, pola, dan korelasi yang memberikan wawasan bagi organisasi. Data ini biasanya diambil dari sumber internal seperti database transaksional untuk mempermudah analisis tanpa memperlambat kinerja sistem operasional.

Data warehouse memiliki karakteristik penting seperti subject oriented (berorientasi pada subjek), integrated (terintegrasi dari berbagai sumber), time variant (menyimpan data historis), dan non-volatile (data yang tidak dapat diubah setelah dimasukkan). Ini memungkinkan pengambilan keputusan yang lebih baik dengan menyediakan pandangan yang komprehensif dan mendalam.

Untuk membangun data warehouse yang efektif, digunakan alat ETL (Extract, Transform, Load) seperti Kettle atau Pentaho Data Integration. Alat ini membantu dalam mengambil, mengolah, dan menyimpan data dari berbagai format. Kettle, sebagai produk

ETL open source, populer karena kemampuannya yang luas dan dukungan komunitas.

Arsitektur data warehouse terdiri dari beberapa tingkatan yang bekerja sama untuk memastikan akses data yang cepat dan efisien. Tingkatan ini meliputi proses ETL untuk memindahkan data, penyimpanan data dalam data warehouse, OLAP untuk analisis multidimensional, dan alat front-end untuk visualisasi dan analisis data.

Elemen utama dalam arsitektur data warehouse termasuk back-end tier (ETL dan data staging area), data warehouse tier (enterprise data warehouse, data mart, dan metadata repository), OLAP tier (server OLAP untuk pandangan multidimensional), dan front-end tier (alat untuk analisis dan visualisasi data).

Data warehouse menyimpan data organisasi secara keseluruhan untuk mendukung keputusan strategis, sedangkan data mart adalah subset data warehouse yang disesuaikan dengan kebutuhan departemen tertentu. Data mart dapat berbagi data dengan bagian lain dari organisasi dan memiliki struktur yang mirip dengan data warehouse tetapi lebih kecil.

Desain data warehouse dapat dilakukan dengan pendekatan top-down, yang mengumpulkan kebutuhan pengguna di berbagai tingkat sebelum membuat skema global, atau pendekatan bottom-up, di mana setiap data mart dibuat dengan skema terpisah dan kemudian digabungkan. Pemilihan metode tergantung pada faktor-faktor seperti keterampilan tim, ukuran data warehouse, dan dukungan finansial.

F. Soal Latihan

1. Mengapa data menjadi aset yang sangat bernilai bagi organisasi dalam era digital?
2. Sebutkan beberapa teknik business intelligence yang digunakan untuk memanfaatkan data warehouse dan berikan contoh masing-masing teknik tersebut.

3. Apa peran utama data warehouse dalam analisis data dan bagaimana ia mengatasi kelemahan dari sistem database transaksional?
4. Sebutkan dan jelaskan empat karakteristik utama dari data warehouse menurut Bill Inmon.
5. Apa itu alat ETL, dan bagaimana Kettle atau Pentaho Data Integration berkontribusi dalam pembangunan data warehouse?
6. Apa saja komponen utama dari arsitektur data warehouse dan fungsi masing-masing komponen tersebut?
7. Jelaskan perbedaan antara back-end tier dan front-end tier dalam arsitektur data warehouse.
8. Apa peran OLAP server dalam arsitektur data warehouse dan bagaimana ia mempengaruhi analisis data?
9. Apa perbedaan antara data warehouse dan data mart?
10. Jelaskan dua pendekatan utama dalam merancang data warehouse dan data mart.

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

BAB IV

Database

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu menguasai prinsip dan teknik perancangan *Database*.

Setelah membaca buku ini diharapkan mahasiswa mengetahui pengertian data base, pembuatan database dan impor database

Materi yang diberikan :

1. Pengertian Database.
2. Pembuatan Database.
3. Impor Database.
4. Model Data.
5. Deskripsi Kasus.
6. Implementasi.

Capaian Pembelajaran:

Capaian pembelajaran mata kuliah pada bab ini adalah mahasiswa mampu menguasai prinsip dan teknik perancangan Database. Capaian pembelajaran pada bab ini adalah mampu menguasai prinsip dan teknik perancangan sistem terintegrasi dengan pendekatan sistem.

B. Pengertian Database

Perusahaan mengandalkan sistem informasi untuk menjalankan operasionalnya. Aktivitas yang berkaitan dengan penyimpanan dan pengelolaan data sangat penting bagi organisasi yang bergantung pada data. Jika sistem tidak tersedia, operasional perusahaan bisa terganggu atau bahkan berhenti total. Infrastruktur data storage yang handal untuk operasi IT mengurangi risiko terjadinya gangguan. Data adalah sekumpulan unit informasi kecil yang unik. Data dapat berbentuk teks, angka, media, bytes, dan lain-lain, dan dapat disimpan dalam berbagai cara, seperti di atas kertas atau dalam memori elektronik. Database adalah kumpulan data yang disimpan tanpa mempedulikan struktur atau kontennya (DAMA International, 2017). Database dapat diartikan sebagai kumpulan data yang tersusun secara teratur agar dapat diakses dan dikelola dengan mudah. Data dapat diatur dalam tabel, baris, kolom, dan diindeks untuk mempermudah pencarian informasi yang relevan. Tujuan utama database adalah mengelola informasi dalam jumlah besar dengan menyimpan, mengambil, dan mengatur data. Saat ini, banyak situs web dinamis di World Wide Web yang dikelola melalui database. Misalnya, model yang memeriksa ketersediaan tiket pesawat terbang adalah contoh situs web dinamis yang menggunakan database. Menguasai prinsip dan teknik perancangan database sangat penting karena beberapa alasan berikut:

1. **Efisiensi Pengelolaan Data:** Perancangan database yang efektif memastikan data tersimpan secara optimal dan dapat diakses dengan cepat, mengurangi waktu pencarian data dan meningkatkan kinerja sistem.
2. **Konsistensi Data:** Teknik perancangan yang tepat membantu menjaga integritas dan konsistensi data dengan menetapkan

aturan dan batasan yang jelas, sehingga menghindari duplikasi dan kesalahan data.

3. **Kemudahan Perawatan:** Database yang dirancang dengan baik mempermudah proses pemeliharaan dan perbaikan, dengan struktur yang jelas memudahkan pembaruan, penambahan fitur, dan perbaikan masalah.
4. **Kemampuan Skalabilitas:** Dengan memahami prinsip perancangan, kita dapat membuat database yang mampu berkembang sesuai dengan pertumbuhan volume data dan beban kerja, tanpa penurunan performa.
5. **Keamanan Data:** Prinsip desain yang baik mencakup aspek keamanan, seperti kontrol akses dan enkripsi, untuk melindungi data dari akses yang tidak sah dan menjaga kerahasiaan serta integritasnya.
6. **Pengurangan Biaya:** Desain database yang efisien dapat mengurangi kebutuhan akan sumber daya hardware dan software yang berlebihan, sehingga menekan biaya operasional dan meningkatkan efisiensi biaya.
7. **Dukungan Pengambilan Keputusan:** Struktur database yang baik memungkinkan analisis data yang efektif, mendukung pembuatan laporan dan visualisasi, serta pengambilan keputusan berbasis data yang lebih baik.
8. **Integrasi dan Interoperabilitas:** Desain yang matang memudahkan integrasi database dengan sistem lain, baik internal maupun eksternal, memastikan aliran data yang lancar antara berbagai aplikasi.

Dengan menerapkan prinsip dan teknik perancangan database yang tepat, organisasi dapat memastikan bahwa sistem database mereka berfungsi dengan baik, mendukung tujuan bisnis, dan dapat beradaptasi dengan kebutuhan yang terus berkembang.

C. Database Management System

Database modern dikelola oleh Database Management System (DBMS), yaitu perangkat lunak yang digunakan untuk mengelola, mengatur, dan memanipulasi database. DBMS menyediakan antarmuka untuk pengguna atau aplikasi agar dapat menyimpan, mengambil, dan mengelola data dengan efisien. Contoh DBMS termasuk MySQL, Oracle, SQL Server, dan MongoDB, dan lain-lain.

Beberapa fungsi DBMS yaitu:

1. Menyediakan antarmuka untuk melakukan berbagai operasi seperti pembuatan, penghapusan, modifikasi, dan lain-lain.
2. Memungkinkan pengguna untuk membuat database sesuai kebutuhan.
3. Menerima permintaan dari aplikasi dan menyediakan data spesifik melalui sistem operasi.
4. Berisi kumpulan program yang bertindak sesuai dengan instruksi pengguna.
5. Memberikan keamanan kepada database.

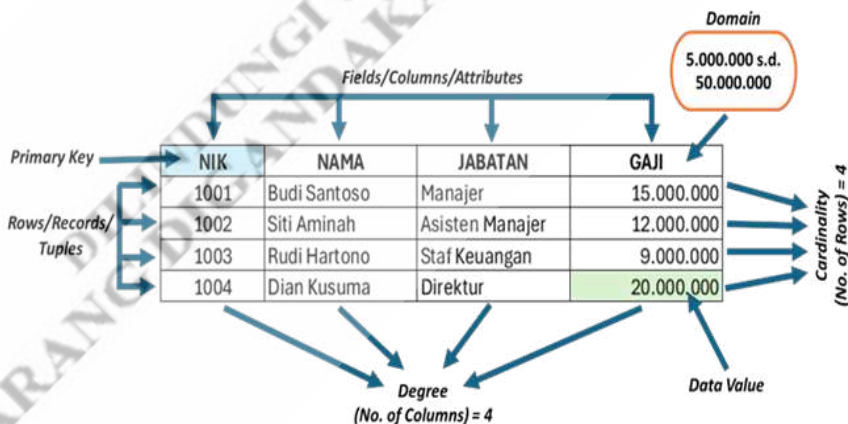
Keuntungan DBMS yaitu:

1. Mengendalikan redundansi, DBMS menyimpan seluruh data dalam satu file database, sehingga dapat mengurangi redundansi data.
2. Berbagi data, pengguna yang memiliki otorisasi dapat membagikan data kepada beberapa pengguna.
3. Backup, DBMS menyediakan sistem backup dan pemulihan. Sistem ini secara otomatis membuat salinan data untuk mengatasi kegagalan sistem dan mengembalikan data jika diperlukan.
4. Berbagai antarmuka pengguna, DBMS menawarkan berbagai jenis antarmuka pengguna seperti GUI dan antarmuka aplikasi.

Relational Database Management System (RDBMS) adalah DBMS berbasis pada model relasional yang dikenalkan oleh E. F. Codd. RDBMS direpresentasikan sebagai tabel yang terdiri dari baris dan kolom. RDBMS terdiri dari beberapa komponen, yaitu:

1. Table.
2. Record/Tuple.
3. Field/Column/Attribute.
4. Instance.
5. Schema.
6. Keys.

Suatu RDBMS merupakan DBMS tabular yang mengelola keamanan (*security*), integritas (*integrity*), akurasi (*accuracy*), and konsistensi (*consistency*) data. Database relasional didasarkan pada model data relasional, yang menyimpan data dalam bentuk baris (*tuple*) dan kolom (*attributes*), yang bersama-sama membentuk sebuah tabel (*relation*).



Gambar 13. Ilustrasi Database Relasional

RDBMS menggunakan tabel untuk penyimpanan data. Tabel adalah kumpulan data terkait yang terdiri dari baris dan kolom. Setiap tabel mewakili objek dunia nyata seperti orang, tempat, atau kejadian yang informasinya dikumpulkan. Karakteristik tabel yaitu:

1. Setiap tabel memiliki nama unik yang mengidentifikasinya dalam database.
2. Tabel tidak boleh memiliki tuple yang duplikat.
3. Tuple dalam tabel tidak memiliki urutan yang spesifik.
4. Semua atribut dalam tabel adalah atomik, artinya setiap sel dalam tabel mengandung satu nilai saja.

Sebuah baris dalam tabel juga dikenal sebagai record atau tuple. Baris ini menyimpan informasi spesifik untuk setiap entri dalam tabel dan merupakan entitas horizontal. Karakteristik dari sebuah baris yaitu:

1. Tidak ada dua tuple yang sama persis dalam semua entri.
2. Semua tuple dalam tabel memiliki format dan jumlah entri yang sama.
3. Urutan tuple tidak penting, karena identifikasi dilakukan berdasarkan isi, bukan posisinya.

Sebuah kolom atau atribut merupakan entitas vertikal dalam tabel yang mencakup semua informasi terkait dengan satu bidang tertentu di dalam tabel. Karakteristik dari sebuah atribut yaitu:

1. Setiap atribut dalam sebuah relasi harus memiliki nama.
2. Nilai null diperbolehkan untuk atribut.
3. Nilai default dapat ditentukan untuk sebuah atribut dan akan dimasukkan secara otomatis jika tidak ada nilai lain yang diberikan untuk atribut tersebut.
4. Atribut yang secara unik mengidentifikasi setiap tuple dalam sebuah relasi disebut primary key.

Unit data terkecil dalam tabel adalah item/cell data individu, yang disimpan di pertemuan antara tuple dan atribut. Karakteristik dari cell data yaitu:

1. Cell data bersifat atomik.

2. Cell data untuk suatu atribut harus berasal dari domain yang sama.

Jumlah total atribut yang terdapat dalam sebuah tabel disebut sebagai derajat (*degree*) tabel. Jumlah total tuple dalam sebuah tabel pada suatu waktu disebut kardinalitas (*cardinality*) tabel. Jika sebuah tabel memiliki kardinalitas 0, maka itu disebut tabel kosong.

Domain mengacu pada nilai-nilai yang dapat dimiliki oleh setiap atribut. Domain ini bisa ditentukan menggunakan tipe data standar seperti integer, floating numbers, dan lainnya. Sebagai contoh, atribut dengan nama Marital_Status mungkin hanya dibatasi pada nilai menikah atau belum menikah. Nilai NULL dalam tabel menunjukkan bahwa kolom tersebut dibiarkan kosong saat record dibuat. Nilai ini berbeda dari nilai nol atau kolom yang berisi spasi.

Ada beberapa kategori integritas data yang berlaku dalam RDBMS:

1. **Entity integrity:** Menentukan bahwa tidak boleh ada baris yang duplikat dalam sebuah tabel.
2. **Domain integrity:** Menjamin entri yang valid untuk kolom tertentu dengan membatasi tipe, format, atau rentang nilai.
3. **Referential integrity:** Menyatakan bahwa baris yang sedang digunakan oleh record lain tidak boleh dihapus.
4. **User-defined integrity:** Menegakkan aturan bisnis tertentu yang ditentukan oleh pengguna. Aturan ini berbeda dari integritas entitas, domain, atau referensial.

DBMS mengelola data agar tetap terintegrasi meskipun terjadi perubahan. Ini penting karena gangguan pada integritas data dapat menyebabkan kerusakan dan kekacauan pada seluruh data. Untuk menjaga integritas data, terdapat empat karakteristik yang dijelaskan dalam sistem manajemen database, yang dikenal sebagai karakteristik ACID, yaitu:

1. **Atomicity (A):** Memastikan bahwa operasi data akan berhasil sepenuhnya atau gagal secara keseluruhan. Konsep ini mengikuti prinsip 'semua atau tidak sama sekali'. Sebagai

contoh, sebuah transaksi akan dilaksanakan sepenuhnya atau dibatalkan.

2. **Consistency (C):** Setelah melakukan operasi pada data, nilai data sebelum dan sesudah operasi harus tetap konsisten. Contohnya, saldo rekening harus benar dan tetap sama sebelum dan sesudah transaksi.
3. **Isolation (I):** Beberapa pengguna bisa mengakses data secara bersamaan dari database. Oleh karena itu, data harus tetap terisolasi antar transaksi. Misalnya, efek dari satu transaksi tidak boleh mempengaruhi transaksi lain yang terjadi secara bersamaan.
4. **Durability (D):** Ini memastikan bahwa setelah operasi selesai dan data disimpan, perubahan yang dilakukan akan bersifat permanen.

Berikut adalah contoh yang menunjukkan penerapan prinsip ACID dalam sistem manajemen database (DBMS) pada skenario transfer dana antar akun sebagai berikut:

Aldi ingin mentransfer \$500 dari akun tabungannya ke akun tabungan Reza. Proses ini melibatkan beberapa langkah yang harus dijalankan sebagai satu transaksi untuk memastikan integritas data.

1. **Atomicity:** Jika sistem gagal setelah pengurangan \$500 dari akun Aldi tetapi sebelum penambahan \$500 ke akun Reza, sistem harus membatalkan seluruh transaksi sehingga saldo Aldi tetap \$2000 dan saldo Reza tetap \$3000.
2. **Consistency:** Setelah transaksi berhasil, saldo akhir Aldi dan Reza harus sesuai dengan nilai yang diharapkan, yaitu saldo Aldi menjadi \$1500 dan saldo Reza menjadi \$3500.
3. **Isolation:** Selama transaksi transfer berlangsung, perubahan saldo akun Aldi tidak boleh terlihat oleh transaksi lain yang membaca atau menulis data saldo yang sama sampai transaksi transfer selesai.
4. **Durability:** Setelah transaksi selesai dan dikomit, saldo akhir yang baru harus tetap ada meskipun sistem mengalami crash

atau gangguan lainnya.

Skenario ini menunjukkan bagaimana prinsip-prinsip ACID diterapkan untuk memastikan transaksi database dilakukan dengan cara yang aman dan konsisten.

Ekstraksi data menggunakan teknik web scraping, yaitu teknik yang digunakan untuk mengekstrak data dari situs web yang tidak terstruktur serta mengubah data tersebut menjadi terstruktur agar mudah dipahami seperti spreadsheet, database, atau file comma-separated values (CSV). Web scraping merupakan proses penggalan data atau informasi dari situs web dan digunakan dalam proses pengambilan data secara otomatis. Dokumen pada situs web harus dipelajari terlebih dahulu sebelum melakukan web scraping. Hasil dari web scraping akan disimpan dan diagendakan untuk otomatisasi perintah pembaruan data agar data yang dihasilkan selalu terkini (Adila N.,2022).

D. Model Data

Model adalah gambaran atau pola dari sesuatu yang ada atau akan dibuat. Model dapat mencakup satu atau lebih diagram. Diagram model menggunakan simbol standar yang membantu orang memahami isinya. Peta, bagan organisasi, dan denah bangunan adalah contoh model yang sering digunakan. Model data menggambarkan data suatu organisasi sesuai dengan pemahaman organisasi tersebut, atau sesuai dengan keinginan organisasi. Model data ini berisi serangkaian simbol dan label teks yang berusaha merepresentasikan kebutuhan data secara visual seperti yang disampaikan kepada data modeler. Model ini berfungsi sebagai dokumentasi untuk kebutuhan dan definisi data yang dihasilkan dari proses pemodelan. Model data adalah alat utama untuk menyampaikan kebutuhan data, hampir semua model data memiliki elemen dasar yang sama, yaitu entitas, atribut, relasi, dan domain.

ER model adalah singkatan dari Entity-Relationship model, yang merupakan data model tingkat tinggi. Model ini digunakan untuk mendefinisikan elemen-elemen data dan relationship dalam suatu sistem tertentu. ER model membantu mengembangkan desain

konseptual untuk database. Model ini juga menyajikan pandangan data yang sangat sederhana dan mudah untuk dirancang. Dalam ER modeling, struktur database diilustrasikan melalui sebuah diagram yang disebut entity-relationship diagram (ER Diagram). ER Diagram mengilustrasikan hubungan antar entitas (tabel) serta hubungan primary key-foreign key antar tabel (relationship). Hubungan primary key-foreign key menunjukkan keterkaitan atau hubungan antara entitas, serta mengidentifikasi atribut dan key yang menghubungkannya.

1. Entitas

Dalam pemodelan data, entitas adalah objek yang dikumpulkan informasinya oleh sebuah organisasi. Entitas sering kali dianggap sebagai kata benda dalam konteks organisasi. Sebuah entitas dapat berupa objek, kelas, individu, atau lokasi. Dalam ER diagram, entitas direpresentasikan dengan persegi panjang. Sebagai contoh entitas yaitu pelanggan, produk, penjualan, karyawan, departemen, dan lain-lain.

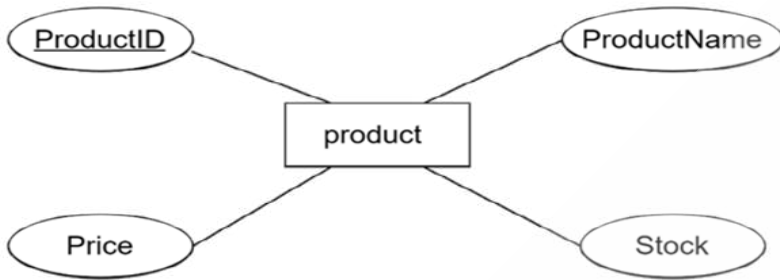


Gambar 14. Ilustrasi Entitas

Gambar 14 di atas merepresentasikan relasi antara entitas Customer dan Sale, biasanya digambarkan sebagai hubungan yang menggambarkan bahwa seorang pelanggan (Customer) dapat melakukan satu atau lebih transaksi penjualan (Sale).

2. Atribut

Atribut adalah properti yang digunakan untuk menjelaskan, mengidentifikasi, atau mengukur sebuah entitas dan dapat memiliki domain. Koresponden fisik dari sebuah atribut dalam entitas adalah kolom atau field yang ada dalam tabel. Atribut direpresentasikan dengan elips. Sebagai contoh dalam merancang database retail, product merupakan entitas yang memiliki atribut seperti ProductID, ProductName, Price, Stock.

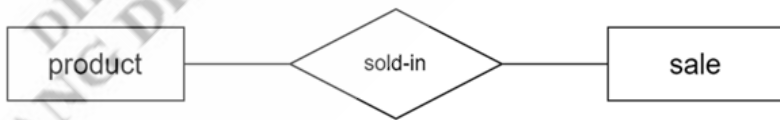


Gambar 15. Contoh Entitas Product

Atribut ProductID digaris bawah menandakan bahwa atribut tersebut merupakan primary key. Primary key (PK) adalah komponen kunci dalam desain database relasional yang berfungsi untuk memastikan bahwa setiap entri dalam tabel memiliki identifikasi yang unik. Sedangkan Foreign key (FK) adalah kolom atau kumpulan kolom dalam satu tabel yang merujuk ke kolom atau kumpulan kolom (biasanya primary key) di tabel lain untuk membentuk integritas referensial dalam database.

3. Relasi

Relasi (*relationship*) adalah keterkaitan antara entitas. Relationship mencerminkan interaksi antar entitas. Relasi direpresentasikan dengan diamond atau belah ketupat. Sebagai contoh dalam merancang database retail, relasi sold-in merupakan relasi antara entitas product dengan entitas sale.

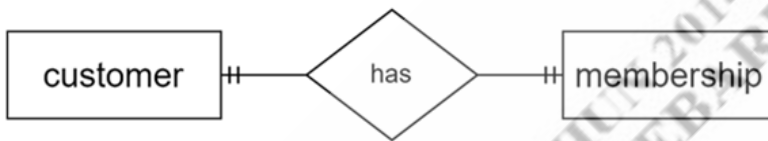


Gambar 16. Contoh Relasi Sold-in

Derajat relationship dapat dijelaskan sebagai jumlah kejadian dalam satu entitas yang berhubungan dengan jumlah kejadian dalam entitas lainnya. Terdapat tiga jenis derajat relationship, yaitu one-to-one (1:1), one-to-many (1:M), dan many-to-many (M:M).

a. One-to-one (1:1)

Dalam relationship one-to-one, setiap kejadian dari satu entitas terhubung hanya dengan satu kejadian di entitas lainnya. Contohnya, seorang pelanggan (*customer*) hanya boleh memiliki satu keanggotaan (*membership*), maka keanggotaan (*membership*) tersebut hanya boleh untuk satu pelanggan (*customer*). Jadi, *customer* dan *membership* memiliki relationship one-to-one.



Gambar 17. Relasi 1:1

b. One-to-many (1:M)

Dalam relationship one-to-many, satu kejadian pada sebuah entitas terhubung dengan banyak kejadian di entitas lain. Contohnya, satu transaksi penjualan (*sale*) dilakukan hanya untuk seorang pelanggan (*customer*), tetapi satu pelanggan (*customer*) dapat melakukan banyak transaksi penjualan (*sale*). Jadi, *customer* dan *sale* memiliki relationship one-to-many.

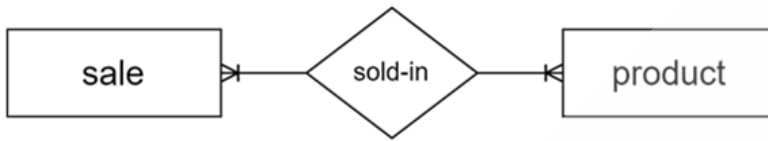


Gambar 18. Relasi 1:M

c. Many-to-many (M:M)

Dalam relationship many-to-many, beberapa kejadian dalam satu entitas terhubung dengan beberapa kejadian di entitas lain. Contohnya, satu produk (*product*) dapat dijual pada banyak transaksi penjualan (*sale*), dan pada satu transaksi penjualan (*sale*) bisa terdapat banyak produk

(*product*) yang dijual. Jadi, *product* dan *sale* memiliki relationship many-to-many.



Gambar 19. Relasi M:M

4. Domain

Dalam pemodelan data, domain adalah kumpulan lengkap nilai yang dapat diberikan kepada sebuah atribut. Domain berfungsi untuk menstandarisasi karakteristik atribut. Misalnya, domain Date, yang mencakup semua tanggal valid yang mungkin, dapat diterapkan pada atribut tanggal dalam Logical Data Model atau kolom/field tanggal dalam Physical Data Model, seperti:

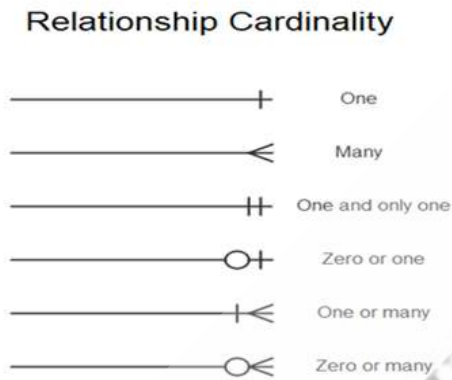
- BirthDate: Menyimpan tanggal lahir karyawan.
- HireDate: Menyimpan tanggal ketika karyawan dipekerjakan.
- ExpirationDate: Tanggal kadaluwarsa suatu item, seperti kontrak, lisensi, atau produk.
- DueDate: Tanggal jatuh tempo untuk pembayaran atau penyelesaian suatu tugas.
- DeliveryDate: Tanggal ketika pengiriman barang diharapkan tiba.

Suatu atribut seharusnya tidak memiliki nilai di luar domain yang telah ditetapkan. Sebagai contoh, EmployeeGenderCode mungkin dibatasi pada domain perempuan dan laki-laki.

5. Notasi ER Diagram

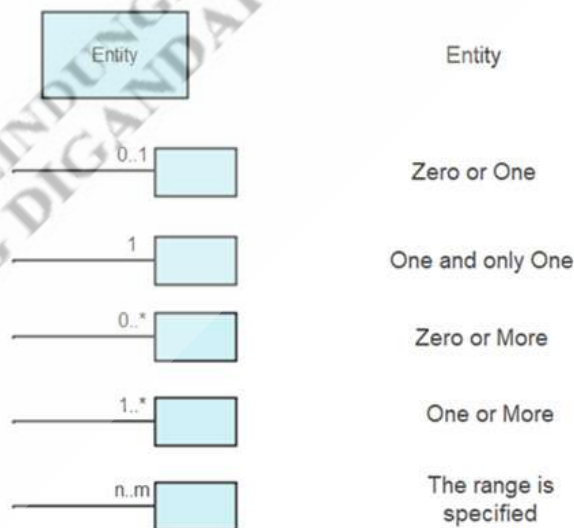
Database dapat digambarkan menggunakan notasi. Dalam ER diagram, berbagai notasi digunakan untuk menunjukkan kardinalitas. Gordon Everest adalah orang yang menciptakan notasi Crow's Foot. Simbol-simbol ER diagram yang digunakan dalam

dapat dilihat pada Gambar 20.



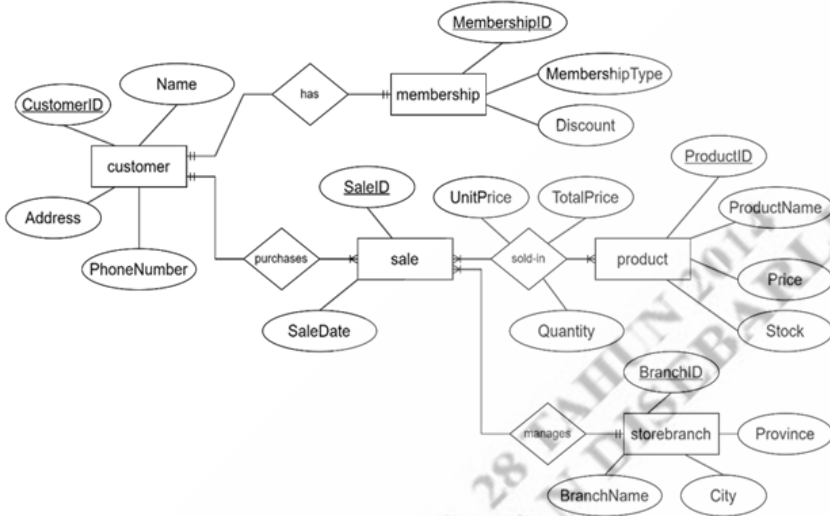
Gambar 20. Notasi Crow's Foot (Edraw Content Team, 2020)

UML (*Unified Modeling Language*) terkenal karena kemampuannya dalam mendokumentasikan, memvisualisasikan, dan menentukan komponen sistem perangkat lunak dan non-perangkat lunak. Simbol-simbol ER diagram yang digunakan dalam notasi UML dapat dilihat pada Gambar 21.



Gambar 21. Notasi UML (Edraw Content Team, 2020)

Sebagai contoh representasi database menggunakan ER diagram dapat dilihat pada Gambar 22.



Gambar 22. Contoh ER Diagram

E. Pembuatan Database

Software RDBMS (*Relational Database Management System*) adalah aplikasi yang dibuat untuk mengelola dan mengatur data dalam format tabel yang saling terhubung. RDBMS mematuhi model relasional, yang mengorganisasi data dalam tabel yang terdiri dari baris dan kolom. MySQL adalah software RDBMS open-source yang banyak digunakan untuk menyimpan dan mengelola data. Berikut adalah cara MySQL digunakan dalam penerapan database:

1. Instalasi dan Konfigurasi

MySQL dapat diinstal pada berbagai platform seperti Windows, Linux, dan macOS, dengan mengunduhnya dari situs resmi <https://www.mysql.com> atau menggunakan manajer paket seperti XAMPP (<https://www.apachefriends.org>). Setelah instalasi, konfigurasi diperlukan, termasuk pengaturan port, password root, dan opsi keamanan dasar.

2. Pembuatan Database

Pembuatan database menggunakan perintah SQL seperti ``CREATE DATABASE nama_database;``. Database berfungsi sebagai wadah untuk tabel-tabel. Pembuatan tabel dalam database dengan perintah ``CREATE TABLE nama_tabel (...);``.

3. Desain Skema

Definisikan entitas (misalnya, pelanggan, produk) dan relationship (misalnya, transaksi pembelian) dalam bentuk tabel, dengan primary key dan foreign key untuk mengatur hubungan antar tabel. Terapkan normalisasi untuk menghindari redundansi dan menjaga integritas data.

4. Manipulasi Data

Gunakan perintah Data Definition Language (DDL) seperti ``ALTER TABLE`` untuk memodifikasi struktur tabel, menambah kolom, atau mengubah tipe data serta gunakan perintah Data Manipulation Language (DML) untuk menambah, mengubah, dan menghapus data dengan perintah ``INSERT``, ``UPDATE``, dan ``DELETE``, misalnya ``INSERT INTO nama_tabel (kolom1, kolom2) VALUES (nilai1, nilai2);``.

5. Query dan Pengambilan Data

Query untuk mengambil data dari tabel menggunakan perintah ``SELECT``, dan juga dapat menambahkan klausa seperti ``WHERE``, ``JOIN``, ``GROUP BY``, dan ``ORDER BY`` untuk menyaring dan mengatur data. Untuk meningkatkan performa query dengan membuat indeks pada kolom yang sering digunakan untuk pencarian atau pengurutan.

6. Keamanan dan Pengelolaan Pengguna

Atur pengguna dan hak akses dengan perintah SQL seperti ``CREATE USER``, ``GRANT``, dan ``REVOKE`` untuk memastikan akses data hanya untuk pengguna yang berwenang. Lakukan backup dan restore database menggunakan alat seperti ``mysqldump`` untuk memastikan data dapat dipulihkan jika terjadi kegagalan.

7. Optimasi dan Pemeliharaan

Optimalkan performa database dengan teknik seperti tuning query, penggunaan indeks yang tepat, dan pengaturan konfigurasi MySQL. Lakukan pemeliharaan rutin, termasuk pemeriksaan integritas tabel, pembersihan data, dan pembaruan versi MySQL.

8. Integrasi dan Akses:

Integrasikan MySQL dengan berbagai aplikasi dan bahasa pemrograman seperti PHP, Python, dan Java untuk interaksi database. Gunakan alat untuk desain visual, pengelolaan, dan query database seperti MySQL Workbench, DrawSQL, SQLyog, dan lain-lain.

Dengan mengikuti langkah-langkah ini dapat menerapkan dan mengelola database menggunakan MySQL dengan efektif.

Tahapan perancangan database relasional mencakup langkah-langkah berikut untuk memastikan database yang dibangun efisien dan sesuai kebutuhan:

- a. **Identifikasi Kebutuhan:** Mengumpulkan informasi dan menganalisis kebutuhan pengguna untuk menentukan data yang perlu disimpan dan cara penggunaannya. Identifikasi Kebutuhan (Setiyani L, *et.al.*, 2020)

Diagram PIECES merupakan kerangka yang dipakai untuk mengkalsifikasikan suatu permasalahan, opportunities dan directives yang terdapat pada bagian scope definition analisa dan perancangan sistem. Kerangka ini dapat digunakan untuk menghasilkan hal-hal baru yang dapat menjadi pertimbangan dalam pengembangan sistem informasi. PIECES Framework terdiri dari enam dimensi yaitu Perfomance, information, economy, control, efficiency, service).

- 1) *Performance* (Perfomansi)

Performansi merupakan suatu kemampuan sistem dalam menyelesaikan tugas dengan cepat sehingga

sasaran dapat segera tercapai. Indikator-indikator dari performansi ini adalah hasil, waktu respon, audibilitas, kelaziman komunikasi, kelengkapan, konsistensi dan toleransi kesalahan.

2) *Information* (Informasi)

Informasi adalah kemampuan sistem dalam memberikan informasi yang akurat, tepat waktu dan relevan sesuai dengan yang di harapkan. Indikator indikator dari information adalah accuracy, relevansi informasi, penyajian informasi dan fleksibilitas data.

3) *Economic*

Economic merupakan pemanfaatan biaya yang digunakan dari pemanfaatan informasi tersebut. peningkatan terhadap kebutuhan informasi yang ekonomis dapat mempengaruhi pengendalian biaya dan peningkatan manfaat terhadap sistem informasi. Indikator-indikator dari economic adalah reusabilitas dan sumber daya.

4) *Control*

Control merupakan pembadingan sistem dari segi integritas , kemudahan akses dan keamanan data.

5) *Efficiency*

Dimensi ini berhubungan dengan bagaimana sumber tersebut dapat digunakan secara optimal. Operasional suatu perusahaan dapat dikatan efisien atau tidak biasanya didasarkan pada tugas dan tanggung jawab dalam melaksanakan kegiatan. Indikator indikator dari efisiensi adalah usabilitas dan maintabilitas

6) *Service*

Service merupakan kemampuan dalam peningkatan layanan yang lebih baik bagi manajemen, pengguna dan bagian lainnya yang merupakan simbol kualitas

dari sistem. Indikator –indikator dari service adalah akurasi, reliabilitas, dan kesederhanaan

- b. **Analisis Konseptual:** Mengembangkan model konseptual dari database yang mencakup identifikasi entitas, atribut, dan relasi menggunakan diagram seperti Entity-Relationship Diagram (ERD).
- c. **Desain Logis:** Mengubah model konseptual menjadi model logis dengan menyusun entitas, atribut, dan relasi menjadi tabel, kolom, dan foreign key dalam model relasional.
- d. **Desain Fisik:** Mendesain implementasi fisik database pada sistem manajemen basis data (DBMS), termasuk pengaturan indeks, partisi, dan aspek lain untuk mengoptimalkan performa.
- e. **Implementasi:** Mewujudkan desain ke dalam database nyata menggunakan alat dan teknologi DBMS, termasuk pembuatan tabel, pengaturan relasi, dan pengisian data awal.

Untuk lebih memahami perancangan database relasional, berikut ini contoh perancangan database relasional dengan studi kasus sistem penjualan pada jaringan toko elektronik.

F. Deskripsi Kasus

Suatu perusahaan retail memiliki jaringan toko yang menjual berbagai produk elektronik dan menawarkan program membership kepada pelanggan. Sistem manajemen penjualan mereka perlu mengelola data pelanggan, membership, produk, penjualan, detail penjualan, serta informasi cabang toko yang mencakup kota dan provinsi.

Untuk menyelesaikan kasus tersebut dilakukan perancangan database relasional dengan langkah-langkah sebagai berikut:

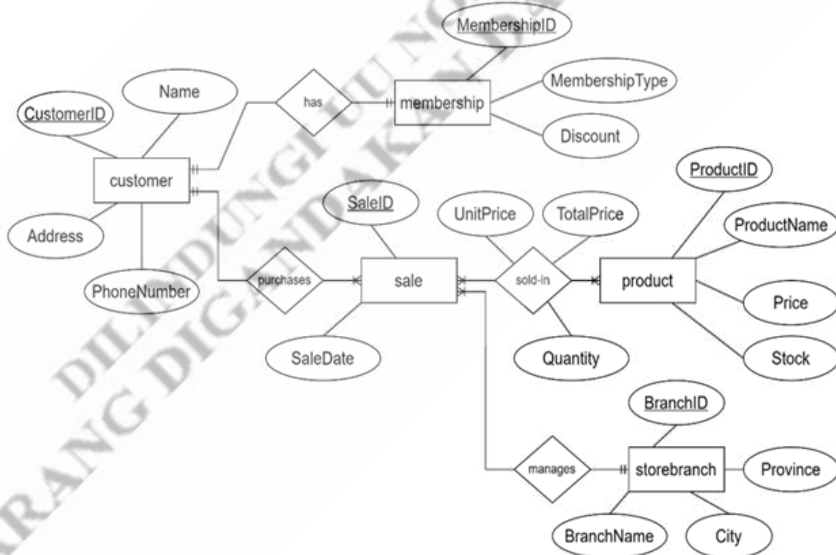
1. Identifikasi Kebutuhan

Diperlukan database untuk mengelola data pelanggan dan

membership, data produk, data transaksi penjualan, data cabang toko sehingga dapat merancang skema database yang terstruktur dan efisien untuk mendukung operasi dan laporan sistem manajemen penjualan perusahaan retail, serta memastikan bahwa database tidak hanya berfungsi dengan baik dalam hal pengelolaan data, tetapi juga mendukung kebutuhan bisnis secara keseluruhan, meningkatkan keamanan, dan memberikan fleksibilitas untuk penyesuaian di masa depan.

2. Analisis Konseptual

Analisis konseptual dalam perancangan database adalah mengembangkan model konseptual yang mencakup identifikasi entitas, atribut, dan relasi. Model konseptual ini sering digambarkan menggunakan diagram Entity-Relationship (ER Diagram). Gambar 23 adalah ER Diagram untuk kasus sistem penjualan ini.



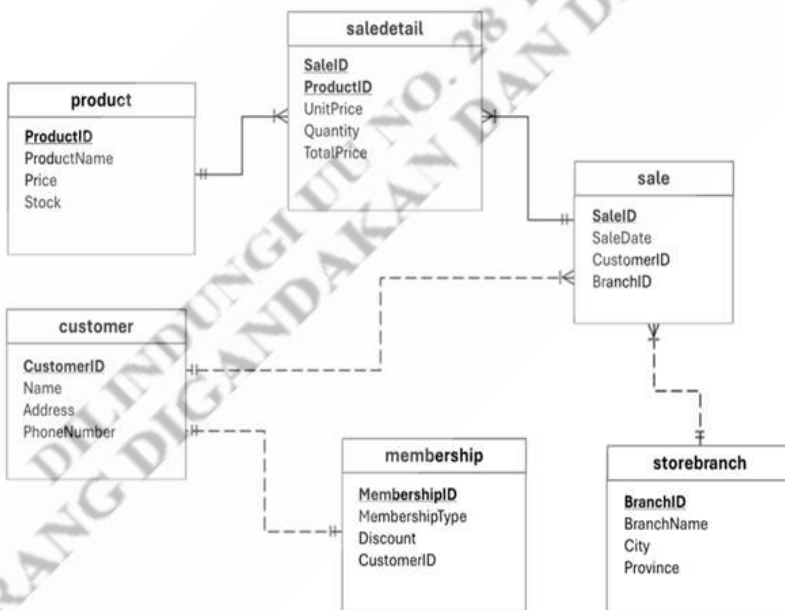
Gambar 23. ER Diagram Sistem Penjualan

3. Desain Logis

Dalam desain logis, model konseptual yang dikembangkan diubah menjadi model logis dengan menyusun entitas, atribut,

dan relasi menjadi tabel, kolom, dan foreign key dalam model relasional. Relasi many-to-many dalam ERD diubah menjadi dua relasi one-to-many dalam desain logis dengan menggunakan tabel *junction* (penghubung). Hal ini dilakukan karena database relasional tidak dapat menangani relasi many-to-many secara langsung. Tabel junction menghubungkan dua entitas menggunakan foreign key dan menyimpan pasangan relasi antara entitas tersebut. Pendekatan ini memastikan integritas data dan menghindari duplikasi informasi, serta memungkinkan pengelolaan relasi yang lebih efisien.

Pada kasus sistem penjualan ini entitas product dan entitas sale memiliki relationship many-to-many, maka digunakan tabel junction bernama saledetail. Gambar 24 adalah diagram logis untuk kasus ini.



Gambar 24. Diagram Logis Sistem Penjualan

4. Desain Fisik

Desain fisik database adalah tahap di mana model logis yang telah dikembangkan diubah menjadi implementasi nyata dalam

sistem manajemen basis data (DBMS). Pada tahap ini, fokusnya adalah pada pengaturan yang mendetail untuk memastikan bahwa database berfungsi secara optimal dan efisien. Dalam SQL, tipe data (datatype) menentukan jenis nilai yang dapat disimpan dalam kolom tabel. Memilih tipe data yang tepat sangat penting untuk efisiensi penyimpanan dan kinerja database. Pada tabel berikut ini adalah penjelasan beberapa tipe data umum dalam SQL:

Tabel 3. Tipe Data SQL

Kategori	Tipe Data	Deskripsi
Numerik	INT	Menyimpan angka bulat. Terdapat variasi seperti TINYINT, SMALLINT, MEDIUMINT, BIGINT.
	FLOAT	Menyimpan angka desimal dengan presisi rendah.
	DOUBLE	Menyimpan angka desimal dengan presisi tinggi.
	DECIMAL	Menyimpan angka desimal dengan presisi tetap, cocok untuk keuangan.
Karakter	CHAR(n)	Menyimpan teks dengan panjang tetap, mengisi sisa dengan spasi jika lebih pendek.
	VARCHAR(n)	Menyimpan teks dengan panjang variabel hingga batas n.
	TEXT	Menyimpan teks panjang, cocok untuk artikel atau deskripsi panjang.

Tanggal dan Waktu	DATE	Menyimpan tanggal dalam format YYYY-MM-DD.
	TIME	Menyimpan waktu dalam format HH:MM:SS.
	DATETIME	Menyimpan kombinasi tanggal dan waktu dalam format YYYY-MM-DD HH:MM:SS.
	TIMESTAMP	Menyimpan tanggal dan waktu dengan format cap waktu UNIX.
	YEAR	Menyimpan tahun dalam format dua atau empat digit.
Biner	BINARY(n)	Menyimpan data biner dengan panjang tetap.
	VARBINARY(n)	Menyimpan data biner dengan panjang variabel.
Boolean	BOOLEAN	Menyimpan nilai TRUE atau FALSE, biasanya sebagai TINYINT (1 untuk TRUE, 0 untuk FALSE).

DrawSQL (<https://drawsql.app>) adalah aplikasi berbasis web yang dirancang untuk merancang dan mendokumentasikan skema database relasional. Alat ini memfasilitasi pembuatan desain fisik database secara visual dengan cara yang mudah dan efisien dan mendukung pembuatan diagram ER (Entity-Relationship) yang krusial dalam perancangan dan pengelolaan database relasional. Fitur utama DrawSQL yaitu:

- Antarmuka Visual: DrawSQL menyediakan antarmuka grafis yang mempermudah pembuatan dan modifikasi diagram skema database, termasuk menambah, menghapus, dan mengedit tabel serta relasi.
- Pembuatan Tabel: Pengguna dapat mendefinisikan tabel dengan menetapkan kolom, tipe data, kunci utama (*primary key*), dan kunci asing (*foreign key*).

- c. Relasi dan Indeks: Memungkinkan penyusunan relasi antar tabel dan penambahan indeks untuk meningkatkan kinerja query.
- d. Ekspor dan Integrasi: Diagram yang dibuat dapat diekspor dalam format gambar atau dokumen serta diintegrasikan dengan alat lain seperti sistem manajemen basis data atau alat dokumentasi.

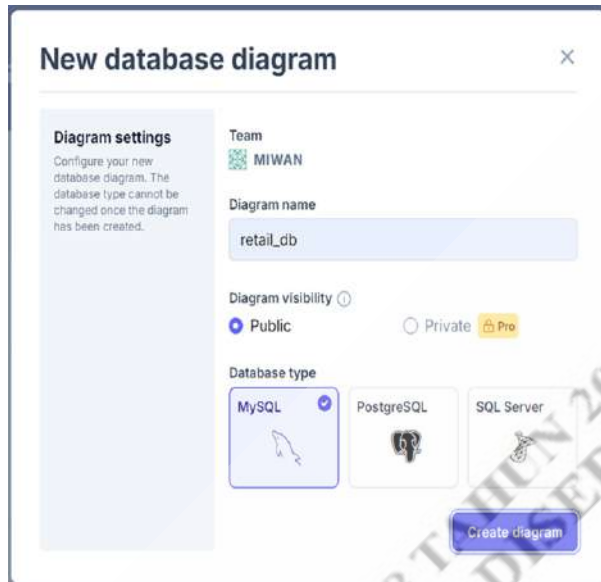
Berikut ini adalah langkah-langkah merancang dan mendokumentasikan skema database relasional menggunakan DrawSQL sehingga menghasilkan desain fisik database:

1. Mengakses aplikasi DrawSQL melalui link <https://drawsql.app>, silahkan register atau login untuk menggunakan aplikasi. Setelah berhasil login akan tampil halaman utama dan tekan tombol **New Diagram** untuk mulai membuat diagram.



Gambar 25. Halaman Utama DrawSQL

2. Mengisi nama diagram dengan nama retail_db dan memilih tipe database MySQL, kemudian tekan tombol **Create diagram** dan akan tampil jendela antarmuka untuk membuat diagram.



Gambar 26. Mengisi Nama Diagram dan Tipe Database

3. Untuk membuat tabel baru dapat dilakukan dengan cara menekan tombol **New Table** pada jendela antarmuka.

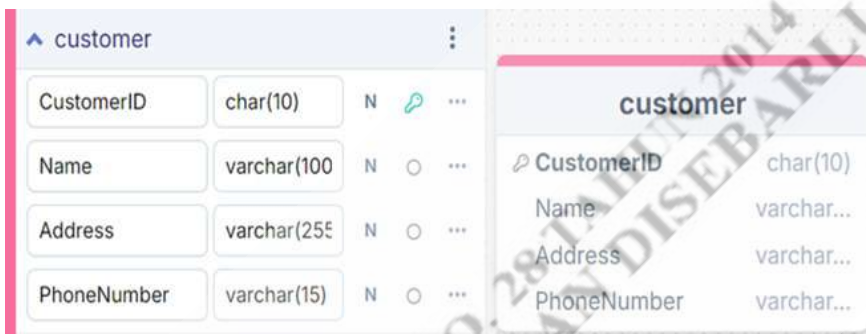


Gambar 27. Jendela Antarmuka DrawSQL

Buatlah tabel customer dengan spesifikasi sebagai berikut:

<u>CustomerID</u>	char(10) NOT NULL
Name	varchar(100) NOT NULL
Address	varchar(255) NOT NULL
PhoneNumber	varchar(15) NOT NULL

Gambar 28 adalah tampilan desain tabel customer di DrawSQL.



Gambar 28. Desain Tabel Customer di DrawSQL

Ulangi langkah pembuatan tabel untuk membuat tabel-tabel selanjutnya dengan spesifikasi sebagai berikut:

Tabel membership

MembershipID	char(10) NOT NULL
MembershipType	enum('Silver','Gold','Platinum') NOT NULL
Discount	decimal(5,2) NOT NULL
CustomerID	char(10) NOT NULL

Tabel product

ProductID	char(10) NOT NULL
ProductName	varchar(100) NOT NULL
Price	decimal(10,2) NOT NULL
Stock	int(11) NOT NULL

Tabel storebranch

BranchID	char(10) NOT NULL
BranchName	varchar(100) NOT NULL
City	varchar(100) NOT NULL
Province	varchar(100) NOT NULL

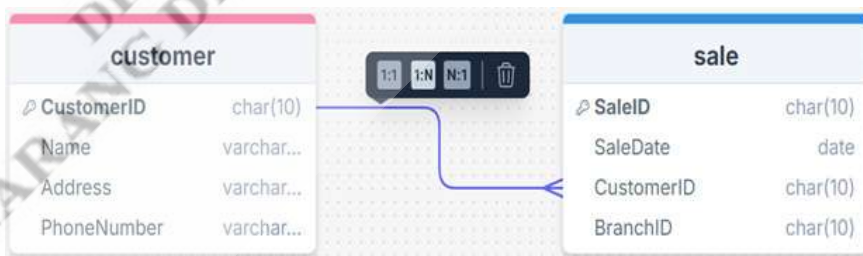
Tabel sale

SaleID	char(10) NOT NULL
SaleDate	date NOT NULL
CustomerID	char(10) NOT NULL
BranchID	char(10) NOT NULL

Tabel saledetail

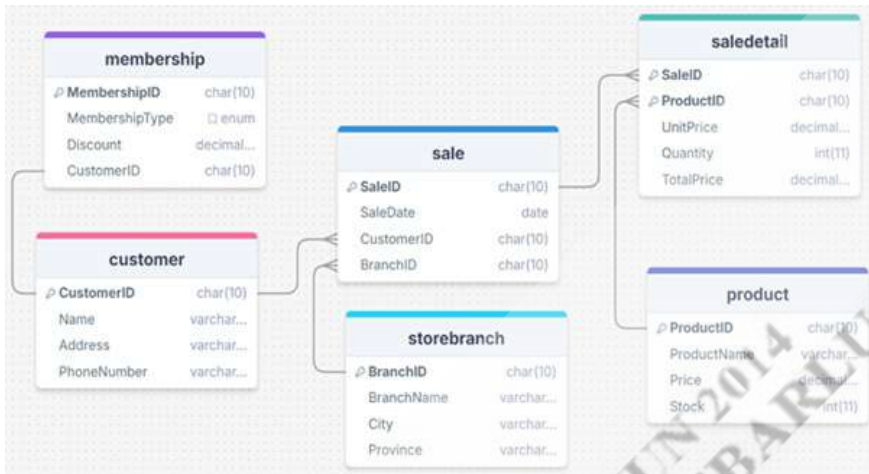
SaleID	char(10) NOT NULL
ProductID	char(10) NOT NULL
UnitPrice	decimal(10,2) NOT NULL
Quantity	int(11) NOT NULL
TotalPrice	decimal(10,2) NOT NULL

Setelah semua tabel terbentuk, selanjutnya membuat integritas referensial antar tabel dengan menarik garis dari Primary Key suatu tabel ke Foreign Key di tabel lain dan menentukan derajat relasinya. Seluruh Foreign Key ditambahkan sebagai indeks untuk meningkatkan kinerja query.



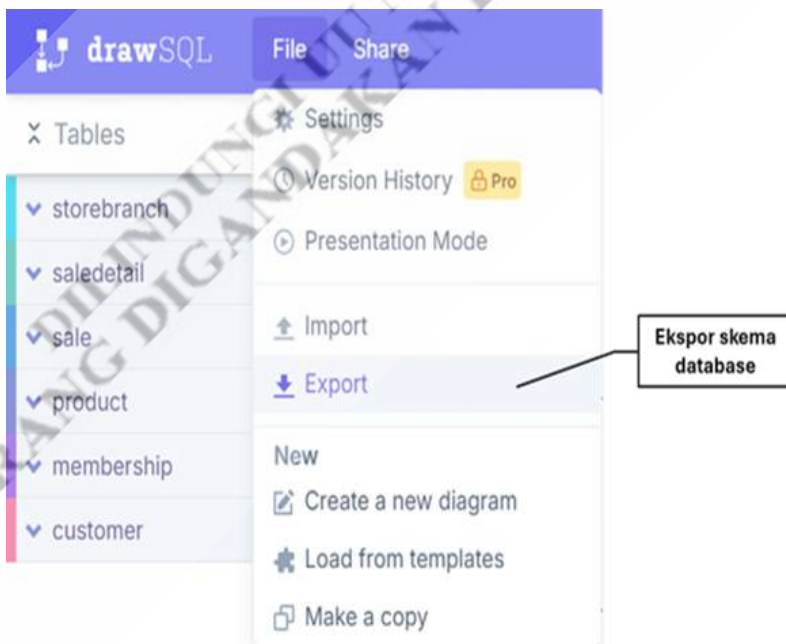
Gambar 29. Integritas Referensial di DrawSQL

Gambar 30 adalah diagram fisik untuk kasus sistem penjualan.



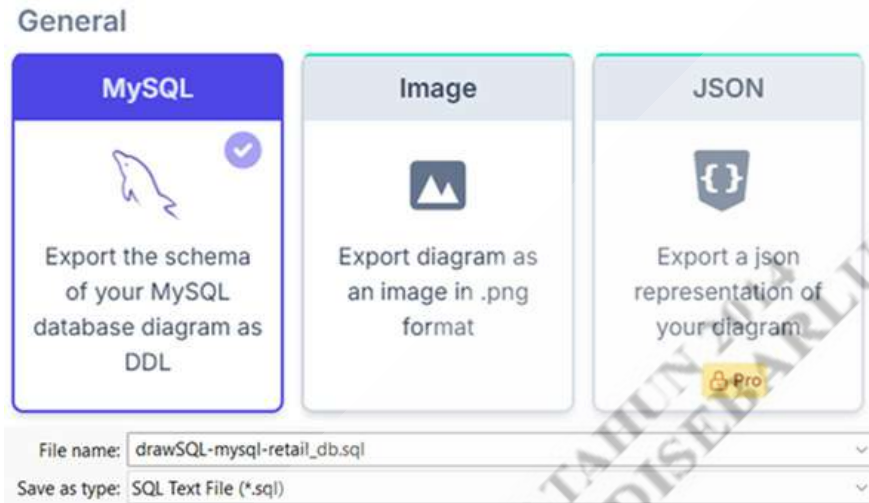
Gambar 30. Diagram Fisik Sistem Penjualan

Setelah semua diagram fisik terbentuk, selanjutnya melakukan ekspor skema dari diagram fisik dalam bentuk skrip SQL (DDL) melalui menu **File - Export**.



Gambar 31. Menu Ekspor Skema Menjadi Skrip SQL

Pilih **General** - **MySQL** dan ekspor/simpan dalam file SQL.



Gambar 32. Ekspor Skema Menjadi Skrip SQL

File skrip SQL yang terbentuk dapat dibuka menggunakan aplikasi text editor, seperti aplikasi Notepad. Berikut ini skrip SQL yang dihasilkan:

```

CREATE TABLE storebranch(
    BranchID CHAR(10) NOT NULL,
    BranchName VARCHAR(100) NOT NULL,
    City VARCHAR(100) NOT NULL,
    Province VARCHAR(100) NOT NULL,
    PRIMARY KEY(BranchID)
);

CREATE TABLE customer(
    CustomerID CHAR(10) NOT NULL,
    Name VARCHAR(100) NOT NULL,
    Address VARCHAR(255) NOT NULL,
    PhoneNumber VARCHAR(15) NOT NULL,
    PRIMARY KEY(CustomerID)
);

CREATE TABLE product(
    ProductID CHAR(10) NOT NULL,
    ProductName VARCHAR(100) NOT NULL,
    Price DECIMAL(10, 2) NOT NULL,
    Stock INT NOT NULL,
    PRIMARY KEY(ProductID)
);

CREATE TABLE sale(
    SaleID CHAR(10) NOT NULL,
    SaleDate DATE NOT NULL,
    CustomerID CHAR(10) NOT NULL,
    BranchID CHAR(10) NOT NULL,
    PRIMARY KEY(SaleID)
);

CREATE TABLE saledetail(
    SaleID CHAR(10) NOT NULL,
    ProductID CHAR(10) NOT NULL,
    UnitPrice DECIMAL(10, 2) NOT NULL,
    Quantity INT NOT NULL,
    TotalPrice DECIMAL(10, 2) NOT NULL,
    PRIMARY KEY(SaleID, ProductID)
);

CREATE TABLE membership(
    MembershipID CHAR(10) NOT NULL,
    MembershipType ENUM('Silver', 'Gold', 'Platinum') NOT NULL,
    Discount DECIMAL(5, 2) NOT NULL,
    CustomerID CHAR(10) NOT NULL,
    PRIMARY KEY(MembershipID)
);

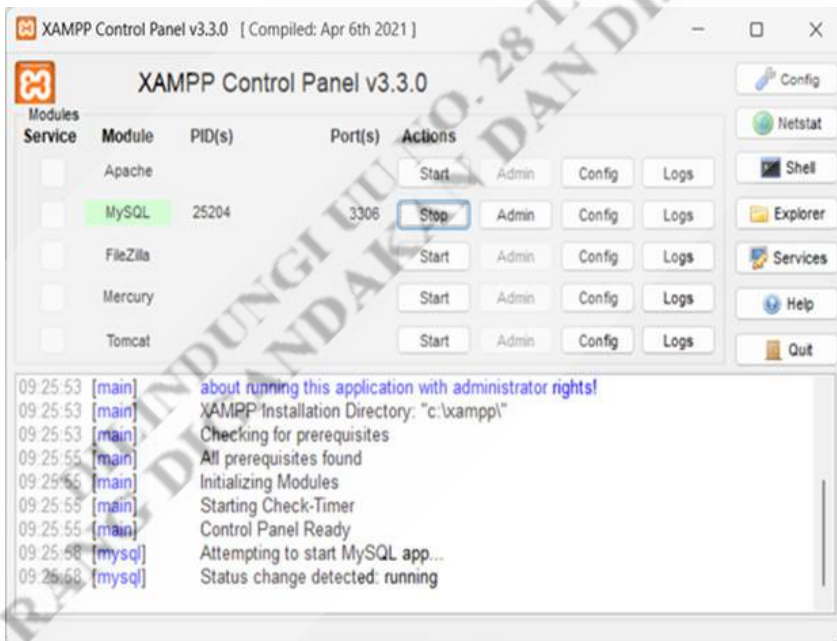
ALTER TABLE
    saledetail ADD CONSTRAINT saledetail_saleid_foreign FOREIGN KEY(SaleID) REFERENCES
sale(SaleID);
ALTER TABLE
    saledetail ADD CONSTRAINT saledetail_productid_foreign FOREIGN KEY(ProductID) REFER-
ENCES product(ProductID);
ALTER TABLE
    membership ADD CONSTRAINT membership_customerid_foreign FOREIGN KEY(CustomerID)
REFERENCES customer(CustomerID);
ALTER TABLE
    sale ADD CONSTRAINT sale_customerid_foreign FOREIGN KEY(CustomerID) REFERENCES
customer(CustomerID);
ALTER TABLE
    sale ADD CONSTRAINT sale_branchid_foreign FOREIGN KEY(BranchID) REFERENCES store-
branch(BranchID);

```

G. Implementasi

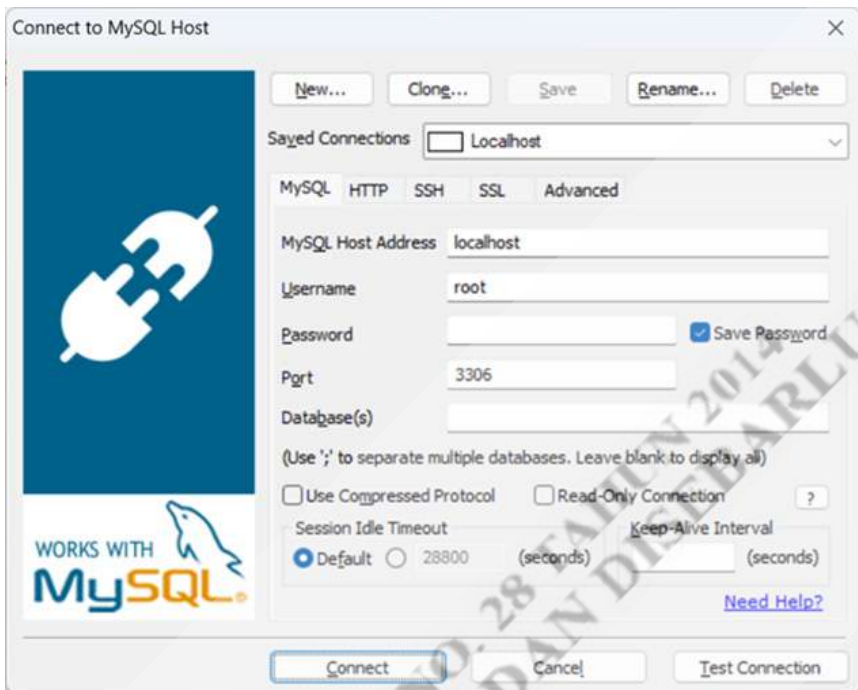
Implementasi adalah langkah di mana desain database diubah menjadi sistem database nyata dengan memanfaatkan alat dan teknologi dari Sistem Manajemen Basis Data (DBMS). Manfaatkan alat dan fitur pada DBMS dalam mendukung proses implementasi dengan menggunakan skrip SQL untuk membuat tabel, mengatur indeks, dan menambahkan batasan. Pada contoh ini MySQL (<https://www.mysql.com>) digunakan sebagai software DBMS untuk implementasi database dan aplikasi SQLyog (<https://webyog.com/product/sqlyog/>) digunakan sebagai alat untuk desain visual, pengelolaan, dan query database.

Langkah awal implementasi database yaitu menjalankan server MySQL.



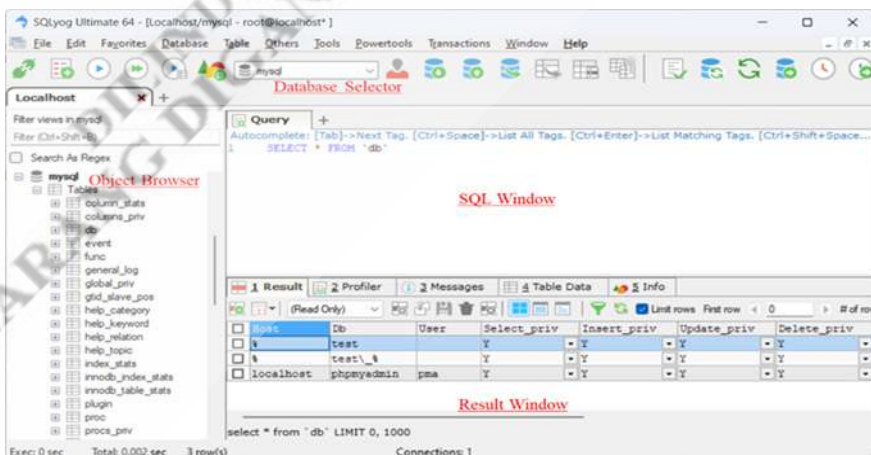
Gambar 33. Menjalankan Server MySQL Menggunakan Paket XAMPP

Selanjutnya menjalankan aplikasi SQLyog dan membuat koneksi ke server MySQL.



Gambar 34. Membuat Koneksi Server MySQL pada SQLyog

Setelah terhubung dengan server MySQL akan tampil jendela antarmuka SQLyog yang terdiri dari beberapa komponen, seperti Object Browser, SQL Window, Result Window, dan lain-lain.

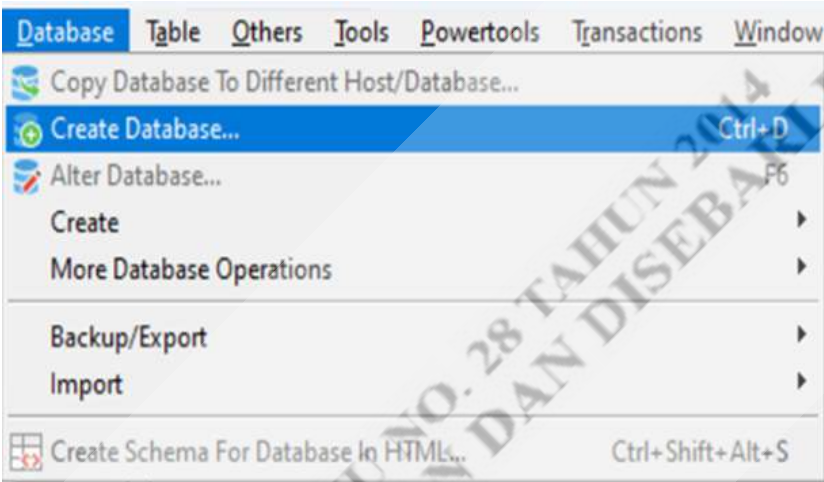


Gambar 35. Jendela Antarmuka SQLyog

Proses pembuatan database dan tabel-tabel yang telah dirancang dapat dilakukan dengan langkah-langkah berikut:

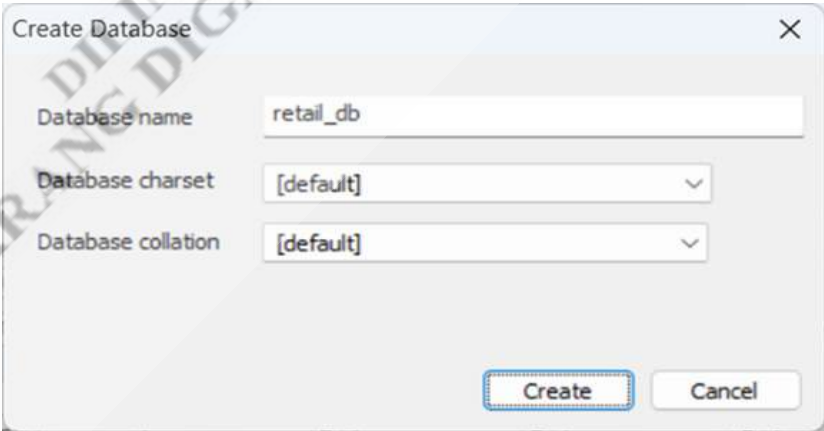
1. Membuat dan Memilih Database

Pilih menu **Database - Create Database** untuk membuat database baru di server MySQL.



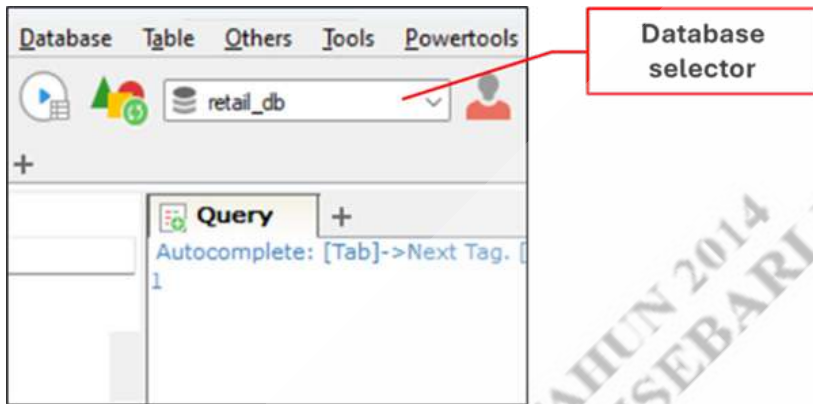
Gambar 36. Menu Create Database

Tentukan nama database dan tetapkan pengaturan lainnya seperti karakter set dan collation. Buatlah database dengan nama `retail_db`.



Gambar 37. Jendela Create Database

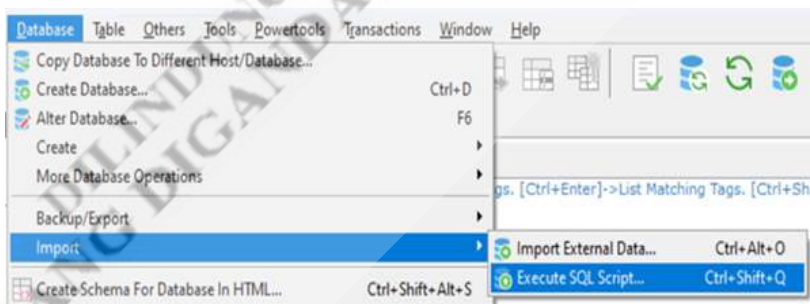
Database Selector menampilkan daftar semua database yang tersedia pada server. Klik nama database retail_db di daftar database selector untuk memilihnya.



Gambar 38. Database Selector

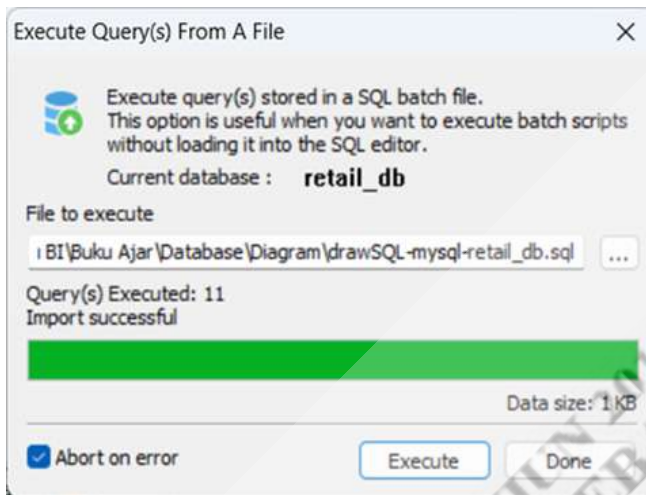
2. Membuat Tabel

Setelah database retail_db dipilih, selanjutnya membuat tabel-tabel yang telah dirancang dengan cara pilih menu **Database - Import - Execute SQL Script**.



Gambar 39. Menu Execute SQL Script

Pastikan dengan benar database dan file skrip SQL yang dipilih sebelum eksekusi skrip SQL. Pastikan juga skrip SQL sukses dieksekusi tanpa error.



Gambar 40. Jendela Eksekusi SQL Script

3. Menyisipkan Data

Langkah selanjutnya setelah membuat tabel-tabel pada database yaitu menyisipkan data ke dalam tabel. Menyisipkan data ke tabel MySQL dapat dilakukan dengan menggunakan perintah SQL insert yang ditulis di SQL Window pada aplikasi SQLyog.



Gambar 41. Menulis Perintah SQL Insert

Tombol pada SQLyog digunakan untuk mengeksekusi seluruh query yang ditulis di SQL Window. Berikut ini adalah perintah SQL insert yang digunakan untuk menyisipkan data ke dalam tabel pada database retail_db:

```

/*Data      for      the      table      'storebranch'      */
INSERT INTO 'storebranch'('BranchID','BranchName','City','Province') VALUES
('B0001','Main Branch','Jakarta','DKI Jakarta'),
('B0002','East Branch','Surabaya','Jawa Timur'),
('B0003','West Branch','Bandung','Jawa Barat'),
('B0004','North Branch','Medan','Sumatera Utara'),
('B0005','South Branch','Makassar','Sulawesi Selatan');

/*Data      for      the      table      'customer'      */
INSERT INTO 'customer'('CustomerID','NAME','Address','PhoneNumber') VALUES
('C0001','Alice Johnson','123 Elm St, Jakarta','061234567890'),
('C0002','Bob Smith','456 Oak St, Surabaya','062345678901'),
('C0003','Charlie Brown','789 Pine St, Jakarta','063456789012'),
('C0004','David Wilson','135 Maple St, Bandung','064567890123'),
('C0005','Eva Davis','246 Cedar St, Medan','065678901234');

/*Data      for      the      table      'membership'      */
INSERT INTO 'membership'('MembershipID','MembershipType','Discount','CustomerID') VALUES
('M0001','Platinum',0.30,'C0001'),
('M0002','Silver',0.10,'C0002'),
('M0003','Gold',0.20,'C0003'),
('M0004','Gold',0.20,'C0004'),
('M0005','Silver',0.10,'C0005');

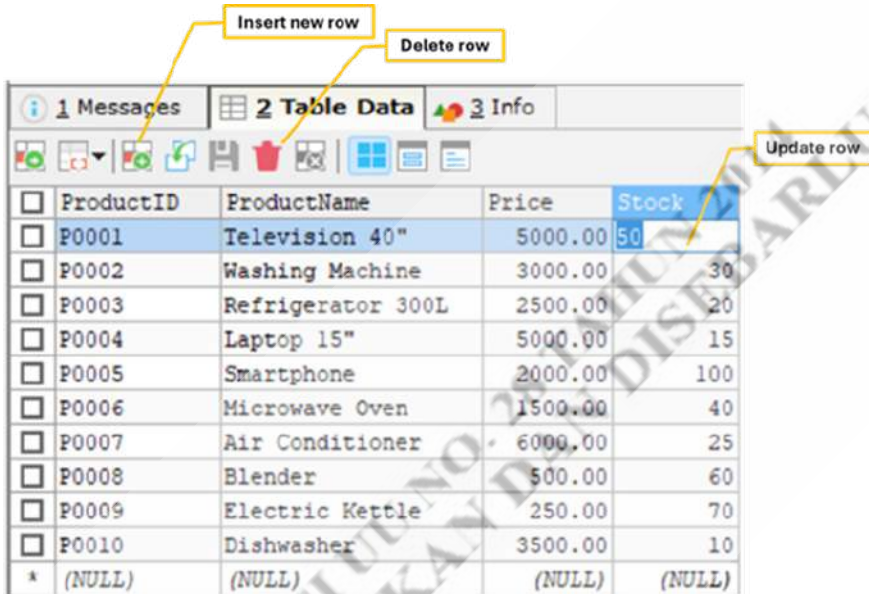
/*Data      for      the      table      'product'      */
INSERT INTO 'product'('ProductID','ProductName','Price','Stock') VALUES
('P0001','Television',4000.00,50),
('P0002','Washing Machine',3000.00,30),
('P0003','Refrigerator',2500.00,20),
('P0004','Laptop',5000.00,15),
('P0005','Smartphone',2000.00,100),
('P0006','Microwave',1500.00,40),
('P0007','Air Conditioner',6000.00,25),
('P0008','Blender',500.00,60),
('P0009','Electric Kettle',250.00,70),
('P0010','Dishwasher',3500.00,10);

/*Data      for      the      table      'sale'      */
INSERT INTO 'sale'('SaleID','SaleDate','CustomerID','BranchID') VALUES
('S0001','2023-01-05','C0001','B0001'),
('S0002','2023-01-12','C0002','B0002'),
('S0003','2023-01-15','C0003','B0003'),
('S0004','2023-01-20','C0004','B0004');

/*Data      for      the      table      'saledetail'      */
INSERT INTO 'saledetail'('SaleID','ProductID','UnitPrice','Quantity','TotalPrice') VALUES
('S0001','P0001',4000.00,1,4000.00),
('S0001','P0004',5000.00,1,5000.00),
('S0001','P0005',2000.00,2,4000.00),
('S0001','P0006',1500.00,1,1500.00),
('S0002','P0002',3000.00,1,3000.00),
('S0002','P0005',2000.00,1,2000.00),
('S0002','P0007',6000.00,1,6000.00),
('S0002','P0008',500.00,2,1000.00),
('S0003','P0003',2500.00,1,2500.00),
('S0003','P0005',2000.00,2,4000.00),
('S0003','P0006',1500.00,1,1500.00),
('S0003','P0009',250.00,3,750.00),
('S0004','P0001',4000.00,1,4000.00),
('S0004','P0004',5000.00,1,5000.00),
('S0004','P0007',6000.00,1,6000.00);

```

SQLyog menyediakan antarmuka grafis untuk menyisipkan data ke dalam tabel melalui jendela tabel data tanpa harus menggunakan perintah SQL secara langsung. Jendela tabel data dapat digunakan untuk menyisipkan, membaca, memperbarui, dan menghapus data pada tabel.



Gambar 42. Jendela Tabel Data

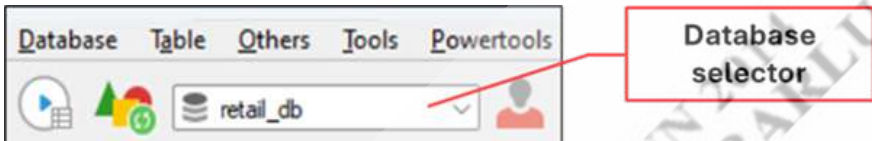
4. Ekspor dan Impor Database

Backup dan recovery adalah proses penting dalam pengelolaan database MySQL untuk melindungi data dari kehilangan atau kerusakan. Backup adalah membuat salinan data dari database untuk mencegah kehilangan data akibat kesalahan pengguna, kerusakan perangkat keras, atau serangan keamanan. Recovery adalah mengembalikan database dari backup setelah kehilangan atau kerusakan data, untuk memulihkan data ke keadaan yang konsisten dan dapat digunakan. Utilitas command-line untuk backup database MySQL menggunakan perintah `mysqldump` yang berfungsi mengekspor data dan struktur menjadi file SQL script. File backup dalam bentuk SQL script dapat diimpor kembali ke MySQL. Aplikasi SQLyog menyediakan antarmuka grafis untuk

backup melalui fitur expor dan antarmuka grafis untuk recovery melalui fitur impor.

Ekspor database menggunakan SQLyog dilakukan dengan mengikuti langkah-langkah berikut:

- a. Buka aplikasi SQLyog dan hubungkan ke server MySQL.
- b. Pilih database yang akan diekspor pada Database Selector.



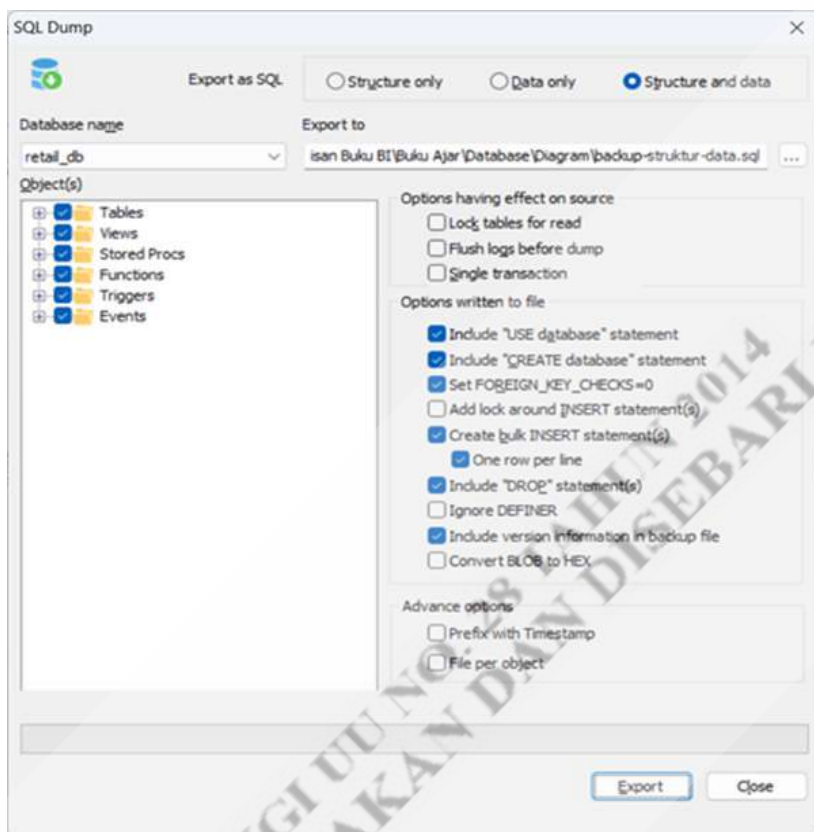
Gambar 43. Pilih Database yang Diekspor

Buka Fitur Ekspor melalui menu **Database - Backup/Export - Backup Database As SQL Dump**.



Gambar 44. Menu Backup Database

Lakukan pengaturan ekspor pada jendela pengaturan sesuai kebutuhan seperti tipe backup (structure only, data only, structure and data), lokasi dan nama file SQL dump, serta pengaturan lainnya.

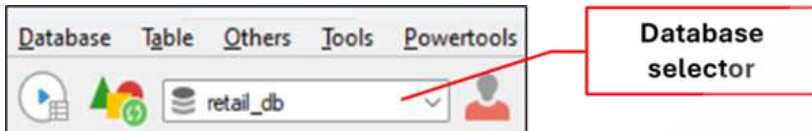


Gambar 45. Jendela Pengaturan Ekspor

- a. SQLyog akan membuat file dump SQL di lokasi yang telah ditentukan saat memulai proses ekspor.

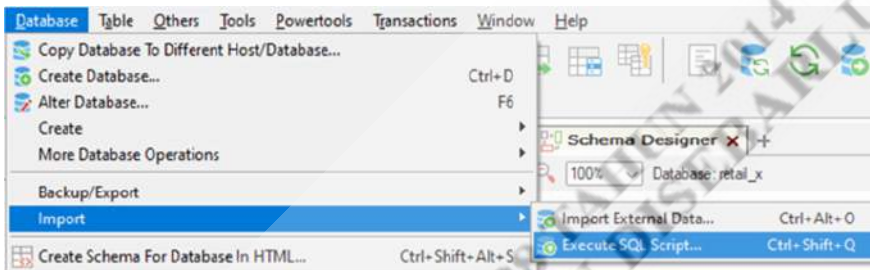
Impor database menggunakan SQLyog dilakukan dengan mengikuti langkah-langkah berikut:

- b. Buka aplikasi SQLyog dan hubungkan ke server MySQL
- c. Pilih database tujuan pada Database Selector (jika diperlukan).



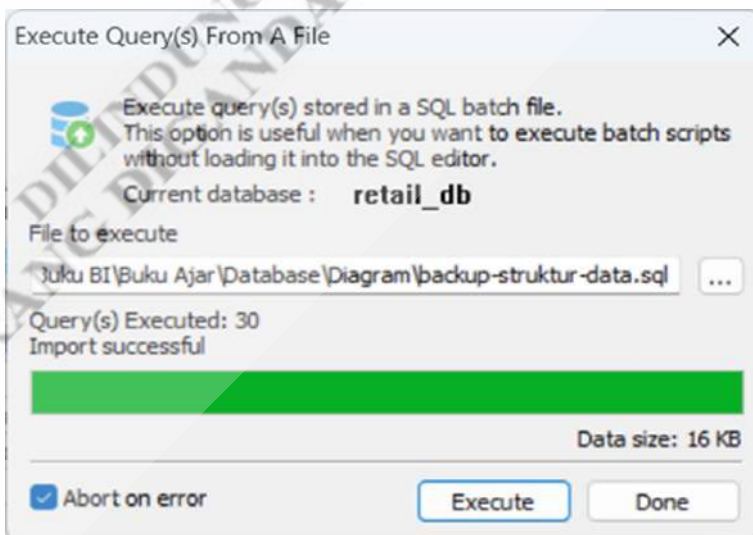
Gambar 46. Pilih Database Tujuan

- d. Buka Fitur Ekspor melalui menu **Database - Import - Execute SQL Script**.



Gambar 47. Menu Impor Database

- e. Pilih file SQL dump yang akan diimpor dan lakukan eksekusi skrip SQL. Pastikan skrip SQL sukses dieksekusi tanpa error.



Gambar 48. Jendela Impor Database

H. Rangkuman

1. Sistem informasi penting bagi operasional perusahaan yang mengandalkan data. Ketidaktersediaan sistem dapat menghentikan operasi perusahaan.
2. Database adalah kumpulan data yang diatur secara terstruktur untuk memudahkan akses dan pengelolaan, membantu dalam penyimpanan, pengambilan, dan pengaturan data dalam jumlah besar.
3. Manfaat perancangan database yang baik adalah efisiensi pengelolaan, konsistensi data, kemudahan perawatan, kemampuan skalabilitas, keamanan data, pengurangan biaya, dukungan pengambilan keputusan, dan integrasi yang lebih mudah.
4. Database Management System (DBMS) adalah perangkat lunak untuk mengelola database, seperti MySQL, Oracle, SQL Server, dan MongoDB. DBMS menyediakan antarmuka untuk operasi data dan menawarkan berbagai keuntungan seperti mengurangi redudansi dan menyediakan backup.
5. Relational Database Management System (RDBMS) menggunakan tabel untuk menyimpan data dengan struktur baris dan kolom. Fitur utama termasuk integritas, akurasi, dan konsistensi data, serta karakteristik ACID (Atomicity, Consistency, Isolation, Durability).
6. ER model adalah model data tingkat tinggi yang membantu dalam desain konseptual database dengan entitas, atribut, relasi, dan domain.
7. Proses pembuatan database dengan MySQL Meliputi instalasi dan konfigurasi, pembuatan database dan tabel, desain skema, manipulasi data, query dan pengambilan data, keamanan dan pengelolaan pengguna, serta optimasi dan pemeliharaan.
8. Langkah-langkah perancangan database relasional yaitu identifikasi kebutuhan, analisis konseptual, desain logis, desain fisik, dan implementasi database.

9. Fungsi dan keuntungan DBMS adalah mengendalikan redundansi data, memungkinkan berbagi data, menyediakan backup dan pemulihan, serta menawarkan berbagai antarmuka pengguna.
10. Karakteristik dan kategori integritas data dalam RDBMS yaitu tabel dalam RDBMS memiliki nama unik dan tidak boleh memiliki tuple duplikat. Kategori integritas data termasuk integritas entitas, domain, referensial, dan yang didefinisikan oleh pengguna untuk menjaga konsistensi dan akurasi data.

I. Soal Latihan

1. Mengapa sistem informasi menjadi sangat penting bagi operasional perusahaan yang bergantung pada data, dan apa yang bisa terjadi jika sistem tersebut tidak tersedia?
2. Bagaimana database membantu dalam pengelolaan data yang besar, dan apa saja manfaat dari perancangan database yang baik?
3. Sebutkan tiga contoh perangkat lunak Database Management System (DBMS) untuk mengelola database,
4. Apa yang dimaksud dengan Relational Database Management System (RDBMS) dan bagaimana karakteristik ACID (Atomicity, Consistency, Isolation, Durability) menjaga integritas data?
5. Langkah-langkah apa saja yang terlibat dalam perancangan database relasional?
6. Jelaskan cara membuat database dan tabel menggunakan MySQL untuk kasus penyewaan kendaraan sebagai berikut:

Sebuah perusahaan penyewaan kendaraan ingin membangun sistem basis data untuk mengelola operasional mereka. Sistem ini harus mampu menangani data tentang kendaraan, pelanggan, transaksi penyewaan, dan detail pembayaran. Tujuan dari sistem ini adalah untuk meningkatkan efisiensi pengelolaan data, memudahkan pelaporan, dan meningkatkan pengalaman pelanggan.

Tabel yang perlu dikelola:

Kendaraan

ID_Kendaraan	int
Nomor_Registrasi	varchar(15)
Merek	varchar(50)
Model	varchar(50)
Tahun	int
Status	enum('Tersedia','Disewa')

Pelanggan

ID_Pelanggan	int
Nama	varchar(100)
Alamat	varchar(255)
Nomor_Telepon	varchar(15)
Email	varchar(100)

Penyewaan

ID_Penyewaan	int
ID_Kendaraan	int
ID_Pelanggan	int
Tanggal_Mulai	date
Tanggal_Selesai	date
Status	varchar(15)

Pembayaran

ID_Pembayaran	int
ID_Penyewaan	int
Tanggal_Pembayaran	date
Jumlah	decimal(10
Metode_Pembayaran	varchar(50)

Diagram fisik:



Gambar 49. Diagram Fisik Penyewaan Kendaraan

BAB V

Data Warehouse

Menggunakan Pentaho

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu menguasai prinsip dan teknik perancangan data warehouse dengan menggunakan Pentaho Data Integration.

Setelah membaca buku ini diharapkan mahasiswa mengetahui data warehouse menggunakan pentaho, komponen aplikasi kete, transformasi, cara kerja spoon.

Materi yang diberikan :

1. Datawarehouse menggunakan Pentaho.
2. Komponen Aplikasi Kete.
3. Transformasi.
4. Cara kerja Spoon.

Capaian Pembelajaran:

Capaian pembelajaran mata kuliah pada bab ini adalah mahasiswa mampu menguasai prinsip dan teknik perancangan datawarehouse dengan menggunakan *Pentaho Data Integration*. Capaian pembelajaran pada bab ini adalah mampu menguasai prinsip dan teknik perancangan sistem terintegrasi dengan pendekatan sistem.

B. Data Warehouse Menggunakan Pentaho

Pentaho Data Integration (PDI) menawarkan fungsi Extract, Transform, and Load (ETL) yang mendukung proses pengambilan, pembersihan, dan penyimpanan data dalam format yang konsisten dan seragam, sehingga dapat diakses dan relevan bagi pengguna akhir dan teknologi Internet of Things (IoT). Pentaho Data Integration sering disebut sebagai "Kettle." Proyek ini awalnya merupakan proyek open source bernama "Kettle." Istilah K.E.T.T.L.E adalah istilah rekursif yang berarti Kettle Extraction Transformation Transport Load Environment. Setelah Pentaho mengakuisisi Kettle, namanya diubah menjadi Pentaho Data Integration. Nama komponen PDI lainnya seperti Spoon, Pan, dan Kitchen awalnya terinspirasi dari metafora "kuliner" dalam penawaran ETL (Hitachi Vantara, 2024).

Klien PDI (yang juga dikenal sebagai Spoon) adalah aplikasi desktop untuk membuat transformasi serta menjadwalkan dan menjalankan tugas. Penggunaan umum klien PDI mencakup:

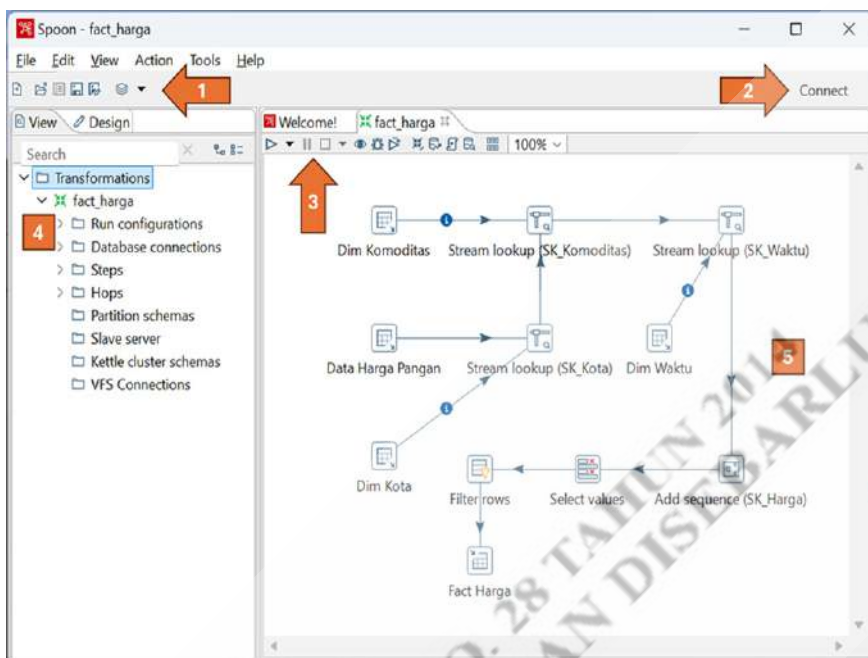
1. Migrasi data antar basis data dan aplikasi yang berbeda.
2. Memasukkan set data besar ke dalam basis data.
3. Pembersihan data dengan berbagai langkah.
4. Integrasi data, termasuk kemampuan untuk menggunakan real-time ETL sebagai sumber data untuk Pentaho Reporting.
5. Pengisian data warehouse.

C. Komponen Aplikasi Kettle

Kettle adalah alat yang digunakan untuk merancang dan menjalankan tugas ETL. Juga dikenal sebagai Pentaho Data Integration, Kettle adalah bagian dari suite Pentaho Business Analytics. Kettle terdiri dari komponen-komponen berikut (Vaisman & Zimányi, 2014):

1. Data Integration Server, yang menjalankan tugas-tugas integrasi data. Fungsi utamanya termasuk mengeksekusi job dan transformation, mengatur dan mengelola keamanan, menyediakan fasilitas manajemen konten untuk mengelola job dan transformation dalam lingkungan pengembangan kolaboratif, serta menawarkan layanan untuk penjadwalan dan pemantauan aktivitas.
2. Spoon, antarmuka pengguna grafis untuk merancang job dan transformation. Transformation dapat dijalankan baik secara lokal dalam Spoon maupun di Data Integration Server. Spoon memungkinkan pembuatan job ETL yang kompleks tanpa perlu menulis atau membaca kode.
3. Pan, alat baris perintah mandiri untuk mengeksekusi transformation. Pan membaca dan menulis data ke berbagai sumber data serta digunakan untuk memanipulasi data.
4. Kitchen, alat baris perintah mandiri untuk mengeksekusi job. Job umumnya dijadwalkan untuk berjalan dalam mode batch pada interval tertentu.
5. Carte, server ringan untuk menjalankan job dan transformation di host remote. Carte memiliki kemampuan eksekusi serupa dengan Data Integration Server tetapi tidak menawarkan fasilitas penjadwalan, keamanan, dan manajemen konten.

Kettle memungkinkan untuk membuat transformation dan job.



Gambar 50. Jendela Antarmuka Kettle

Gunakan item bernomor pada Gambar 50 untuk merujuk ke toolbar, tombol, dan area di Kettle.

Tabel 4. Fitur Antarmuka Kettle

Item	Feature	Description
1	Toolbar	Gunakan toolbar ini untuk mengakses tindakan yang sering dilakukan: <ul style="list-style-type: none"> - Tombol New file untuk membuat job, transformation, database connection, atau slave server baru. - Tombol Open file untuk membuka transformation atau job dari file. - Tombol Explore Repository untuk menjelajahi repositories.

		<ul style="list-style-type: none"> - Tombol Save untuk menyimpan transformation atau job saat ini ke file atau repository. - Tombol Save As untuk menyimpan transformation atau job dengan nama atau tipe file yang berbeda. - Tombol Perspectives untuk beralih antara berbagai perspektif: <ul style="list-style-type: none"> - Data Integration Perspective: Membuat ETL transformation dan job. - Schedule Perspective: Mengelola kegiatan ETL yang dijadwalkan di Pentaho Server.
2	Connect button	Gunakan tombol ini untuk mengakses menu untuk membuat dan menghubungkan ke repositories guna menyimpan ETL job dan transformation secara terpusat.
3	Sub-toolbar	<p>Gunakan toolbar ini untuk melakukan tindakan pada transformation atau job:</p> <ul style="list-style-type: none"> - Tombol Run untuk menjalankan transformation atau job: <ul style="list-style-type: none"> - Run: Menjalankan transformation atau job saat ini dari file XML atau repository. - Run Options: Mengatur Run Options kemudian menjalankan transformation atau job saat ini dari file XML atau repository. - Tombol Pause untuk menjeda transformation atau job yang sedang berjalan. - Tombol Stop untuk menghentikan transformation atau job yang sedang berjalan: <ul style="list-style-type: none"> - Stop: Menghentikan transformation atau job secara langsung.

		<ul style="list-style-type: none"> - Stop input processing: Menghentikan langkah input ke transformation atau job, namun membiarkan rekaman yang sudah diambil atau dimulai untuk diproses sebelum dihentikan. - Tombol Preview untuk menjalankan transformation dalam mode preview untuk memeriksa baris yang dihasilkan oleh langkah-langkah tertentu. - Tombol Debug untuk menjalankan transformation dalam mode debug guna memecahkan masalah eksekusi. - Tombol Replay untuk memutar ulang pemrosesan transformation. - Tombol Verify untuk memverifikasi transformation. - Tombol Analyze untuk menjalankan analisis dampak pada database. - Tombol SQL untuk menghasilkan SQL yang diperlukan untuk menjalankan transformation yang dimuat. - Tombol Explore DB untuk membuka Database Explorer guna melakukan tindakan seperti melihat pratinjau data, menjalankan query SQL, dan menghasilkan DDL. - Tombol Results untuk menampilkan panel Execution Results. - Tombol Lock untuk mengunci transformation.
4	Explore pane	<p>Gunakan panel ini untuk mengakses tab Design dan View:</p> <ul style="list-style-type: none"> - Tab Design menyediakan daftar step atau entry yang digunakan untuk membangun transformation atau job. - Tab View menyediakan informasi tentang koneksi database yang tersedia serta step dan hop yang digunakan untuk transformation atau job.

5	Canvas	Gunakan kanvas ini untuk merancang dan membangun transformation serta job untuk aktivitas ETL yang ingin dilakukan.
---	--------	---

D. Transformasi

Kettle menerapkan istilah workflow sebagai elemen dasar dalam mentransformasi data dan melakukan tugas-tugas lainnya. Workflows dibangun dengan menggunakan step atau entry saat merancang transformation dan job. Aliran data antar step atau entry dihubungkan oleh hop yang memindahkan data dari satu elemen ke elemen berikutnya. Sebuah transformation adalah jaringan tugas logis yang dikenal sebagai step. Dua komponen utama yang terkait dengan transformasi adalah step dan hop. Transformation pada intinya merupakan aliran data. Sebagai contoh, pengembang database membuat transformation yang membaca flat file, memfilternya, mengurutkannya, dan memuatnya ke dalam tabel database relasional. Pada dasarnya, transformation adalah grafik berarah dari serangkaian konfigurasi transformasi data logis. File transformation memiliki ekstensi .ktr.

Kettle memiliki beberapa elemen yang berfungsi sebagai berikut:

1. Transformation: Tugas logis yang terdiri dari step yang dihubungkan oleh hop. Transformation berfungsi sebagai aliran data untuk mengekstrak, mentransformasi, dan memuat data.
2. Step: Elemen dasar dalam sebuah transformation. Setiap step menjalankan tugas tertentu, seperti membaca data dari flat file, memfilter baris, atau menulis ke database. Step di Kettle dikelompokkan berdasarkan fungsinya, seperti input, output, dan scripting. Step dalam transformation dijalankan secara paralel dalam thread masing-masing.
3. Hop: Jalur data yang menghubungkan step, memungkinkan data berpindah dari satu step ke step berikutnya. Hop menentukan aliran data antara step, meskipun tidak selalu

menentukan urutan pelaksanaannya.

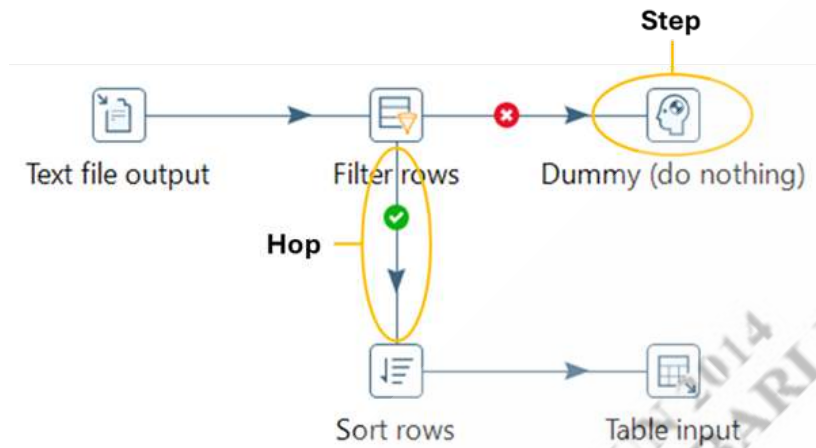
4. Job: Workflows yang mengoordinasikan berbagai fungsi untuk melaksanakan seluruh proses ETL. Job terdiri dari job entries, job hops, dan job settings.
5. Job entries: Blok dasar dalam sebuah job yang setara dengan step dalam transformation.
6. Job hops: Menentukan urutan pelaksanaan job entries dan kondisi di mana mereka dijalankan berdasarkan hasil dari entry sebelumnya. Job hops berbeda dari hop dalam transformation.
7. Job settings: Opsi yang mengontrol perilaku job dan cara pencatatan tindakan job tersebut.

Step memperluas dan mengembangkan fungsionalitas transformation di Pentaho Data Integration (PDI). Berikut beberapa step yang umum digunakan dalam proses ETL di PDI:

1. CSV file input: Step ini digunakan untuk mengambil data dari file CSV. Dengan cara menentukan file CSV yang akan dibaca, mengatur delimiter, encoding, serta format data yang akan dimasukkan ke dalam transformasi.
2. Microsoft Excel input: Step ini untuk mengambil data dari file Microsoft Excel (.xls atau .xlsx). Dengan cara memilih sheet, rentang sel, dan mengatur cara pembacaan data, termasuk format kolom dan headernya.
3. Table input: Step ini digunakan untuk menjalankan query SQL pada database dan mengambil hasilnya sebagai input untuk transformasi. Ini berguna untuk mengambil data langsung dari tabel dalam database.
4. Text file output: Step ini digunakan untuk menulis data ke dalam file teks. Dengan cara menentukan format file, delimiter, encoding, serta cara data ditulis, misalnya apakah akan menambahkan header atau tidak.
5. Microsoft Excel output: Step ini untuk menulis data ke dalam file Microsoft Excel. Dengan cara menentukan file Excel tujuan,

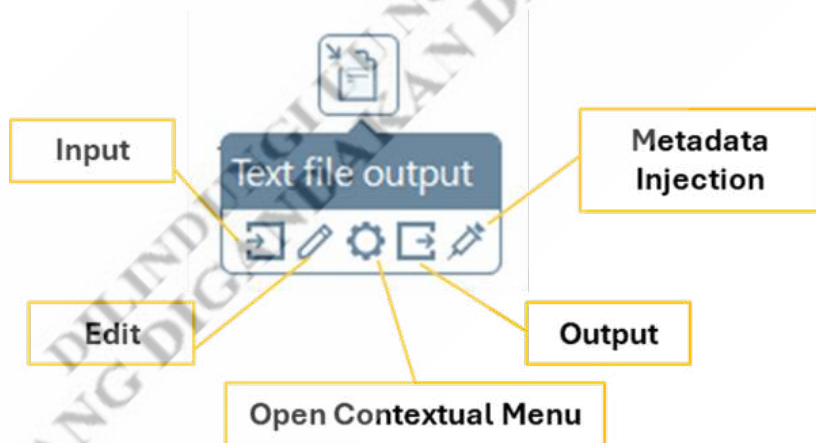
sheet yang akan digunakan, dan format penulisan data.

6. Table output: Step ini digunakan untuk memasukkan data ke dalam tabel di database. Dengan cara mengatur tabel tujuan, cara menangani data yang sudah ada (misalnya update atau insert), serta metode koneksi ke database.
7. Add sequence: Step ini digunakan untuk menambahkan nomor urut (sequence) pada data yang sedang diproses. Sequence ini dapat digunakan sebagai kunci utama atau untuk keperluan lainnya yang memerlukan penomoran data secara berurutan.
8. Calculator: Step ini untuk melakukan operasi perhitungan matematis pada data, seperti penjumlahan, pengurangan, perkalian, atau fungsi matematis lainnya. Hasil perhitungan ini dapat disimpan dalam kolom baru.
9. Select values: Step ini digunakan untuk memilih, mengganti nama, mengubah tipe data, atau menghapus kolom tertentu dari data yang sedang diproses. Ini berguna untuk menyiapkan data sebelum diproses lebih lanjut.
10. Stream lookup: Step ini digunakan untuk mencari nilai dalam aliran data lain yang sedang diproses dalam transformasi. Ini mirip dengan melakukan join antar tabel, tetapi dilakukan dalam aliran data yang berbeda dalam transformasi yang sama.
11. Setiap step ini memainkan peran penting dalam proses ETL, memungkinkan untuk mengelola dan mentransformasi data sesuai kebutuhan bisnis. Sebuah step dapat memiliki beberapa koneksi. Beberapa step terhubung ke step lainnya, sementara yang lain berfungsi sebagai input atau output untuk step lain. Aliran data melewati step menuju step lainnya dalam sebuah transformation. Hop ditampilkan sebagai panah di Spoon. Hop memungkinkan data berpindah dari satu step ke step berikutnya dan menentukan arah serta aliran data melalui step tersebut.



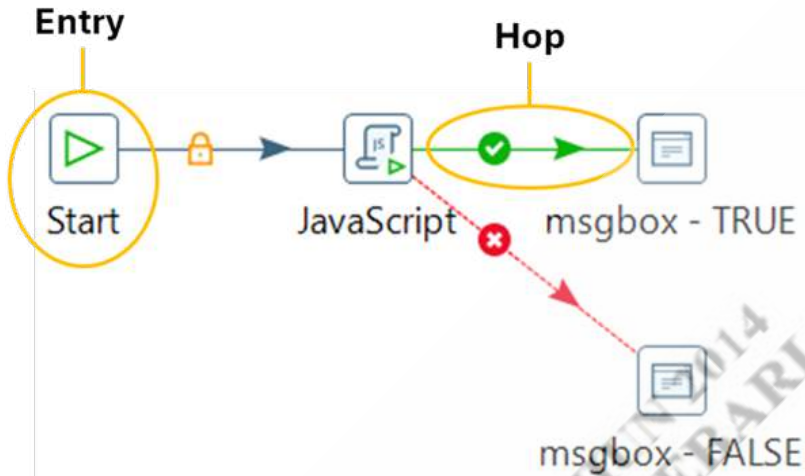
Gambar 51. Ilustrasi Transformation

Dengan menggunakan menu hover dapat menghubungkan step, mengedit step, dan membuka menu kontekstual step dengan mengklik untuk mengedit step.



Gambar 52. Menu Hover Step

Job merupakan model yang mirip dengan workflow untuk mengatur sumber daya, eksekusi, dan ketergantungan dalam aktivitas ETL. Job mengumpulkan berbagai fungsi individu untuk melaksanakan proses secara keseluruhan.



Gambar 53. Ilustrasi Job

E. Cara Kerja Spoon

Spoon adalah antarmuka pengguna grafis untuk Pentaho Data Integration (PDI) yang digunakan untuk merancang dan mengelola transformation serta job. Berikut adalah cara kerja Spoon secara umum:

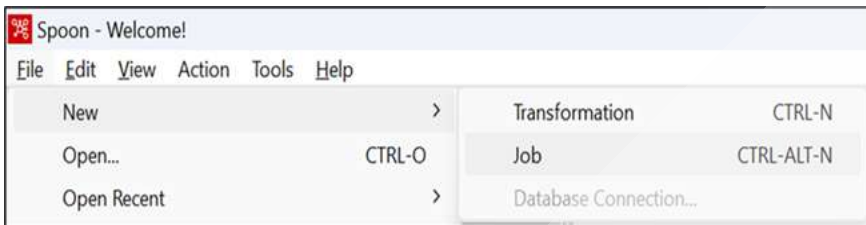
1. **Antarmuka Utama:** Antarmuka utama terdiri dari beberapa bagian, termasuk toolbar, area desain, panel navigasi, dan panel hasil. Melalui antarmuka ini memungkinkan untuk mengakses berbagai fungsi dan fitur yang diperlukan untuk merancang dan mengelola transformation serta job.
2. **Desain Transformation:** Canvas digunakan untuk mendesain transformation dengan menambahkan dan menghubungkan step. Step dapat ditambahkan ke kanvas dari panel step dan mengonfigurasi setiap step sesuai kebutuhan. Step ini dihubungkan dengan hop, yang menunjukkan aliran data antar step. Hop digunakan untuk menghubungkan step dan menentukan arah aliran data.
3. **Desain Job:** Untuk mendesain job dapat menggunakan canvas di mana menambahkan job entries dan menghubungkannya

dengan hop. Job entries adalah unit fungsional dalam job yang dapat mencakup langkah-langkah seperti menjalankan transformation, mengirim email, atau memanggil skrip. Seperti pada transformation, hop dalam job menunjukkan aliran eksekusi antar job entries.

4. **Pengaturan dan Konfigurasi:** Setiap step atau entry dalam transformation atau job dapat dikonfigurasi dengan mengklik ganda pada elemen tersebut. Cara ini dapat mengatur parameter, opsi, dan properti lain yang relevan. Untuk melihat pratinjau data yang dihasilkan oleh transformation dengan menggunakan tombol Preview yang dapat membantu memeriksa hasil dan memastikan bahwa data diproses dengan benar.
5. **Eksekusi dan Debugging:** Menjalankan transformation atau job langsung dari Spoon dengan menggunakan tombol Run untuk menguji dan menerapkan desain. Spoon menyediakan fitur debugging untuk membantu memecahkan masalah dengan menjalankan transformation dalam mode debug dan melihat log serta hasil eksekusi secara rinci.
6. **Simpan dan Kelola:** Menyimpan transformation dan job dalam format file .ktr (untuk transformation) atau .kjb (untuk job), serta mengelola file-file ini di repository yang telah dikonfigurasi. Spoon juga memungkinkan untuk mengelola versi transformation dan job, serta berkolaborasi dengan tim dalam pengembangan dan pemeliharaan proyek ETL.

Secara keseluruhan, Spoon berfungsi sebagai alat grafis yang memudahkan perancangan, konfigurasi, dan eksekusi transformation dan job dalam Pentaho Data Integration. Sebagai contoh penggunaan Spoon, berikut ini cara membuat suatu job sederhana yang akan menampilkan satu kotak pesan ketika dijalankan.

Buat file job baru dengan cara pilih menu File – New – Job.



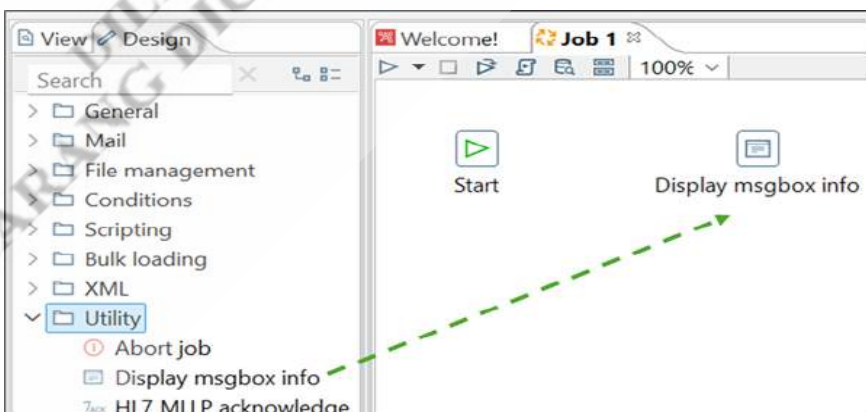
Gambar 54. Menu Membuat Job

Di panel kiri pada opsi Design, buka folder General, lalu pilih step Start dan seret (drag) ke canvas.



Gambar 55. Menambahkan Step Start

Tambahkan step Display Msgbox Info dari folder Utility dengan cara yang sama.



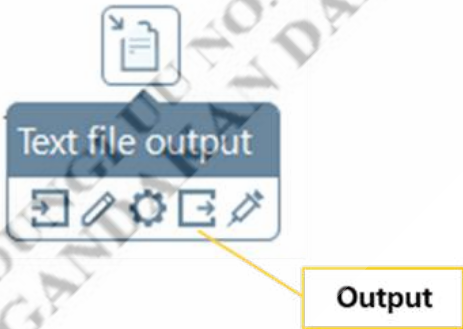
Gambar 56. Menambahkan Step Display Msgbox Info

Hubungkan kedua step pada canvas dengan cara berikut tekan CTRL+klik pada step sumber (Start), lalu tekan CTRL+klik pada step tujuan (Display Msgbox Info). Setelah itu, klik kanan dan pilih New Hop.



Gambar 57. Menu New Hop

Sebagai alternatif menggambar hop yaitu dengan mengarahkan kursor ke sebuah step hingga menu hover muncul. Seret ikon hop painter dari step sumber ke step tujuan.



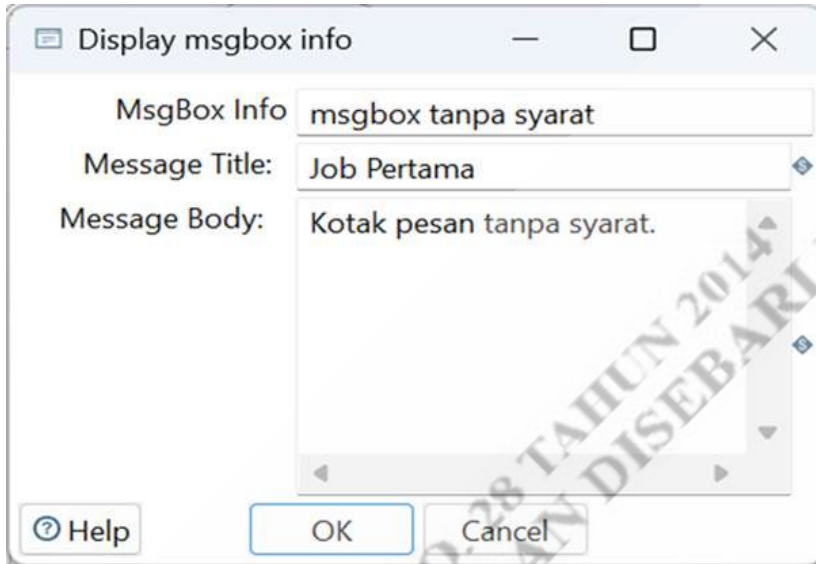
Gambar 58. Menu Hover New Hop

Kedua step akan terhubung oleh hop, yang menandakan kondisi unconditional (tanpa syarat).



Gambar 59. Hop Antara Step Start dan Step Display Msgbox Info

Klik dua kali pada step Display MsgBox Info, lalu masukkan konfigurasi sesuai dengan gambar di bawah ini.



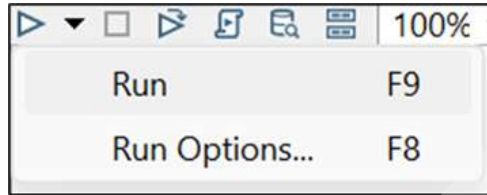
Gambar 60. Konfigurasi Step Display MsgBox Info

Klik tombol OK, dan nama step telah berubah menjadi msgbox tanpa syarat. Kemudian simpan file job.



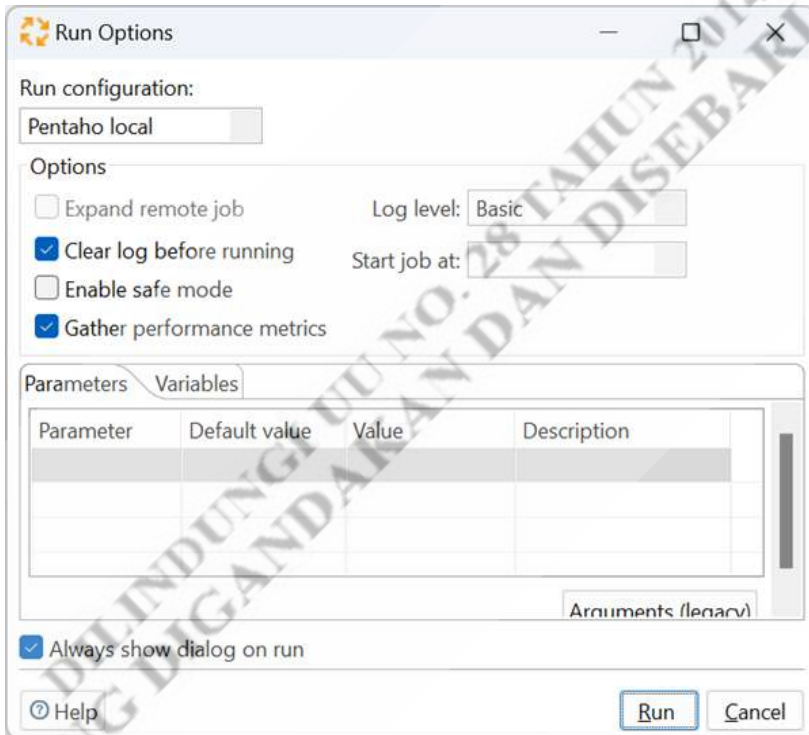
Gambar 61. Simpan File Job

Jalankan job dengan cara klik tombol Run pada sub-toolbar.



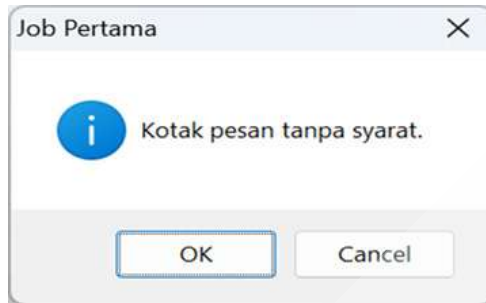
Gambar 62. Tombol Run

Klik tombol Run pada dialog Run Option.



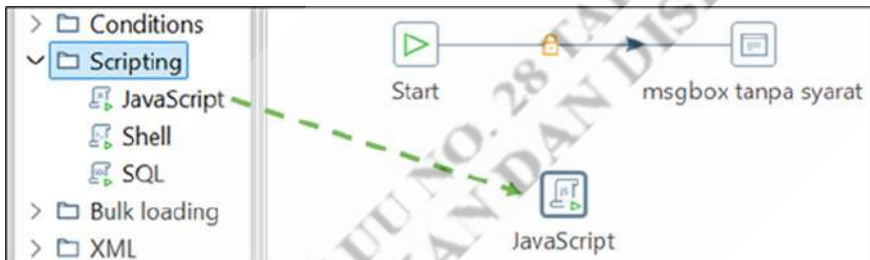
Gambar 63. Jendela Run Option

Hasil dari eksekusi job ini adalah tampilnya sebuah kotak pesan seperti yang terlihat pada Gambar 64.



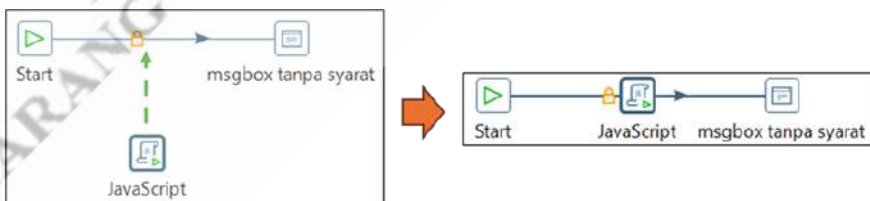
Gambar 64. Menampilkan Kotak Pesan Unconditional

Lanjutkan pekerjaan sebelumnya dengan menambahkan step JavaScript ke dalam canvas.



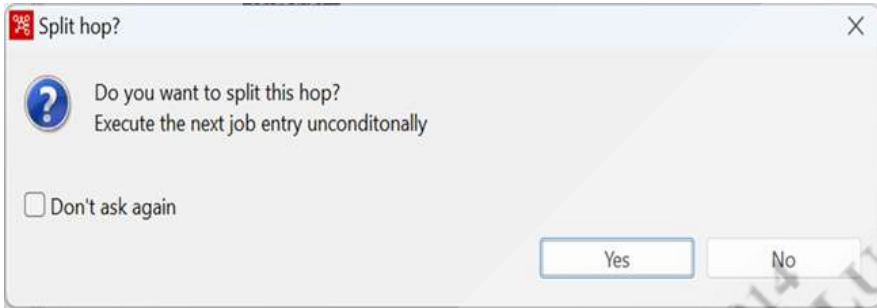
Gambar 65. Menambahkan Step JavaScript

Klik dan tahan step JavaScript, lalu seret step tersebut ke hop yang menghubungkan step Start dan msgbox tanpa syarat hingga hop tersebut tampak lebih tebal. Setelah itu, lepaskan klik dari step tersebut.



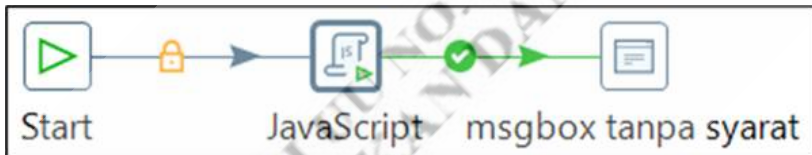
Gambar 66. Menggabungkan Step JavaScript

Klik Yes pada dialog Split hop.



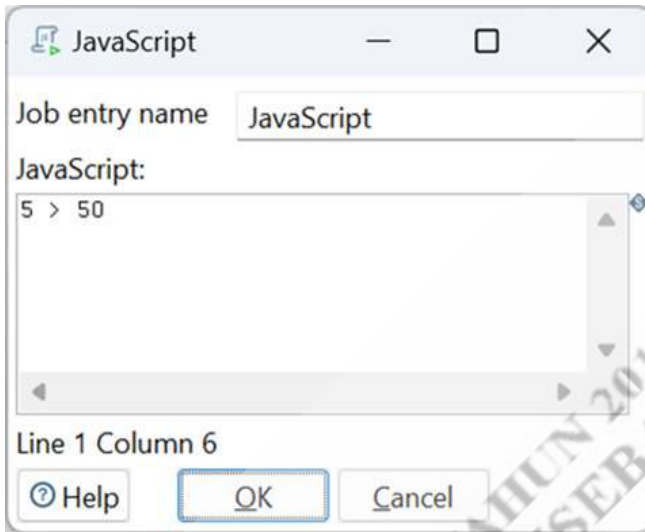
Gambar 67. Jendela Dialog Split Hop

Step JavaScript akan menjadi step intermediate yang baru. Hop yang menuju step msgbox tanpa syarat akan berwarna hijau, menandakan bahwa step tersebut akan dieksekusi jika step JavaScript berhasil dijalankan.



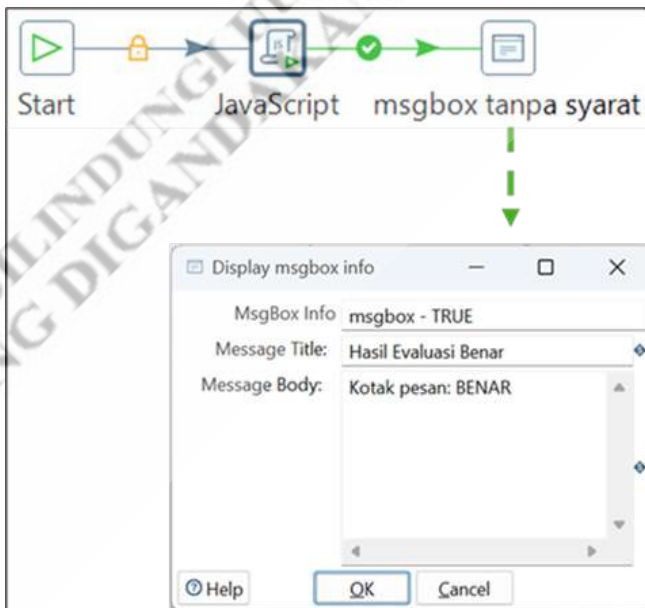
Gambar 68. Step Intermediate JavaScript

Klik dua kali pada step JavaScript untuk membuka editor JavaScript. Masukkan ekspresi perbandingan nilai sesuai dengan Gambar 69, lalu klik tombol OK setelah selesai.



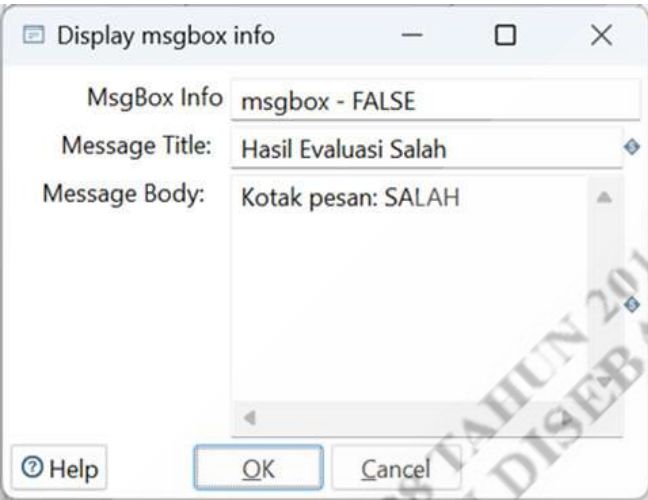
Gambar 69. Editor JavaScript

Sesuaikan properti step msgbox tanpa syarat sesuai dengan Gambar 70.



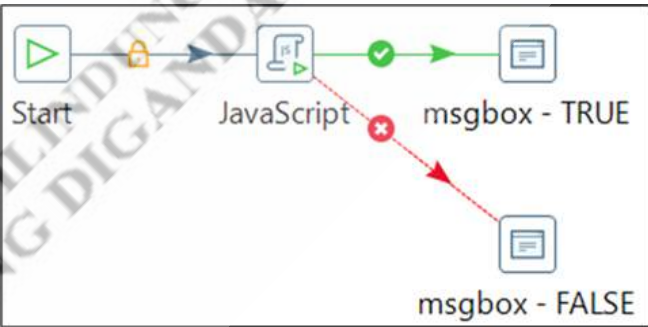
Gambar 70. Mengubah Properti Msgbox

Tambahkan step Display MsgBox Info baru ke dalam kanvas dengan konfigurasi sesuai Gambar 71.



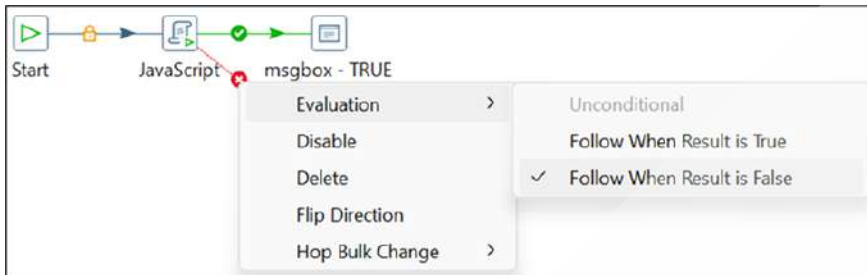
Gambar 71. Konfigurasi Display MsgBox Info Baru

Buat hop baru menggunakan menu hover dari step JavaScript ke msgbox – False. Hop yang baru terbentuk akan berwarna merah secara otomatis.



Gambar 72. Job Kotak Pesan Conditional

Cara untuk mengubah konfigurasi evaluation true atau false dapat dilakukan dengan klik kanan pada hop, lalu pilih Evaluation. Pilih Follow when result is true untuk melanjutkan jika nilai kondisi dari step sebelumnya benar atau pilih Follow when result is false untuk melanjutkan jika nilai kondisi dari step sebelumnya salah.



Gambar 73. Menu Evaluation

Jalankan job dan hasil yang diharapkan yaitu tampilnya dialog seperti berikut.



Gambar 74. Menampilkan Kotak Pesan Conditional

Hasil ini menunjukkan bahwa evaluasi pada step JavaScript adalah salah. Klik tombol OK untuk mengakhiri eksekusi job. Job dapat dianalisis dengan memeriksa rincian eksekusi job pada panel Execution Results.

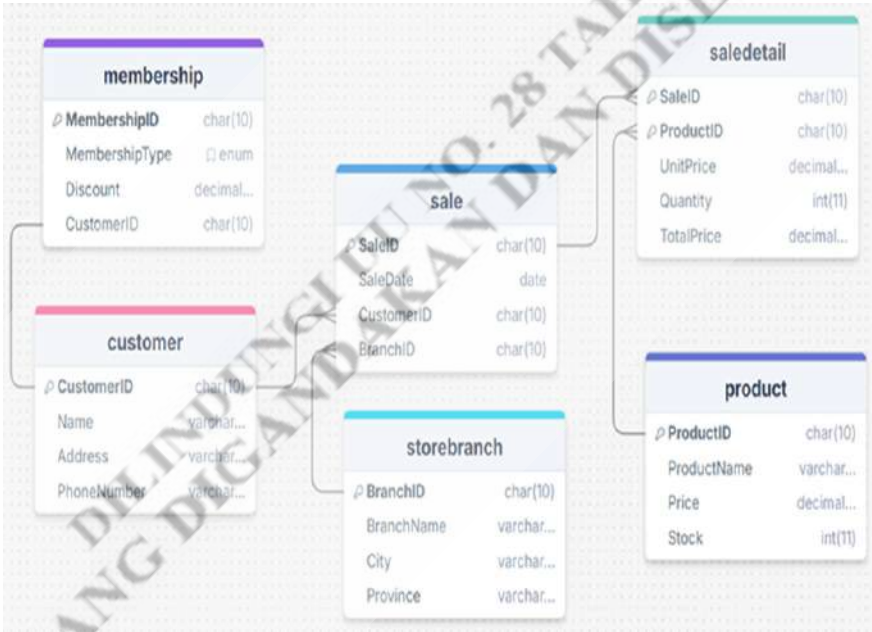
Execution Results			
<div> <div>Logging</div> <div>History</div> <div>Job metrics</div> <div>Metrics</div> </div>			
Job / Job Entry	Comment	Result	Reason
▼ msgbox-tanpa_syarat			
Job: msgbox-tanpa_syarat	Start of job execution		start
Start	Start of job execution		start
Start	Job execution finished	Success	
JavaScript	Start of job execution		Followed unconditional link
JavaScript	Job execution finished	Failure	
msgbox - FALSE	Start of job execution		Followed link after failure
msgbox - FALSE	Job execution finished	Success	
Job: msgbox-tanpa_syarat	Job execution finished	Success	finished

Gambar 75. Panel Execution Results

Berikut ini akan ditunjukkan cara membuat ETL studi kasus sistem penjualan pada jaringan toko elektronik dengan sumber data berupa database transaksi penjualan yang sudah dibuat pada bab sebelumnya bernama `retail_db`.

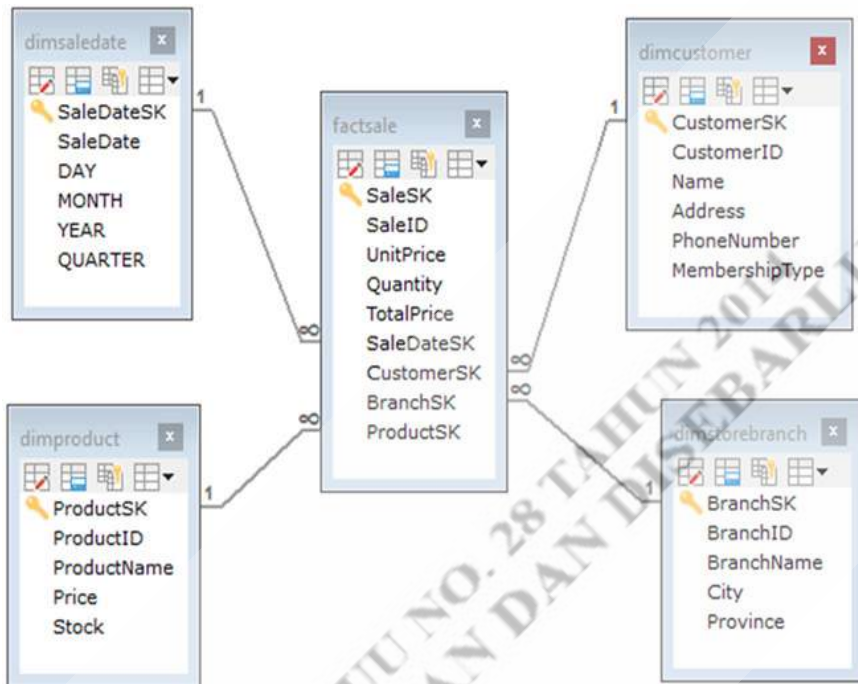
1. Sumber Data

Sebuah perusahaan retail dengan jaringan toko yang menjual berbagai produk elektronik dan menawarkan program membership kepada pelanggan memerlukan sistem manajemen penjualan yang dapat mengelola data pelanggan, membership, produk, penjualan, detail penjualan, serta informasi cabang toko termasuk kota dan provinsi. Sebagai solusi, telah dirancang sebuah database relasional bernama `retail_db` sebagai berikut:



Gambar 76. Sumber Data Sistem Penjualan

2. Desain Model Dimensional



Gambar 77. Model Dimensional Sistem Penjualan

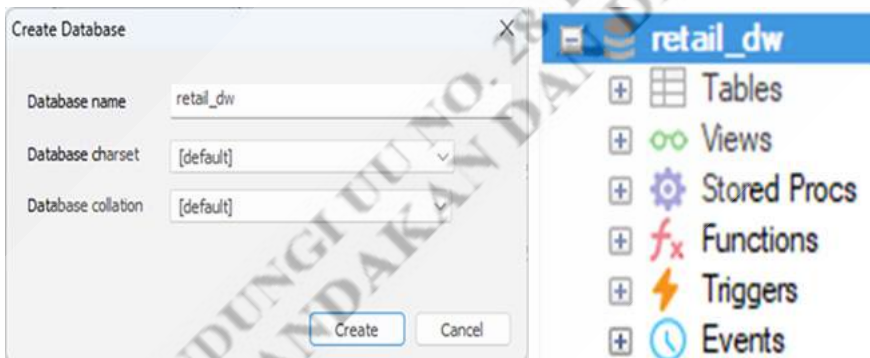
- DimCustomer, DimProduct, DimStoreBranch, dan DimSaleDate adalah tabel dimensi yang menyimpan informasi pelanggan, produk, cabang toko, dan tanggal penjualan, masing-masing dengan Primary Key yang sesuai.
- FactSale menyimpan rincian setiap penjualan dan menghubungkannya dengan tabel dimensi produk serta tabel fakta penjualan.
- Kolom Surrogate Key (SK) adalah kunci pengganti yang tidak memiliki makna bisnis dan biasanya berupa nilai numerik atau unik yang dihasilkan secara otomatis. Ini digunakan sebagai pengidentifikasi unik untuk setiap baris dalam tabel dimensi.

- d. Kolom SaleDateSK, CustomerSK, BranchSK, dan ProductSK menghubungkan tabel fakta dengan tabel dimensi terkait.
- e. Kolom UnitPrice, Quantity, dan TotalPrice menyimpan informasi rincian penjualan.

3. Membuat Koneksi Database

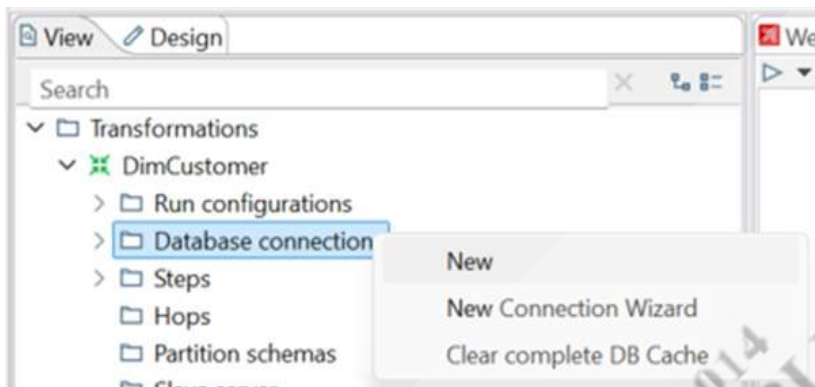
Database pada data warehouse digunakan untuk mengumpulkan dan menyimpan data dari berbagai sumber dalam format yang seragam dan terintegrasi. Data ini biasanya berasal dari sistem operasional, file, atau sumber lainnya. Pembuatan database pada MySQL dapat dilakukan dengan menggunakan SQLYog.

Buatlah database retail_dw menggunakan SQLYog dengan cara yang sudah dijelaskan pada bab sebelumnya.



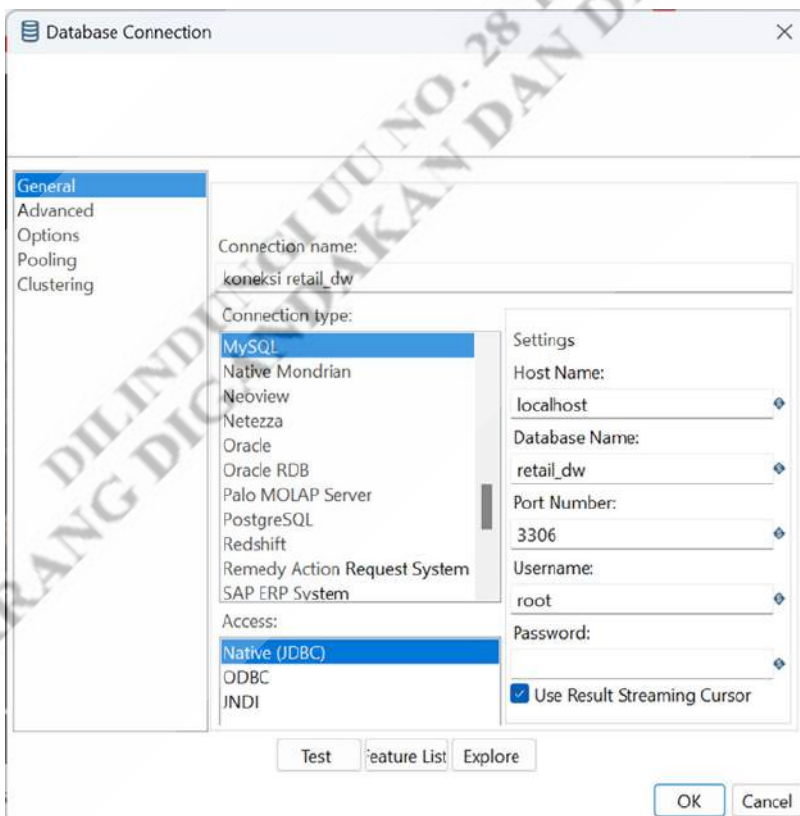
Gambar 78. Membuat Database retail_dw

Buatlah koneksi database untuk data warehouse menggunakan Kettle dengan cara klik kanan Database Connection pada tab View, lalu pilih New.



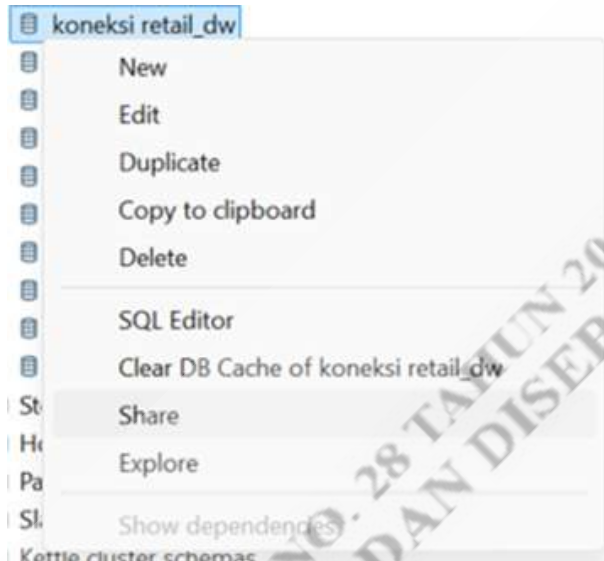
Gambar 79. Membuat Koneksi Database

Lakukan konfigurasi koneksi database untuk data warehouse sesuai Gambar 80.



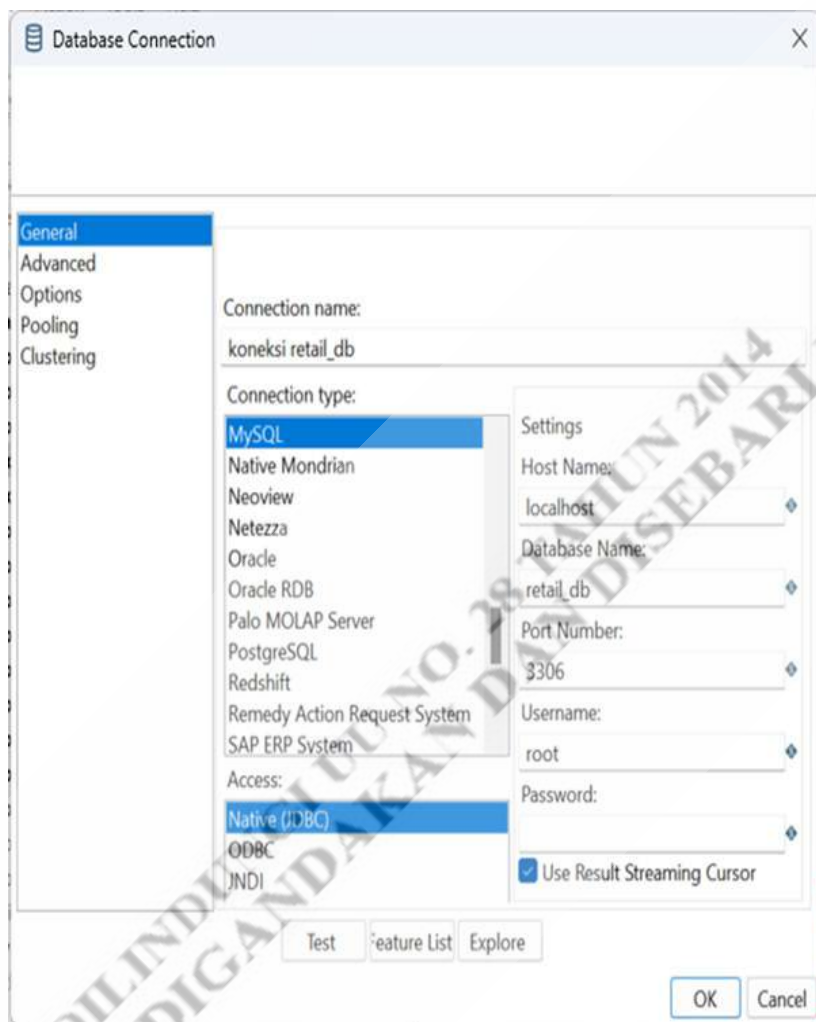
Gambar 80. Konfigurasi Koneksi Data Warehouse

Setelah koneksi database terbentuk, kemudian lakukan share koneksi dengan cara klik kanan nama koneksi pada Explore pane dan pilih menu Share.



Gambar 81. Menu Share Koneksi Data Warehouse

Buatlah koneksi database untuk sumber data berupa database bernama retail_db menggunakan Kettle dengan cara yang sama seperti sebelumnya dan lakukan konfigurasi koneksi database sesuai Gambar 82.

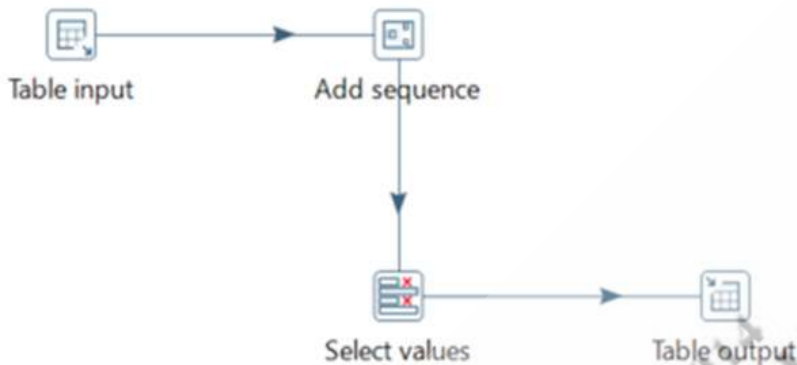


Gambar 82. Konfigurasi Koneksi Sumber Data

Setelah koneksi database terbentuk, kemudian lakukan share koneksi dengan cara yang sama seperti sebelumnya.

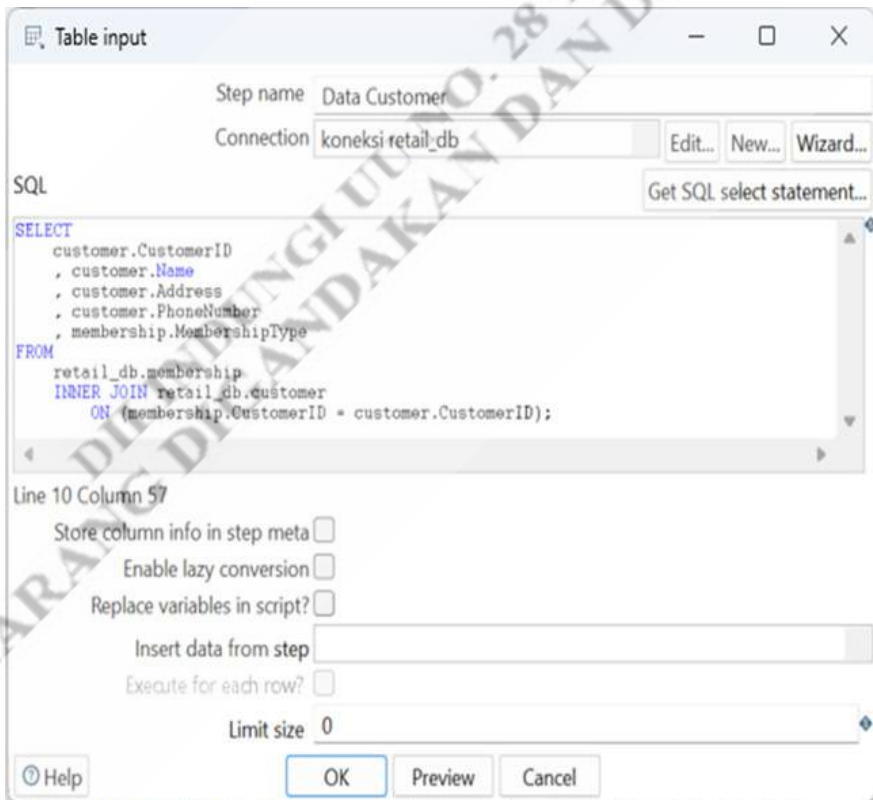
4. Membuat DimCustomer

Buat file transformation baru dengan cara pilih menu File – New – Transformation. Simpan transformation baru dalam file DimCustomer.ktr. Tambahkan beberapa step dan hop pada canvas seperti Gambar 83.



Gambar 83. Step dan Hop DimCustomer

Lakukan konfigurasi step pada DimCustomer seperti pada Gambar 84.



Gambar 84. Konfigurasi DimCustomer Step Tabel Input

Add sequence

Step name: SK_Customer

Name of value: CustomerSK

Use a database to generate the sequence

Use DB to get sequence? ☐

Connection: [Dropdown] Edit... New... Wizard...

Schema name: [Dropdown] Schemas...

Sequence name: SEQ_ Sequences...

Use a transformation counter to generate the sequence

Use counter to calculate sequence? ☒

Counter name (optional): [Text]

Start at value: 1

Increment by: 1

Maximum value: 999999999

Help OK Cancel

Gambar 85. Konfigurasi DimCustomer Step Add Sequence

Klik tombol Get fields to select pada konfigurasi step Select Values untuk memasukan seluruh nama field, lalu lakukan pemilihan dan pengurutan nama field sesuai kebutuhan.

Select values

Step name: Select values

Select & Alter Remove Meta-data

Fields:

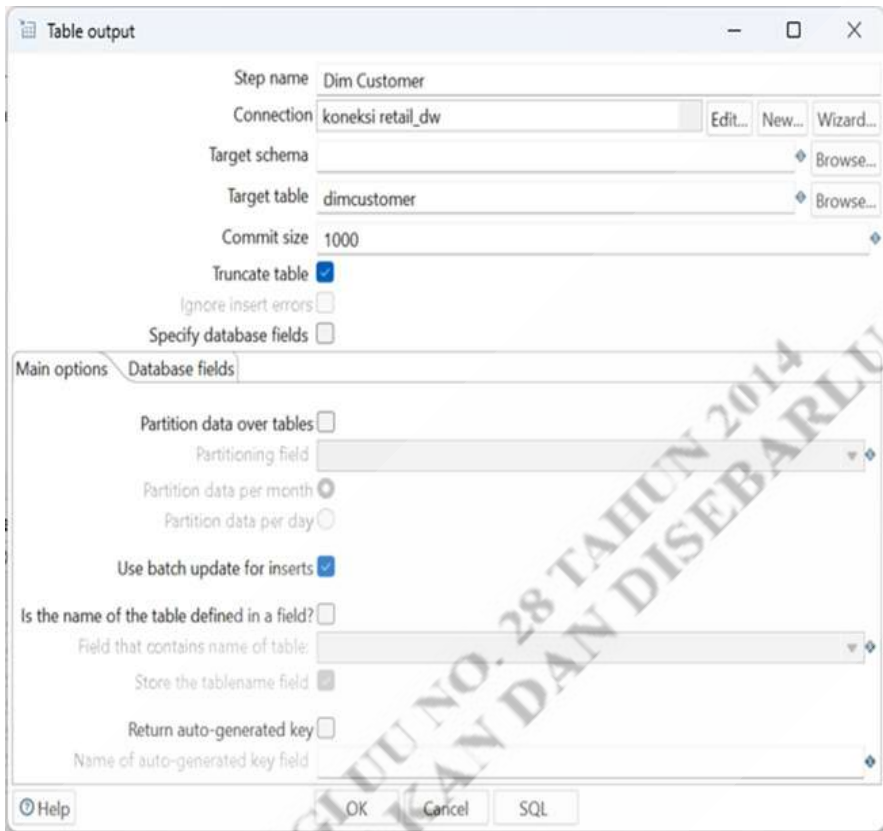
#	Fieldname	Rename to	Length	Precision
1	CustomerSK			
2	CustomerID			
3	Name			
4	Address			
5	PhoneNumber			
6	MembershipType			

Get fields to select Edit Mapping

Include unspecified fields. ☐

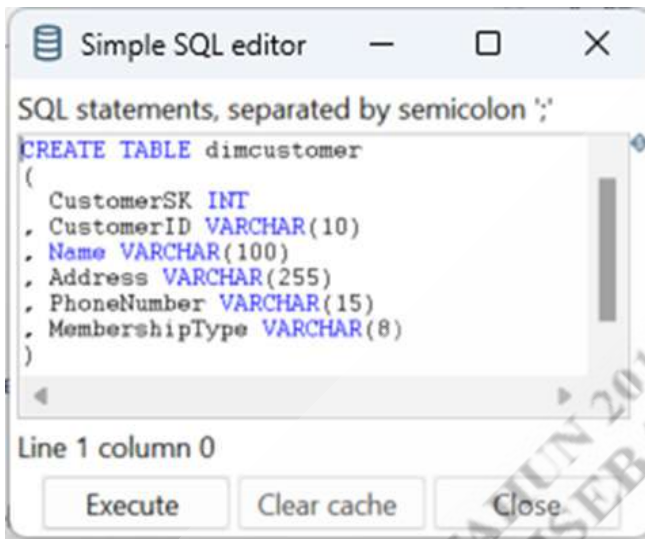
Help OK Cancel

Gambar 86. Konfigurasi DimCustomer Step Select Values



Gambar 87. Konfigurasi DimCustomer Step Table Output

Setelah memilih connection dan mengisi target table pada konfigurasi step Table Output, selanjutnya klik tombol SQL untuk membuat tabel dimcustomer.



Gambar 88. SQL Create Tabel Dimcustomer

Klik tombol Execute untuk membuat tabel dimcustomer, lalu klik tombol Close untuk mengakhiri. Klik tombol OK pada konfigurasi Table Output, kemudian simpan file dan jalankan transformation yang sudah dibuat.



Gambar 89. Transformation DimCustomer Berhasil Dijalankan

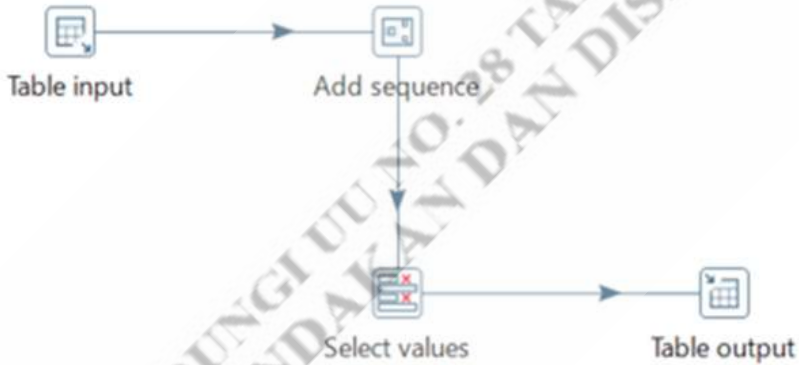
Transformation berhasil dijalankan dan tabel dimcustomer sudah berisi data. SQLYog dapat digunakan untuk melihat data yang telah tersimpan pada tabel dimcustomer.

CustomerSK	CustomerID	Name	Address	PhoneNumber	MembershipType
1	C0001	Alice Johnson	123 Elm St, Jakarta	061234567890	Platinum
2	C0002	Bob Smith	456 Oak St, Surabaya	062345678901	Silver
3	C0003	Charlie Brown	789 Pine St, Jakarta	063456789012	Gold
4	C0004	David Wilson	135 Maple St, Bandung	064567890123	Gold
5	C0005	Eva Davis	246 Cedar St, Medan	065678901234	Silver

Gambar 90. Data Tabel DimCustomer

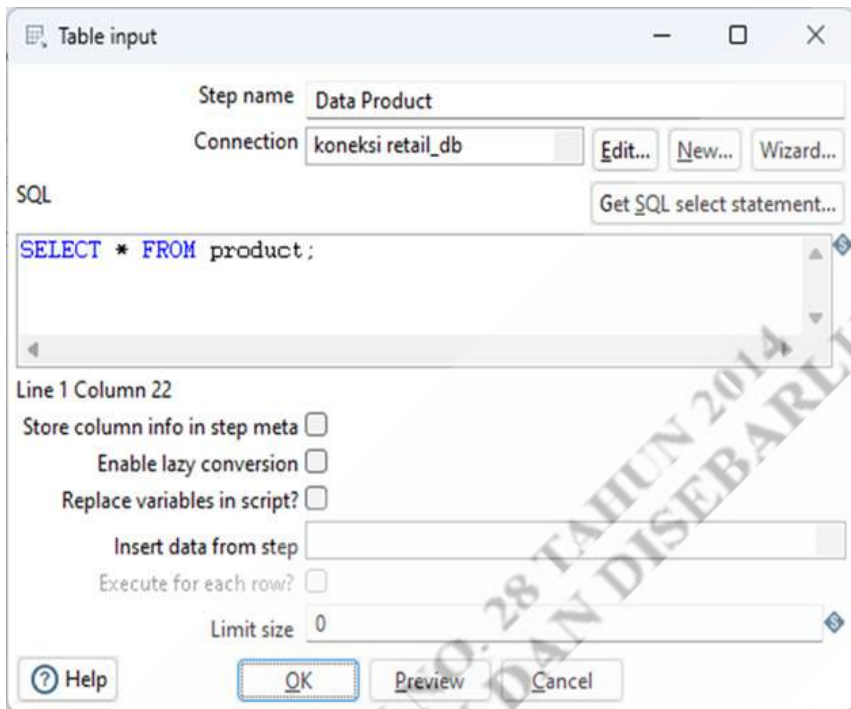
5. Membuat DimProduct

Buat file transformation baru dengan cara pilih menu File – New – Transformation. Simpan transformation baru dalam file DimProduct.ktr. Tambahkan beberapa step dan hop pada canvas seperti Gambar 91



Gambar 91. Step dan Hop DimProduct

Lakukan konfigurasi step pada DimProduct seperti pada Gambar 92.



Gambar 92. Konfigurasi DimProduct Step Tabel Input

Add sequence

Step name: SK_Product

Name of value: ProductSK

Use a database to generate the sequence

Use DB to get sequence? ☐

Connection: [Dropdown] [Edit...] [New...] [Wizard...]

Schema name: [Text] [Schemas...]

Sequence name: SEQ_ [Sequences...]

Use a transformation counter to generate the sequence

Use counter to calculate sequence? ☒

Counter name (optional): [Text]

Start at value: 1

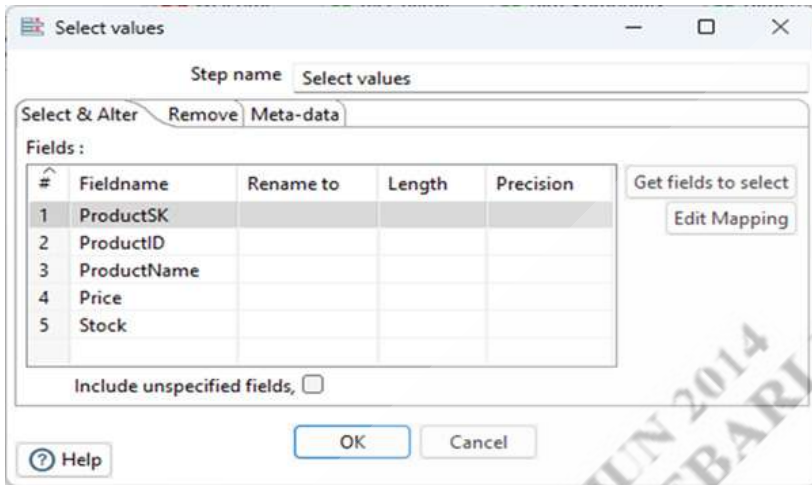
Increment by: 1

Maximum value: 999999999

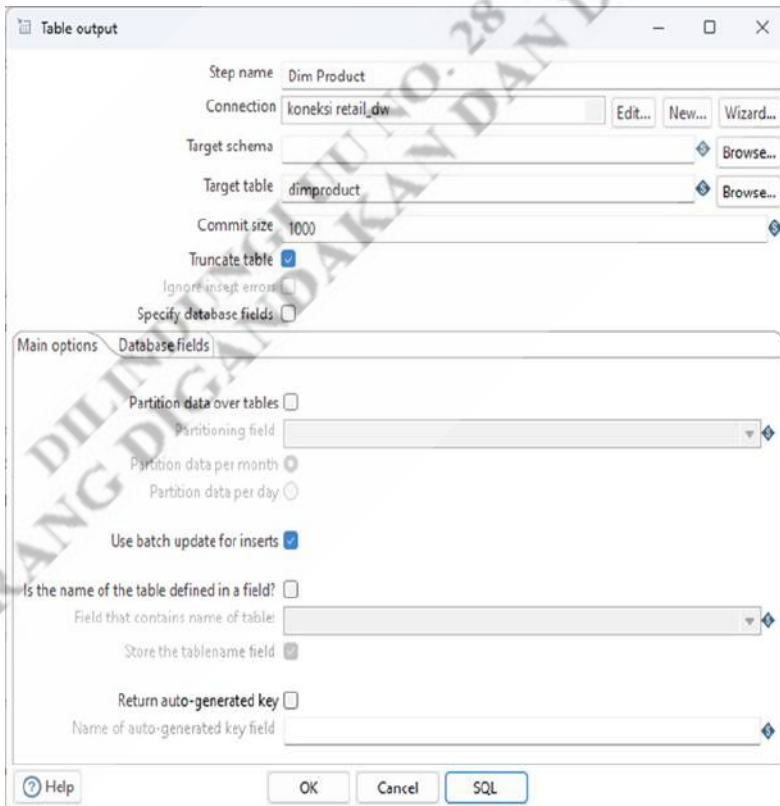
[Help] [OK] [Cancel]

Gambar 93. Konfigurasi DimProduct Step Add Sequence

Klik tombol Get fields to select pada konfigurasi step Select Values untuk memasukan seluruh nama field, lalu lakukan pemilihan dan pengurutan nama field sesuai kebutuhan.

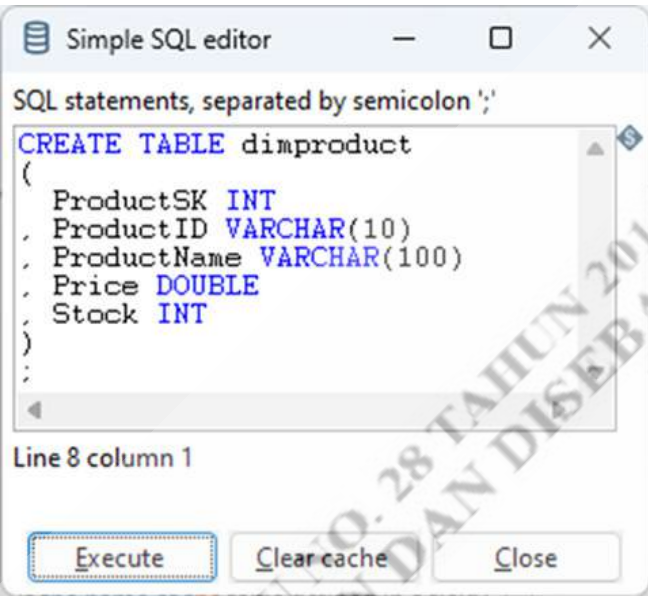


Gambar 94. Konfigurasi DimProduct Step Select Values



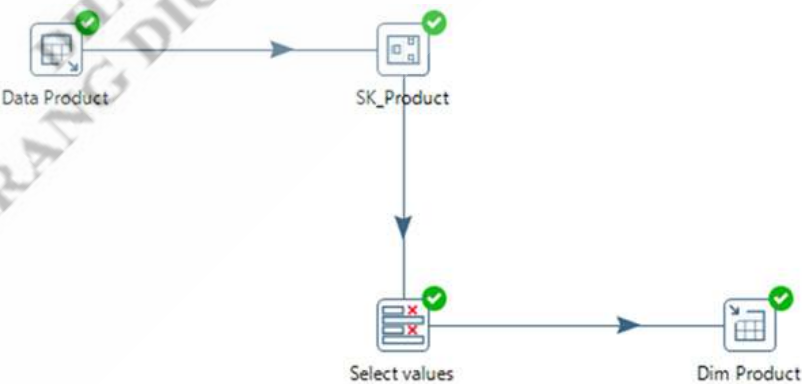
Gambar 95. Konfigurasi DimProduct Step Table Output

Setelah memilih connection dan mengisi target table pada konfigurasi step Table Output, selanjutnya klik tombol SQL untuk membuat tabel dimproduct.



Gambar 96. SQL Create Tabel Dimproduct

Klik tombol Execute untuk membuat tabel dimproduct, lalu klik tombol Close untuk mengakhiri. Klik tombol OK pada konfigurasi Table Output, kemudian simpan file dan jalankan transformation yang sudah dibuat.



Gambar 97. Transformation DimProduct Berhasil Dijalankan

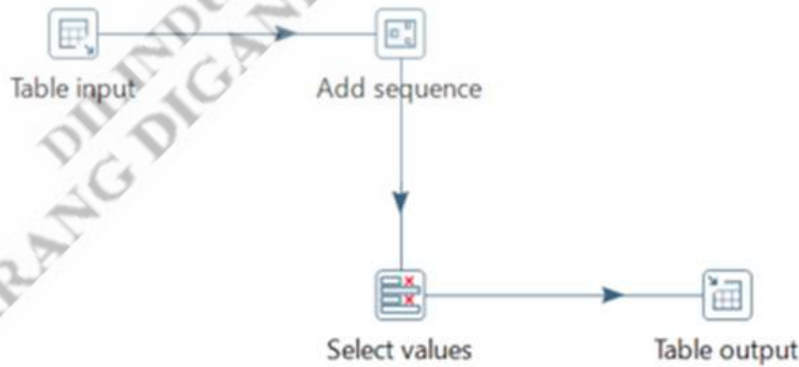
Transformation berhasil dijalankan dan tabel dimproduct sudah berisi data. SQLYog dapat digunakan untuk melihat data yang telah tersimpan pada tabel dimproduct.

ProductSK	ProductID	ProductName	Price	Stock
1	P0001	Television 40"	4000	50
2	P0002	Washing Machine	3000	30
3	P0003	Refrigerator 300L	2500	20
4	P0004	Laptop 15"	5000	15
5	P0005	Smartphone	2000	100
6	P0006	Microwave Oven	1500	40
7	P0007	Air Conditioner	6000	25
8	P0008	Blender	500	60
9	P0009	Electric Kettle	250	70
10	P0010	Dishwasher	3500	10

Gambar 98. Data Tabel Dimproduct

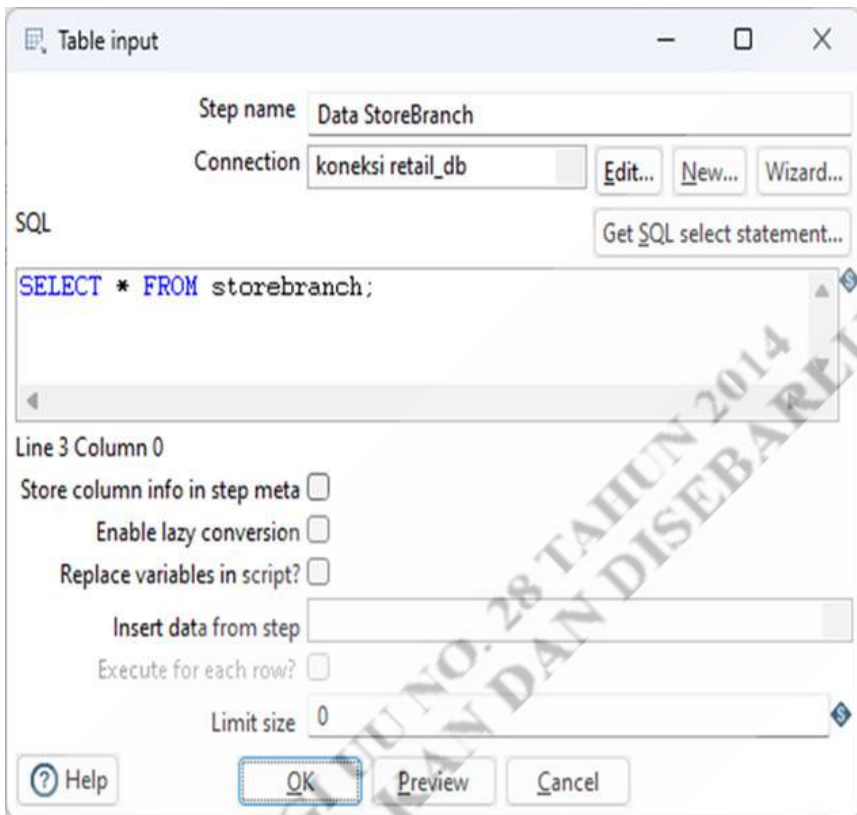
6. Membuat DimStoreBranch

Buat file transformation baru dengan cara pilih menu File – New – Transformation. Simpan transformation baru dalam file DimStoreBranch.ktr. Tambahkan beberapa step dan hop pada canvas seperti Gambar 99.



Gambar 99. Step dan Hop DimStoreBranch

Lakukan kofigurasi step pada Dim Store Branch seperti pada Gambar 100.



Gambar 100. Konfigurasi DimStoreBranch Step Tabel Input

Add sequence

Step name: SK_StoreBranch

Name of value: StoreBranchSK

Use a database to generate the sequence

Use DB to get sequence? ☐

Connection: [Dropdown] [Edit...] [New...] [Wizard...]

Schema name: [Text] [Schemas...]

Sequence name: SEQ [Sequences...]

Use a transformation counter to generate the sequence

Use counter to calculate sequence? ☒

Counter name (optional): [Text]

Start at value: 1

Increment by: 1

Maximum value: 999999999

[Help] [OK] [Cancel]

Gambar 101. Konfigurasi DimStoreBranch Step Add Sequence

Klik tombol Get fields to select pada konfigurasi step Select Values untuk memasukan seluruh nama field, lalu lakukan pemilihan dan pengurutan nama field sesuai kebutuhan.

Select values

Step name: Select values

Select & Alter Remove Meta-data

Fields:

#	Fieldname	Rename to	Length	Precision
1	StoreBranchSK			
2	BranchID			
3	BranchName			
4	City			
5	Province			

[Get fields to select] [Edit Mapping]

Include unspecified fields, ☐

[Help] [OK] [Cancel]

Gambar 102. Konfigurasi DimStoreBranch Step Select Values

The screenshot shows the 'Table output' configuration window for a step named 'Dim StoreBranch'. The window is divided into two main sections: 'Main options' and 'Database fields'.

Main options:

- Step name: Dim StoreBranch
- Connection: koneksi retail_dw (with Edit..., New..., and Wizard... buttons)
- Target schema: (empty field with a Browse... button)
- Target table: dimstorebranch (with a Browse... button)
- Commit size: 1000
- Truncate table: ☒
- Ignore insert errors: ☐
- Specify database fields: ☐

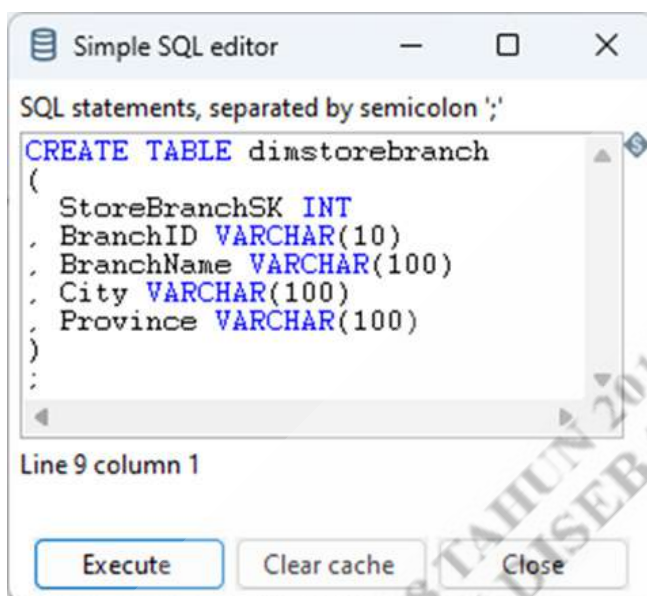
Database fields:

- Partition data over tables: ☐
- Partitioning field: (empty dropdown menu)
- Partition data per month: ☐
- Partition data per day: ☐
- Use batch update for inserts: ☒
- Is the name of the table defined in a field?: ☐
- Field that contains name of table: (empty dropdown menu)
- Store the tablename field: ☒
- Return auto-generated key: ☐
- Name of auto-generated key field: (empty text field)

At the bottom, there are buttons for Help, OK, Cancel, and SQL.

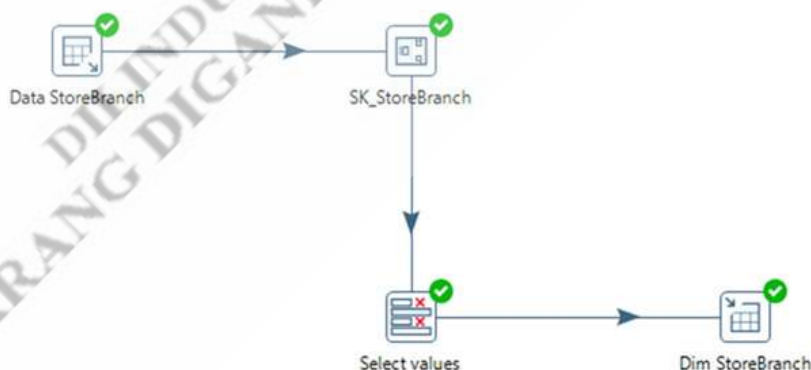
Gambar 103. Konfigurasi DimStoreBranch Step Table Output

Setelah memilih connection dan mengisi target table pada konfigurasi step Table Output, selanjutnya klik tombol SQL untuk membuat tabel dimstorebranch.



Gambar 104. SQL Create Tabel Dimstorebranch

Klik tombol Execute untuk membuat tabel dimstorebranch, lalu klik tombol Close untuk mengakhiri. Klik tombol OK pada konfigurasi Table Output, kemudian simpan file dan jalankan transformation yang sudah dibuat.



Gambar 105. Transformation DimStoreBranch Berhasil Dijalankan

Transformation berhasil dijalankan dan tabel dimstorebranch sudah berisi data. SQLYog dapat digunakan untuk melihat data

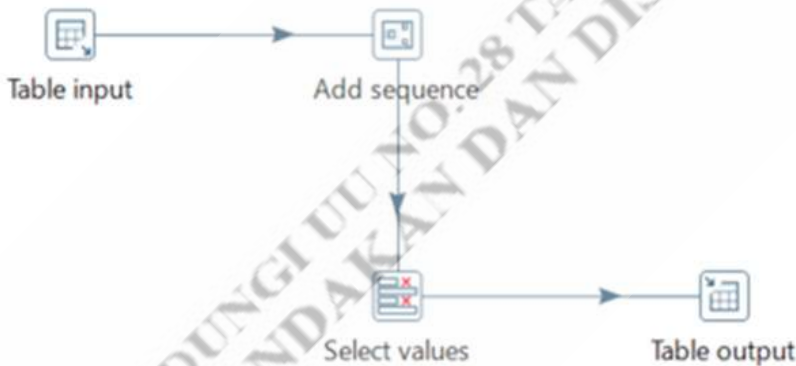
yang telah tersimpan pada tabel dimstorebranch.

StoreBranchSK	BranchID	BranchName	City	Province
1	B0001	Main Branch	Jakarta	DKI Jakarta
2	B0002	East Branch	Surabaya	Jawa Timur
3	B0003	West Branch	Bandung	Jawa Barat
4	B0004	North Branch	Medan	Sumatera Utara
5	B0005	South Branch	Makassar	Sulawesi Selatan

Gambar 106. Data Tabel Dimstorebranch

7. Membuat DimSaleDate

Buat file transformation baru dengan cara pilih menu File – New – Transformation. Simpan transformation baru dalam file DimSaleDate.ktr. Tambahkan beberapa step dan hop pada canvas seperti Gambar 107.



Gambar 107. Step dan Hop DimSaleDate

Lakukan konfigurasi step pada DimSaleDate seperti pada Gambar 108.

Tulis SQL berikut ini pada Tabel Input untuk membuat sumber data berupa data tanggal antara 1 Januari 2020 sampai 31 Desember 2050.

SELECT

d	AS	SaleDate,
DAY(d)	AS	DAY,
MONTH(d)	AS	MONTH,
YEAR(d)	AS	YEAR,
QUARTER(d)	AS QUARTER	

FROM (
SELECT '2020-01-01' + INTERVAL (a.a + (10 * b.a) + (100 * c.a) + (1000 * d.a) + (10000 * e.a)) DAY AS d

FROM (SELECT 0 AS a UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4 UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION ALL SELECT 9) a

CROSS JOIN (SELECT 0 AS a UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4 UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION ALL SELECT 9) b

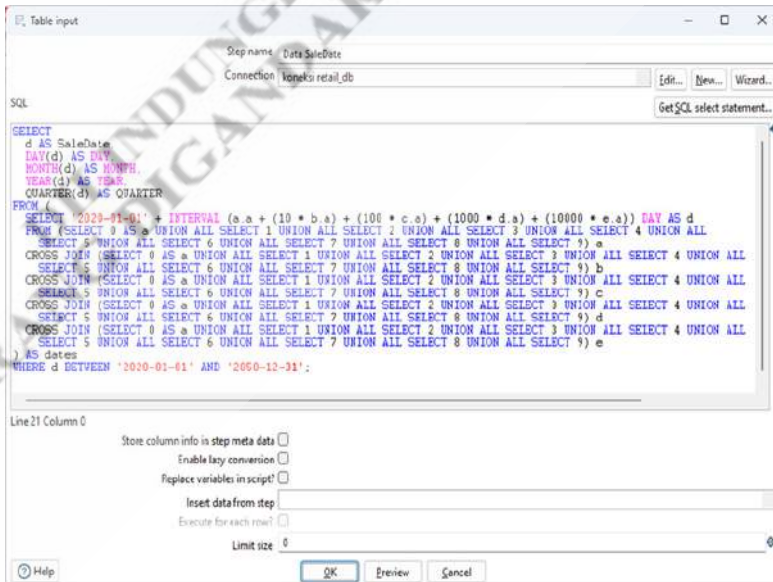
CROSS JOIN (SELECT 0 AS a UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4 UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION ALL SELECT 9) c

CROSS JOIN (SELECT 0 AS a UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4 UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION ALL SELECT 9) d

CROSS JOIN (SELECT 0 AS a UNION ALL SELECT 1 UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4 UNION ALL SELECT 5 UNION ALL SELECT 6 UNION ALL SELECT 7 UNION ALL SELECT 8 UNION ALL SELECT 9) e

) AS dates

WHERE d BETWEEN '2020-01-01' AND '2050-12-31';



Gambar 108. Konfigurasi DimSaleDate Step Tabel Input

Add sequence

Step name: SK_SaleDate

Name of value: SaleDateSK

Use a database to generate the sequence

Use DB to get sequence? ☐

Connection: [Dropdown] [Edit...] [New...] [Wizard...]

Schema name: [Text] [Schemas...]

Sequence name: SEQ_ [Sequences...]

Use a transformation counter to generate the sequence

Use counter to calculate sequence? ☒

Counter name (optional): [Text]

Start at value: 1

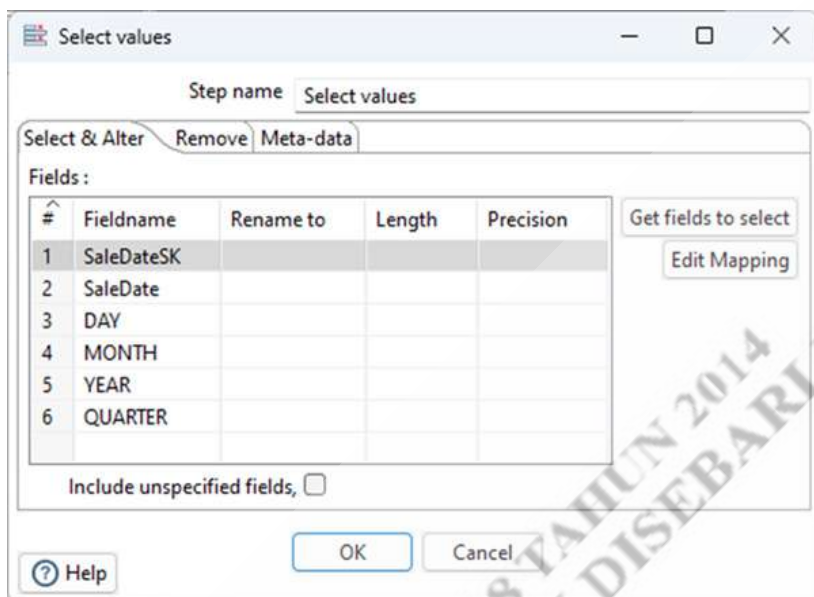
Increment by: 1

Maximum value: 999999999

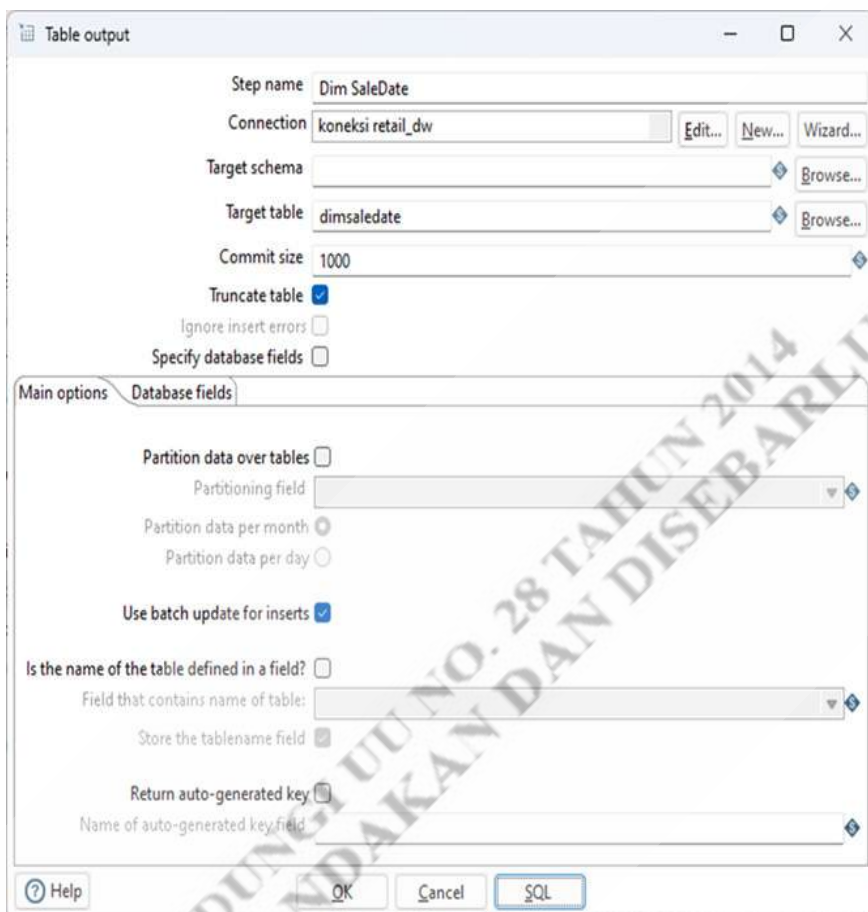
[Help] [OK] [Cancel]

Gambar 109. Konfigurasi DimSaleDate Step Add Sequence

Klik tombol Get fields to select pada konfigurasi step Select Values untuk memasukan seluruh nama field, lalu lakukan pemilihan dan pengurutan nama field sesuai kebutuhan.

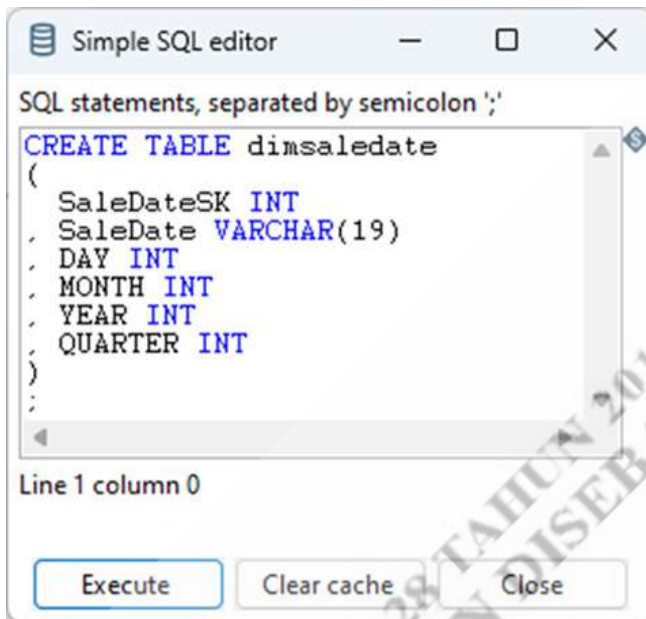


Gambar 110. Konfigurasi DimSaleDate Step Select Values



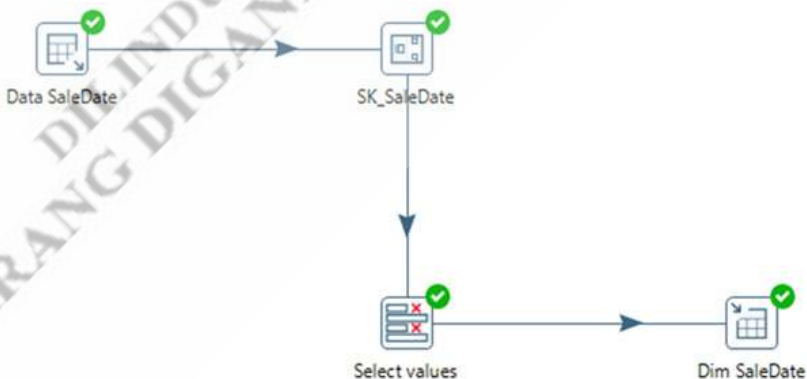
Gambar 111. Konfigurasi DimSaleDate Step Table Output

Setelah memilih connection dan mengisi target table pada konfigurasi step Table Output, selanjutnya klik tombol SQL untuk membuat tabel dimsaledate.



Gambar 112. SQL Create Tabel Dimsaledate

Klik tombol Execute untuk membuat tabel dimsaledate, lalu klik tombol Close untuk mengakhiri. Klik tombol OK pada konfigurasi Table Output, kemudian simpan file dan jalankan transformation yang sudah dibuat.



Gambar 113. Transformation DimSaleDate Berhasil Dijalankan

Transformation berhasil dijalankan dan tabel dimsaledate sudah berisi data. SQLYog dapat digunakan untuk melihat data

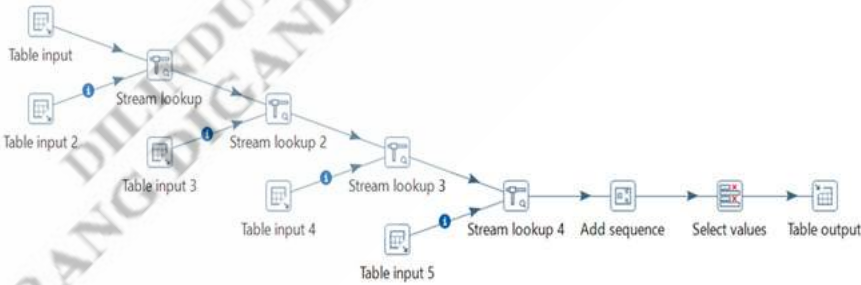
yang telah tersimpan pada tabel `dimsaledate`.

SaleDateSK	SaleDate	DAY	MONTH	YEAR	QUARTER
1	2020-01-01	1	1	2020	1
2	2020-01-02	2	1	2020	1
3	2020-01-03	3	1	2020	1
4	2020-01-04	4	1	2020	1
5	2020-01-05	5	1	2020	1
6	2020-01-06	6	1	2020	1
7	2020-01-07	7	1	2020	1
8	2020-01-08	8	1	2020	1
9	2020-01-09	9	1	2020	1
10	2020-01-10	10	1	2020	1
11	2020-01-11	11	1	2020	1
12	2020-01-12	12	1	2020	1
13	2020-01-13	13	1	2020	1

Gambar 114. Data Tabel `Dimsaledate`

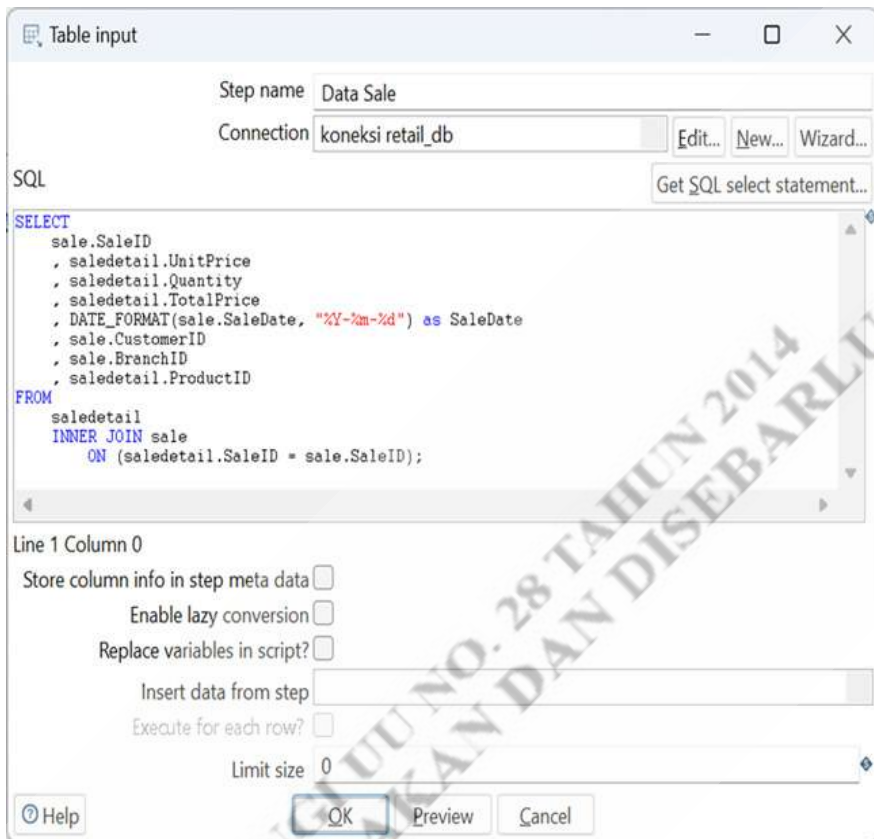
8. Membuat FactSale

Buat file transformation baru dengan cara pilih menu `File – New – Transformation`. Simpan transformation baru dalam file `FactSale.ktr`. Tambahkan beberapa step dan hop pada canvas seperti gambar berikut:

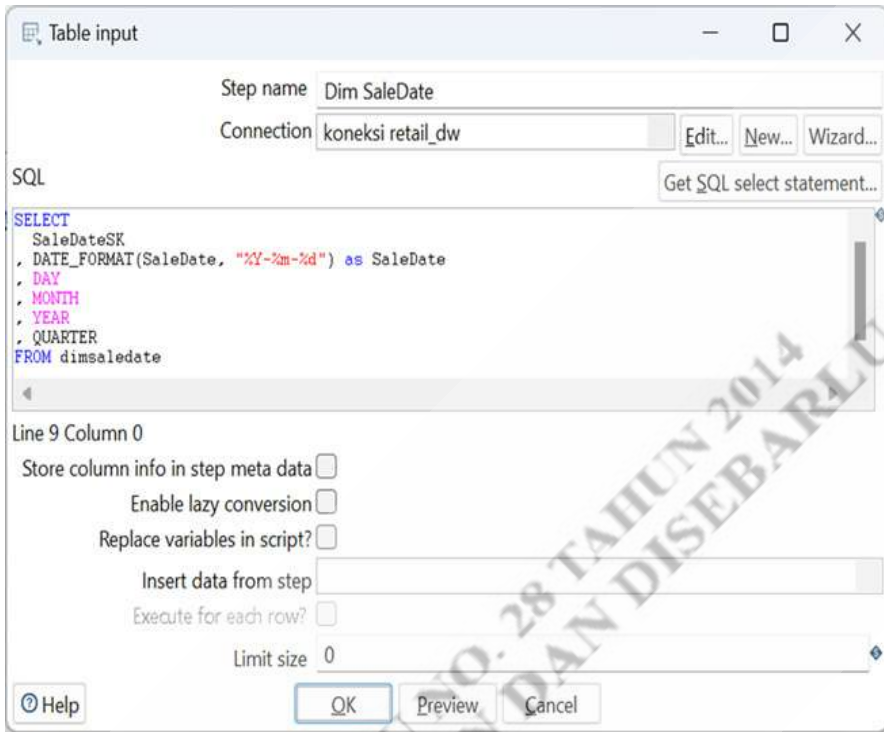


Gambar 115. Step dan Hop FactSale

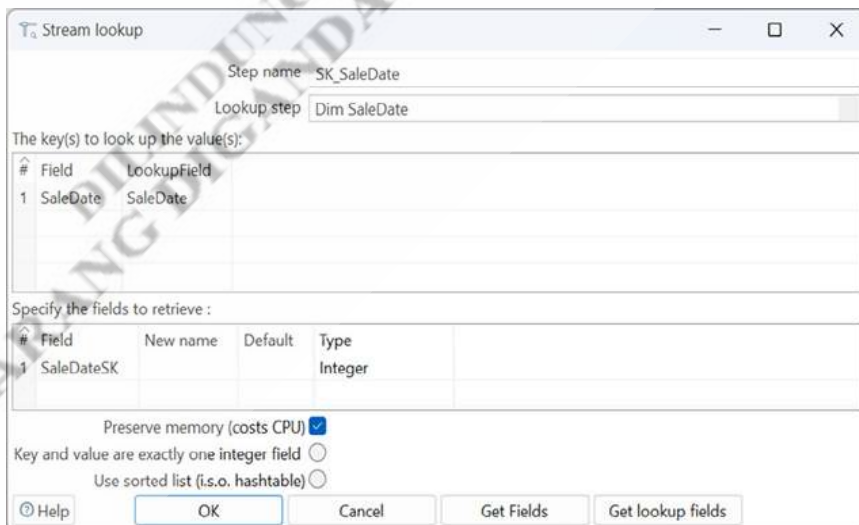
Lakukan konfigurasi step pada Fact Sale seperti pada Gambar 116.



Gambar 116. Konfigurasi FactSale Step Tabel Input



Gambar 117. Konfigurasi FactSale Step Tabel Input 2



Gambar 118. Konfigurasi FactSale Step Stream Lookup

Table input

Step name: Dim Customer

Connection: koneksi retail_dw

SQL

Get SQL select statement...

```
SELECT
  CustomerSK
  CustomerID
  Name
  Address
  PhoneNumber
  MembershipType
FROM dimcustomer
```

Line 9 Column 0

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step

Execute for each row? ☐

Limit size: 0

Help OK Preview Cancel

Gambar 119. Konfigurasi FactSale Step Tabel Input 3

Stream lookup

Step name: SK_Customer

Lookup step: Dim Customer

The key(s) to look up the value(s):

#	Field	LookupField
1	CustomerID	CustomerID

Specify the fields to retrieve:

#	Field	New name	Default	Type
1	CustomerSK			Integer

Preserve memory (costs CPU) ☒

Key and value are exactly one integer field ☐

Use sorted list (i.s.o. hashtable) ☐

Help OK Cancel Get Fields Get lookup fields

Gambar 120. Konfigurasi FactSale Step Stream Lookup 2

Table input

Step name: Dim StoreBranch

Connection: koneksi retail_dw

SQL: `SELECT StoreBranchSK, BranchID, BranchName, City, Province FROM dimstorebranch`

Line 8 Column 0

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step:

Execute for each row? ☐

Limit size: 0

Help OK Preview Cancel

Gambar 121. Konfigurasi FactSale Step Tabel Input 4

Stream lookup

Step name: SK_StoreBranch

Lookup step: Dim StoreBranch

The key(s) to look up the value(s):

#	Field	LookupField
1	BranchID	BranchID

Specify the fields to retrieve:

#	Field	New name	Default	Type
1	StoreBranchSK			Integer

Preserve memory (costs CPU) ☐

Key and value are exactly one integer field ☐

Use sorted list (i.s.o. hashtable) ☐

Help OK Cancel Get Fields Get lookup fields

Gambar 122. Konfigurasi FactSale Step Stream Lookup 3

Table input

Step name: Dim Product

Connection: koneksi retail_dw

SQL: `SELECT ProductSK, ProductID, ProductName, Price, Stock FROM dimproduct`

Line 8 Column 0

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step

Execute for each row? ☐

Limit size: 0

Help OK Preview Cancel

Gambar 123. Konfigurasi FactSale Step Tabel Input 5

Stream lookup

Step name: SK_Product

Lookup step: Dim Product

The key(s) to look up the value(s):

#	Field	LookupField
1	ProductID	ProductID

Specify the fields to retrieve :

#	Field	New name	Default	Type
1	ProductSK			Integer

Preserve memory (costs CPU) ☐

Key and value are exactly one integer field ☐

Use sorted list (i.s.o. hashtable) ☐

Help OK Cancel Get Fields Get lookup fields

Gambar 124. Konfigurasi FactSale Step Stream Lookup 4

Add sequence

Step name: SK_Sale

Name of value: SaleSK

Use a database to generate the sequence

Use DB to get sequence? ☐

Connection: [Dropdown] [Edit...] [New...] [Wizard...]

Schema name: [Text] [Schemas...]

Sequence name: SEQ_ [Sequences...]

Use a transformation counter to generate the sequence

Use counter to calculate sequence? ☒

Counter name (optional): [Text]

Start at value: 1

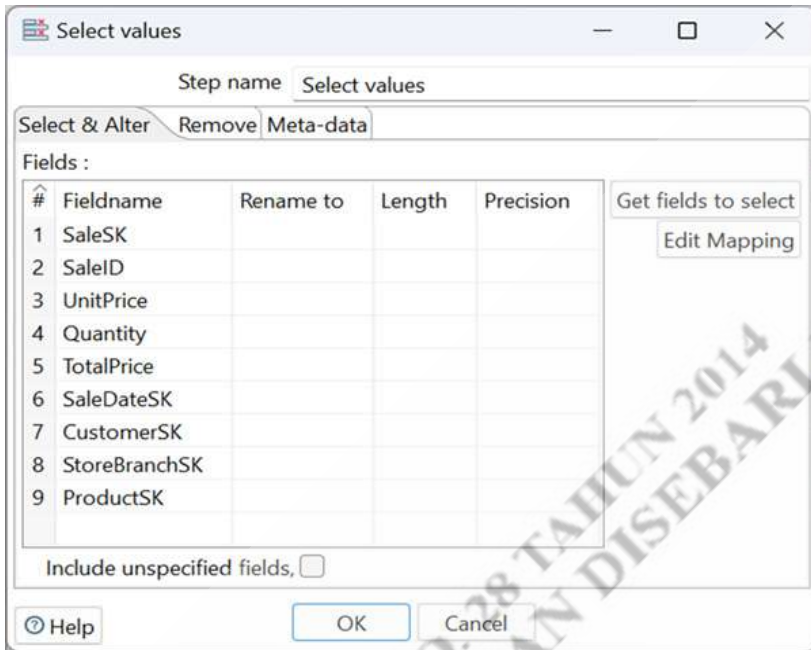
Increment by: 1

Maximum value: 999999999

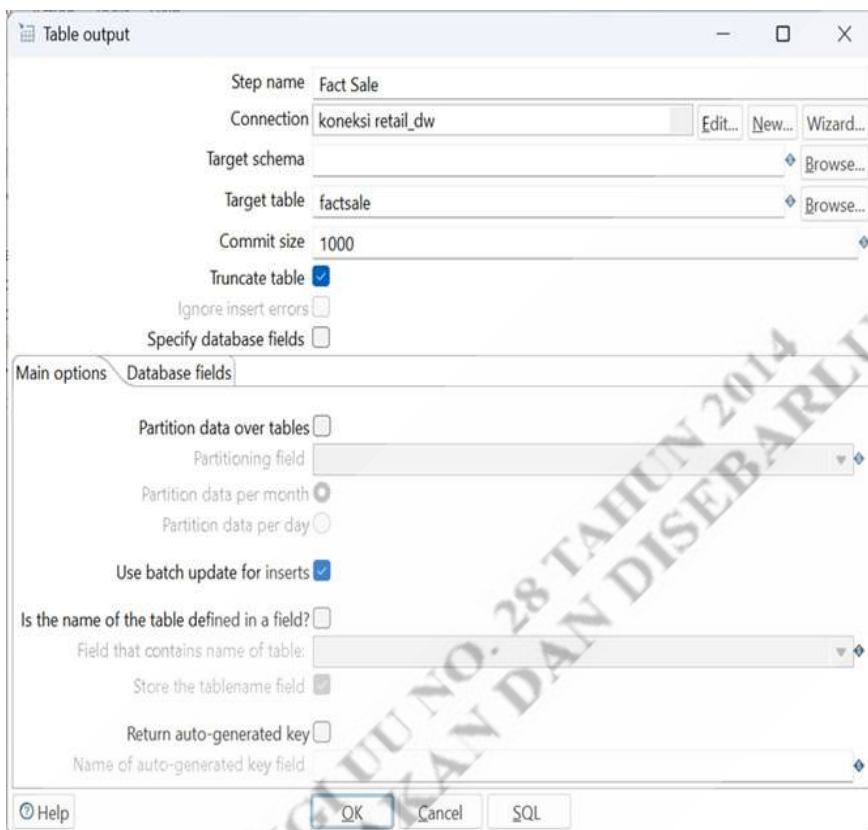
[Help] [OK] [Cancel]

Gambar 125. Konfigurasi FactSale Step Add Sequence

Klik tombol Get fields to select pada konfigurasi step Select Values untuk memasukan seluruh nama field, lalu lakukan pemilihan dan pengurutan nama field sesuai kebutuhan.

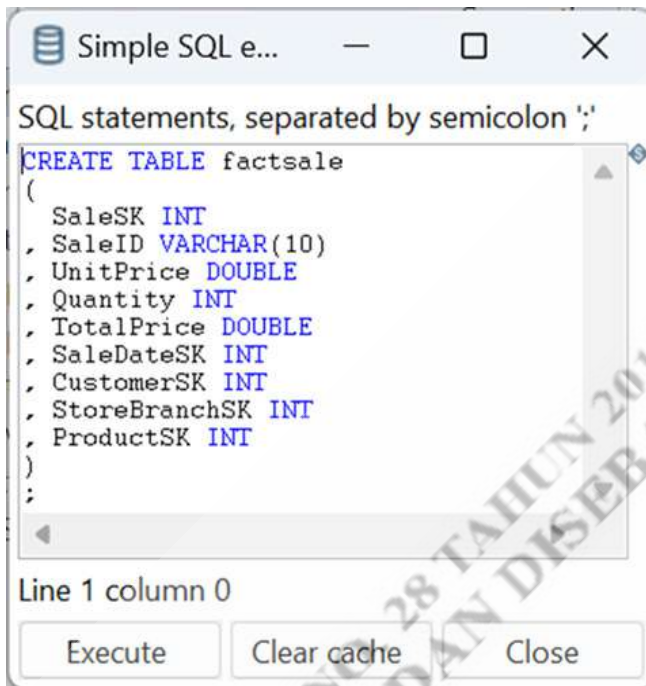


Gambar 126. Konfigurasi FactSale Step Select Values



Gambar 127. Konfigurasi FactSale Step Table Output

Setelah memilih connection dan mengisi target table pada konfigurasi step Table Output, selanjutnya klik tombol SQL untuk membuat tabel factsale.



Gambar 128. SQL Create Tabel Factsale

Klik tombol Execute untuk membuat tabel factsale, lalu klik tombol Close untuk mengakhiri. Klik tombol OK pada konfigurasi Table Output, kemudian simpan file dan jalankan transformation yang sudah dibuat.



Gambar 129. Transformation FactSale Berhasil Dijalankan

Transformation berhasil dijalankan dan tabel factsale sudah berisi data. SQLYog dapat digunakan untuk melihat data yang telah tersimpan pada tabel factsale.

SaleSK	SaleID	UnitPrice	Quantity	TotalPrice	SaleDateSK	CustomerSK	StoreBranchSK	ProductSK
1	S0001	4000	1	4000	1101	1	1	1
2	S0001	5000	1	5000	1101	1	1	4
3	S0001	2000	2	4000	1101	1	1	5
4	S0001	1500	1	1500	1101	1	1	6
5	S0002	3000	1	3000	1108	2	2	2
6	S0002	2000	1	2000	1108	2	2	5
7	S0002	6000	1	6000	1108	2	2	7
8	S0002	500	2	1000	1108	2	2	8
9	S0003	2500	1	2500	1111	3	3	3
10	S0003	2000	2	4000	1111	3	3	5
11	S0003	1500	1	1500	1111	3	3	6
12	S0003	250	3	750	1111	3	3	9
13	S0004	4000	1	4000	1116	4	4	1

Gambar 130. Data Tabel Factsale

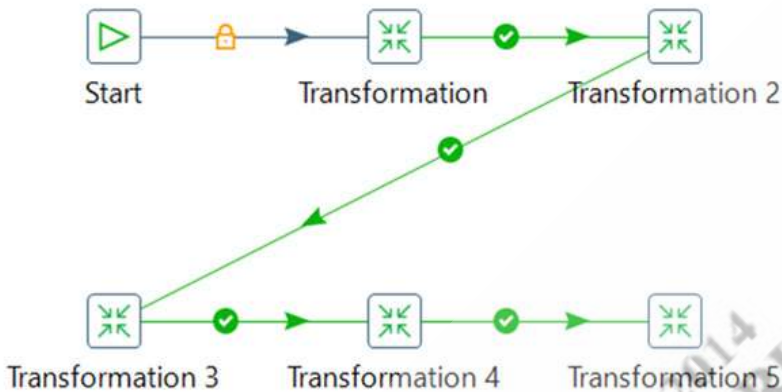
Terdapat 4 tabel dimensi dan 1 tabel fakta yang sudah dibuat dan diisi di data warehouse dengan nama database retail_db. Informasi database retail_db dapat dilihat pada tabel berikut ini.

Tabel 5. Informasi Database retail_db

Table Name	Rows	Data Size
dimcustomer	5	16KB
dimproduct	10	16KB
dimsaledate	11180	1.03MB
dimstorebranch	5	16KB
factsale	240	48KB

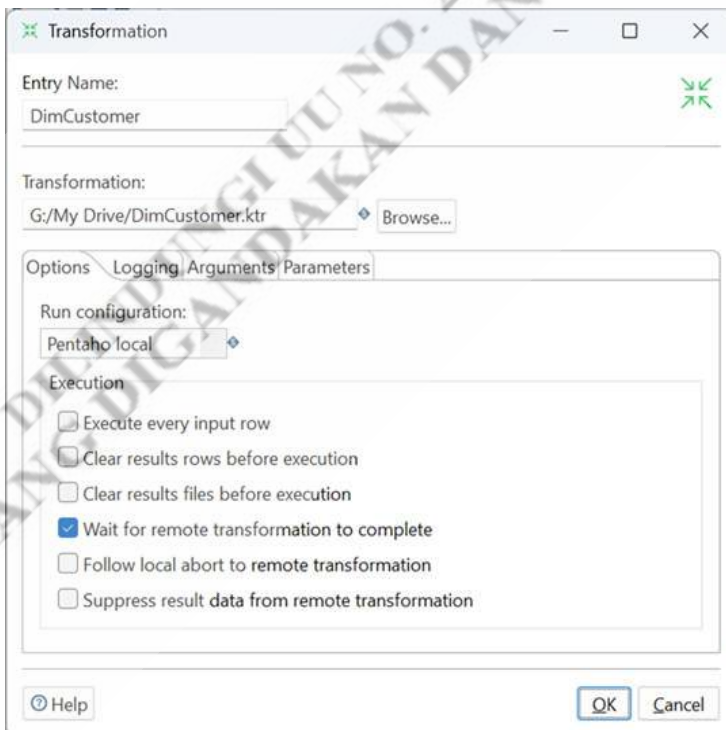
9. Membuat JobSale

Buat file job baru dengan cara pilih menu File – New – Job. Simpan job baru dalam file JobSale.kjb. Tambahkan beberapa step dan hop pada canvas seperti Gambar 131.

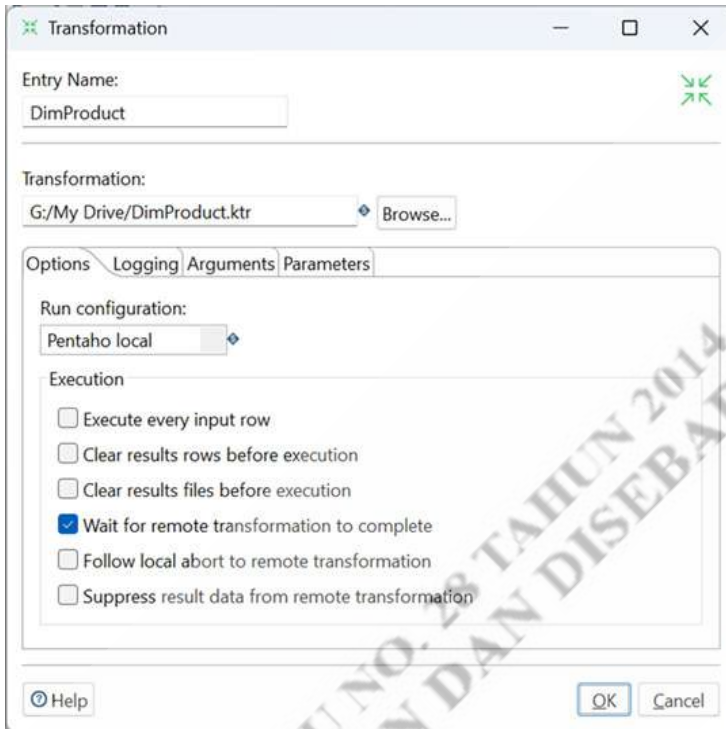


Gambar 131. Step dan Hop JobSale

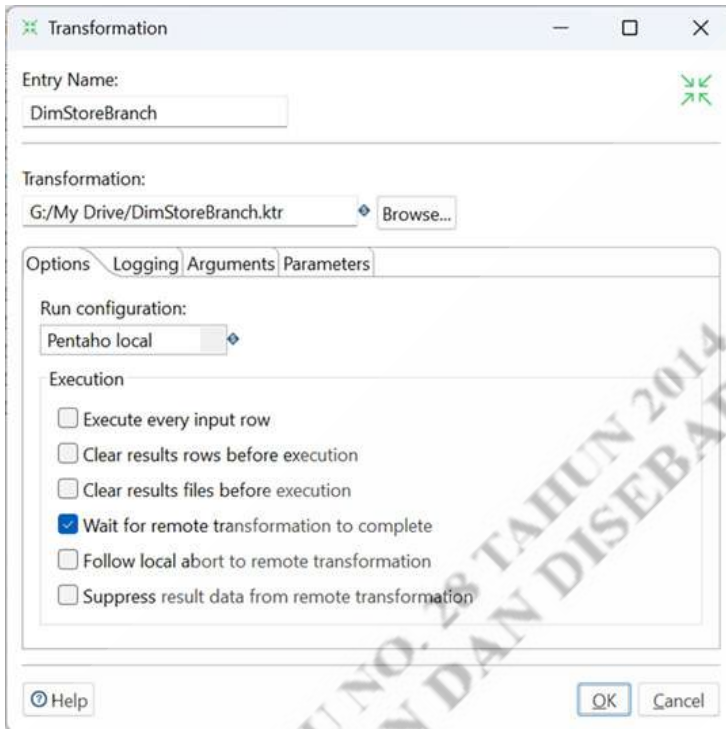
Lakukan konfigurasi step pada JobSale seperti pada Gambar 132.



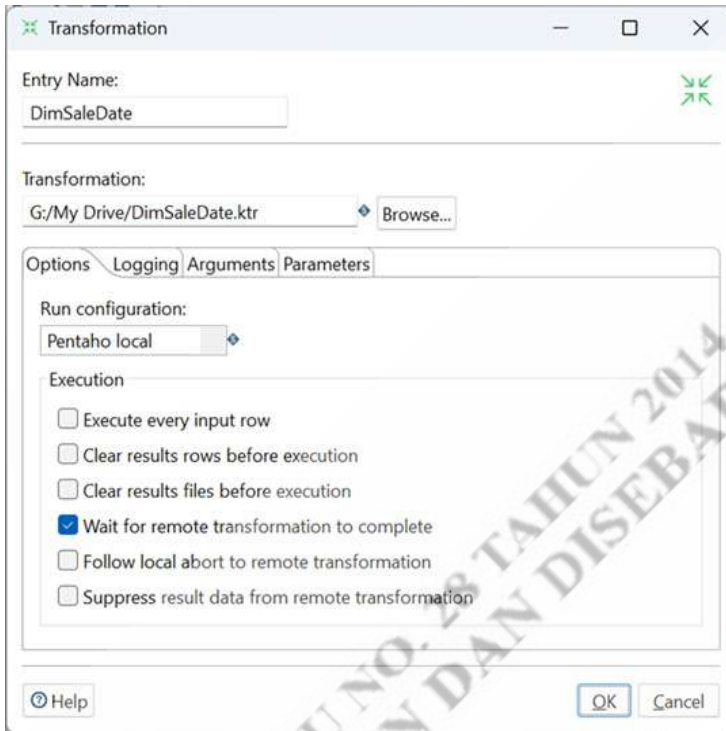
Gambar 132. Konfigurasi JobSale Step Transformation



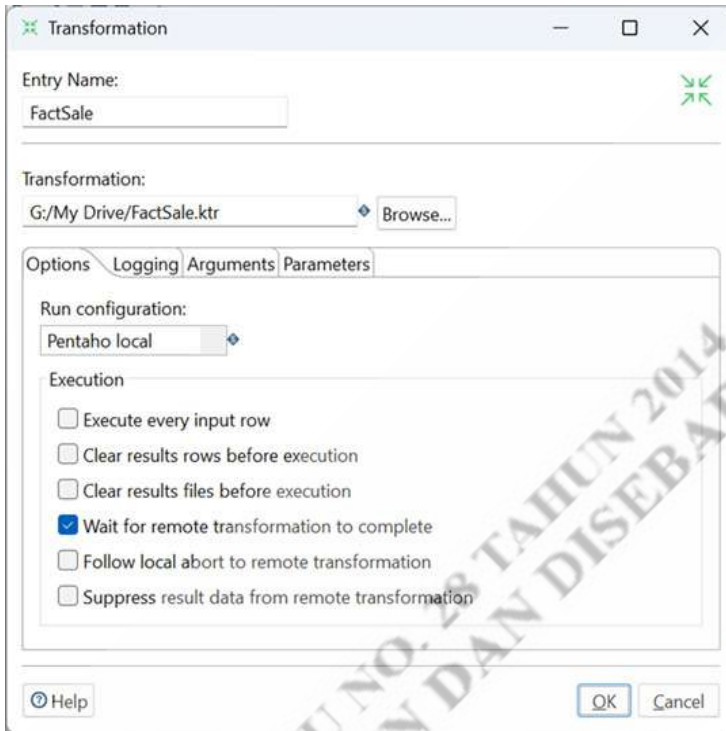
Gambar 133. Konfigurasi JobSale Step Transformation 2



Gambar 134. Konfigurasi JobSale Step Transformation 3

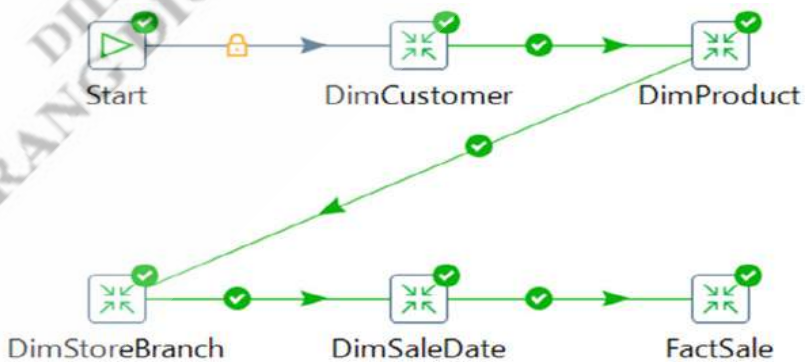


Gambar 135. Konfigurasi JobSale Step Transformation 4



Gambar 136. Konfigurasi JobSale Step Transformation 5

Setelah menyelesaikan konfigurasi semua step, selanjutnya simpan file dan jalankan job yang sudah dibuat.



Gambar 137. Job JobSale Berhasil Dijalankan

Job berhasil dijalankan dan proses ETL telah selesai dilakukan dengan baik.

F. Rangkuman

1. Pentaho Data Integration (PDI), sebelumnya dikenal sebagai "Kettle," adalah alat ETL (Extract, Transform, and Load) yang membantu dalam pengambilan, pembersihan, dan penyimpanan data dalam format yang konsisten. PDI juga mendukung proses integrasi data untuk teknologi Internet of Things (IoT).
2. Klien PDI (Spoon) adalah aplikasi desktop yang digunakan untuk membuat transformasi data, menjadwalkan, dan menjalankan tugas ETL. Fungsinya meliputi migrasi data, pembersihan data, integrasi data secara real-time, dan pengisian data warehouse.
3. Kettle terdiri dari beberapa komponen utama, termasuk Data Integration Server untuk eksekusi dan manajemen job dan transformation, Spoon sebagai antarmuka grafis untuk merancang job dan transformation, serta alat baris perintah seperti Pan dan Kitchen untuk eksekusi transformation dan job.
4. Carte adalah server ringan dalam Kettle yang memungkinkan eksekusi job dan transformation di host remote dengan fungsi eksekusi yang serupa dengan Data Integration Server, namun tanpa fitur penjadwalan, keamanan, dan manajemen konten.
5. Transformation adalah rangkaian tugas logis yang dibangun menggunakan step dan dihubungkan oleh hop. Transformation mengatur aliran data dalam proses ETL, seperti membaca, memfilter, dan memuat data ke dalam database. Step adalah elemen dasar yang menjalankan tugas spesifik, dan step ini dikelompokkan berdasarkan fungsi, misalnya input atau output.
6. Hop adalah jalur yang menghubungkan step dalam transformation, memungkinkan data berpindah dari satu step ke step berikutnya. Selain itu, Kettle memiliki job, yang

merupakan workflow untuk mengoordinasikan berbagai fungsi ETL. Job terdiri dari job entries, job hops, dan job settings, dengan job entries berfungsi seperti step dalam transformation.

7. Berbagai step yang digunakan dalam Kettle meliputi input dan output data dari berbagai format (CSV, Excel, database), serta operasi lainnya seperti penambahan nomor urut, perhitungan matematis, dan seleksi nilai. Setiap step berperan penting dalam mentransformasi dan mengelola data sesuai kebutuhan bisnis.
8. Spoon menyediakan antarmuka grafis yang memudahkan pengguna dalam merancang dan mengelola transformation serta job. Pengguna dapat menambahkan dan menghubungkan step serta job entries di canvas, mengatur aliran data, dan eksekusi dengan mudah.
9. Setiap step atau entry dalam transformation atau job dapat dikonfigurasi dengan detail. Spoon juga menyediakan fitur pratinjau, eksekusi, dan debugging untuk memastikan bahwa data diproses dan dieksekusi dengan benar.
10. Spoon memungkinkan penyimpanan transformation dan job dalam format file khusus serta pengelolaan file-file ini di dalam repository. Alat ini juga mendukung kolaborasi tim dan pengelolaan versi dalam proyek ETL (Extract, Transform, Load).

G. Soal Latihan

1. Apa itu Pentaho Data Integration (PDI) dan untuk apa digunakan?
2. Apa fungsi dari Spoon dalam Pentaho Data Integration?
3. Sebutkan komponen utama dalam Kettle dan apa fungsinya?
4. Apa perbedaan antara Data Integration Server dan Carte?
5. Bagaimana hop dan job bekerja dalam Kettle?
6. Jelaskan cara pembuatan data warehouse menggunakan

Pentaho dengan langkah-langkah di bawah ini:

- a. Membuat file SQL dengan nama order_db.sql yang berisi skrip SQL seperti di bawah ini untuk database operasional pemesanan barang.

```
-- Create database for order_db';
CREATE DATABASE /*!32312 IF NOT EXISTS*/order_db;

USE order_db;

-- Create table for Customer
CREATE TABLE Customer (
    CustomerID CHAR(10) PRIMARY KEY,
    NAME VARCHAR(255) NOT NULL,
    Email VARCHAR(255),
    Phone VARCHAR(20),
    Address TEXT
);

-- Create table for Product
CREATE TABLE Product (
    ProductID CHAR(10) PRIMARY KEY,
    NAME VARCHAR(255) NOT NULL,
    Description TEXT,
    Price DECIMAL(10, 2) NOT NULL,
    StockQuantity INT NOT NULL
);

-- Create table for Order
CREATE TABLE `Order` (
    OrderID CHAR(10) PRIMARY KEY,
    CustomerID CHAR(10),
    OrderDate DATE NOT NULL,
    TotalAmount DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID)
);

-- Create table for OrderItem
CREATE TABLE OrderItem (
    OrderID CHAR(10),
    ProductID CHAR(10),
    Quantity INT NOT NULL,
    Price DECIMAL(10, 2) NOT NULL,
    PRIMARY KEY (OrderID, ProductID),
    FOREIGN KEY (OrderID) REFERENCES `Order`(OrderID),
    FOREIGN KEY (ProductID) REFERENCES Product(ProductID)
);

-- Create table for Payment
CREATE TABLE Payment (
    PaymentID CHAR(10) PRIMARY KEY,
    OrderID CHAR(10),
    PaymentDate DATE NOT NULL,
    Amount DECIMAL(10, 2) NOT NULL,
    PaymentMethod VARCHAR(50),
    FOREIGN KEY (OrderID) REFERENCES `Order`(OrderID)
);
```


- b. Lakukan impor struktur tabel ke dalam database order_db dengan cara impor skrip SQL yang telah dibuat (order_db.sql).
- c. Setelah tabel dalam database order_db terbentuk, lakukan pengisian data berikut ini:

Tabel Customer

CustomerID	Name	Email	Phone	Address
CUST001	Andi Setiawan	andi.setiawan@example.com	0991-555-1234	Jl. Sudirman No. 45, Jakarta
CUST002	Budi Hartono	budi.hartono@example.com	0992-555-5678	Jl. Merdeka No. 10, Bandung
CUST003	Citra Dewi	citra.dewi@example.com	0993-555-8765	Jl. Diponegoro No. 20, Yogyakarta

Tabel Product

ProductID	Name	Description	Price	StockQuantity
PROD001	Laptop Gaming	Laptop dengan performa tinggi untuk gaming	14999.99	10
PROD002	Smartphone	Smartphone terbaru dengan fitur canggih	4999.99	20
PROD003	Tablet	Tablet ringan dengan layar besar	2999.99	15

Tabel Order

OrderID	CustomerID	OrderDate	TotalAmount
ORD001	CUST001	15/01/2024	14999.99
ORD002	CUST002	20/01/2024	4999.99
ORD003	CUST003	10/02/2024	2999.99
ORD004	CUST001	05/03/2024	15999.99
ORD005	CUST002	12/03/2024	5999.99
ORD006	CUST003	25/03/2024	3499.99
ORD007	CUST001	10/04/2024	12999.99
ORD008	CUST002	15/04/2024	5499.99
ORD009	CUST003	05/05/2024	2899.99
ORD010	CUST001	20/05/2024	13999.99
ORD011	CUST002	10/06/2024	4599.99
ORD012	CUST003	15/06/2024	3199.99

Tabel OrderItem

OrderID	ProductID	Quantity	Price
ORD001	PROD001	1	14999.99
ORD001	PROD003	1	2999.99
ORD002	PROD002	1	4999.99
ORD003	PROD003	1	2999.99
ORD004	PROD001	1	15999.99
ORD004	PROD002	1	4999.99
ORD005	PROD002	1	5999.99
ORD006	PROD003	1	3499.99
ORD007	PROD001	1	12999.99
ORD007	PROD003	1	2999.99
ORD008	PROD002	1	5499.99
ORD009	PROD003	1	2899.99
ORD010	PROD001	1	13999.99
ORD010	PROD002	1	4599.99
ORD011	PROD002	1	4599.99
ORD012	PROD003	1	3199.99

Tabel Payment

PaymentID	OrderID	PaymentDate	Amount	PaymentMethod
PAY001	ORD001	16/01/2024	14999.99	Kartu Kredit
PAY002	ORD002	21/01/2024	4999.99	PayPal
PAY003	ORD003	11/02/2024	2999.99	Kartu Debit
PAY004	ORD004	06/03/2024	15999.99	Kartu Kredit
PAY005	ORD005	13/03/2024	5999.99	Kartu Debit
PAY006	ORD006	26/03/2024	3499.99	PayPal
PAY007	ORD007	11/04/2024	12999.99	Kartu Kredit
PAY008	ORD008	16/04/2024	5499.99	PayPal
PAY009	ORD009	06/05/2024	2899.99	Kartu Debit
PAY010	ORD010	21/05/2024	13999.99	Kartu Kredit
PAY011	ORD011	11/06/2024	4599.99	PayPal
PAY012	ORD012	16/06/2024	3199.99	Kartu Debit

- d. Lakukan perancangan data warehouse dan proses ETL menggunakan Pentaho.

BAB VI

Multi Dimensional Modeling

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu menerapkan prinsip *Multi Dimensional Modelling* untuk menyelesaikan masalah

Setelah membaca buku ini diharapkan mahasiswa mampu menerapkan multi dimensional modeling, tahapan model data multi dimensi, fitur model data multi dimensi, keuntungan dan kerugian model data multi dimensi.

Materi yang diberikan :

1. Multi Dimensional Modeling.
2. Tahapan Model Data Multi Dimensi.
3. Fitur Model Data Multi Dimensi.
4. Keuntungan dan Kerugian Model Data Multi Dimensi.

Capaian Pembelajaran:

Capaian pembelajaran mata kuliah pada bab ini adalah mahasiswa mampu menerapkan prinsip Multi Dimensional Modeling untuk menyelesaikan masalah. Capaian pembelajaran pada bab ini adalah mampu menguasai prinsip dan teknik perancangan sistem terintegrasi dengan pendekatan sistem.

B. Multi Dimensional Modeling

Pemodelan Multi Dimensi adalah suatu metode yang digunakan untuk mengurutkan data dalam database beserta penataan dan penyusunan isi database dengan baik.

Model Data Multi Dimensi memungkinkan pelanggan untuk menginterogasi pertanyaan analitis yang terkait dengan tren pasar atau bisnis, tidak seperti database relasional yang memungkinkan pelanggan mengakses data dalam bentuk kueri.

Mereka memungkinkan pengguna dengan cepat menerima jawaban atas permintaan yang mereka buat dengan membuat dan memeriksa data dengan relatif cepat.

OLAP (*Online Analytical Processing*) dan pergudangan data menggunakan database multi dimensi. Ini digunakan untuk menampilkan beberapa dimensi data kepada pengguna. Ini mewakili data dalam bentuk kubus data. Kubus data memungkinkan untuk memodelkan dan melihat data dari berbagai dimensi dan perspektif. Hal ini ditentukan oleh dimensi dan fakta dan diwakili oleh tabel fakta. Fakta adalah ukuran numerik dan tabel fakta berisi ukuran tabel dimensi terkait atau nama fakta

Berdasarkan langkah-langkah yang telah ditentukan sebelumnya, Model Data Multidimensi berfungsi.

1. Tahapan Model Data Multi Dimensi

Tahapan berikut harus diikuti oleh setiap proyek untuk membangun Model Data Multi Dimensi :

a. Tahap 1 : Mengumpulkan data dari klien.

Pada tahap pertama, Model Data Multi Dimensi mengumpulkan data yang benar dari klien. Kebanyakan, profesional perangkat lunak memberikan kemudahan kepada klien tentang jangkauan data yang dapat diperoleh dengan teknologi yang dipilih dan mengumpulkan data lengkap secara rinci.

b. Tahap 2 : Mengelompokkan berbagai segmen sistem.

Pada tahap kedua, Model Data Multi Dimensi mengenali dan mengklasifikasikan semua data ke bagian masing-masing dan juga membangunnya tanpa masalah untuk diterapkan langkah demi langkah.

c. Tahap 3 : Memperhatikan perbedaan proporsi

Pada tahap ketiga, merupakan dasar yang mendasari perancangan sistem. Pada tahap ini, faktor-faktor utama dikenali berdasarkan sudut pandang pengguna. Faktor-faktor ini juga dikenal sebagai “Dimensi”.

d. Tahap 4 : Mempersiapkan faktor-faktor aktual dan kualitasnya masing-masing.

Pada tahap keempat, faktor-faktor yang dikenali pada langkah sebelumnya digunakan lebih lanjut untuk mengidentifikasi kualitas-kualitas terkait. Kualitas ini juga dikenal sebagai “atribut” dalam database.

e. Tahap 5 : Menemukan aktualitas faktor-faktor yang disebutkan sebelumnya dan kualitasnya.

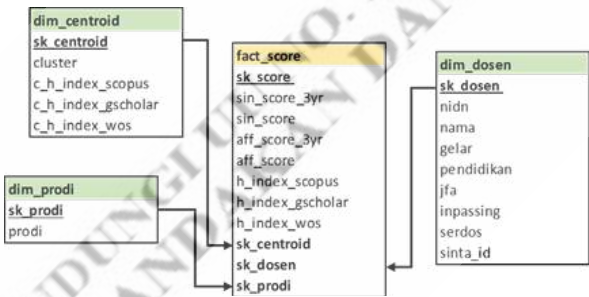
Pada tahap kelima, Model Data Multi Dimensi memisahkan dan membedakan aktualitas dari faktor-faktor yang dikumpulkan olehnya. Ini sebenarnya memainkan peran penting dalam penyusunan Model Data Multi Dimensi.

f. Tahap 6 : Membangun Skema untuk menempatkan data, sehubungan dengan informasi yang dikumpulkan dari langkah-langkah di atas.

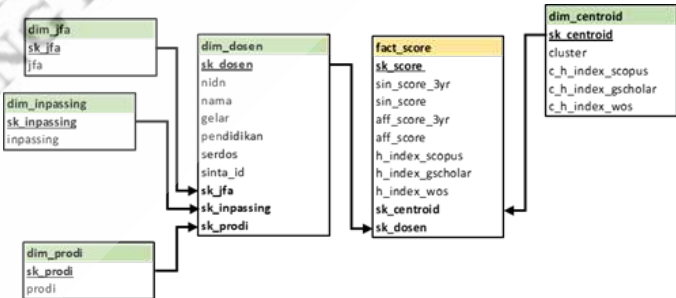
Pada tahap keenam, berdasarkan data yang dikumpulkan sebelumnya, sebuah Skema dibangun.

Model Data multidimensi sifatnya sedikit rumit dan memerlukan tenaga profesional untuk mengenali dan memeriksa data dalam database. Selama kerja Model Data Multi-Dimensi, ketika sistem melakukan cache, terdapat pengaruh yang besar pada kerja sistem. Sifatnya rumit karena database umumnya memiliki desain yang dinamis. Jalan untuk mencapai produk akhir seringkali rumit. Karena Model Data Multi Dimensi memiliki sistem yang rumit, database memiliki jumlah database yang banyak sehingga sistem menjadi sangat tidak aman ketika ada celah keamanan.

Beberapa penelitian sebelumnya model penelitian skema bintang terhadap kinerja dosen dalam kegiatan penelitian (Amin, et.al,2021).Berikut adalah contoh gambar star schema dan snowflakes schema (Hidayat.MK et.al,2025)



(a) Star schema



(b) Snowflakes schema

Gambar 138. Model dimensional score peneliti

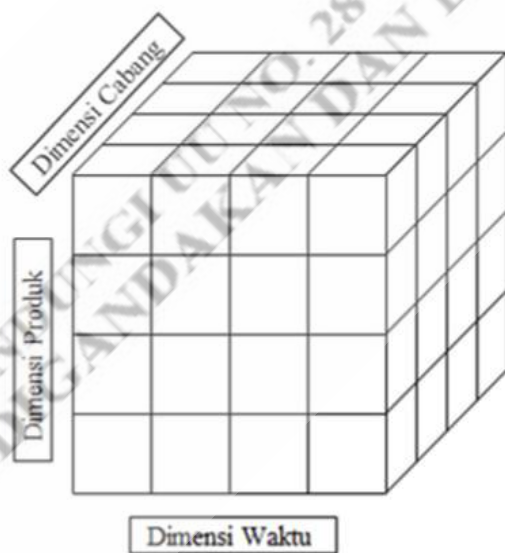
Dalam proses *Extract Transform Load* selanjutnya untuk merancang *Data Warehouse* digunakan pemodelan dimensi yang dilakukan pada berbagai model, salah satunya adalah skema bintang. Sesuai dengan namanya “bintang”, bentuknya sebenarnya seperti *Star Schema* terdiri dari dimensi-dimensi yang terdapat dalam tabel dimensi dan tabel fakta pusat yang memuat fakta. Seperti *Star schema*, nama *Snowflakes Schema* juga muncul karena kemiripannya dengan kepingan salju. Seperti *Star Schema*, skema ini juga digunakan dalam proses ETL dan pada level yang sama. Ada pilihan antara menggunakan salah satu dari keduanya. Berbeda dengan skema bintang, *Snowflakes Schema* terdiri dari tiga jenis tabel, yaitu tabel fakta, tabel dimensi, dan tabel subdimensi. (Karmani, 2020)

2. Fitur model data multidimensi:

- a. **Pengukuran:** Pengukuran adalah data numerik yang dapat dianalisis dan dibandingkan, seperti penjualan atau pendapatan. Pengukuran biasanya disimpan dalam tabel fakta dalam model data multidimensi.
- b. **Dimensi:** Dimensi adalah atribut yang menggambarkan ukuran, seperti waktu, lokasi, atau produk. Dimensi biasanya disimpan dalam tabel dimensi dalam model data multidimensi.
- c. **Kubus:** Kubus adalah struktur yang menggambarkan hubungan multidimensi antara ukuran dan dimensi dalam model data. Kubus menyediakan cara yang cepat dan efisien untuk mengambil dan menganalisis data.
- d. **Agregasi:** Agregasi adalah proses meringkas data di berbagai dimensi dan tingkat detail. Ini adalah fitur utama model data multidimensi, karena memungkinkan pengguna menganalisis data dengan cepat pada berbagai tingkat ketelitian.
- e. **Drill-down dan roll-up:** Drill-down adalah proses berpindah dari ringkasan data tingkat tinggi ke tingkat detail yang lebih rendah, sedangkan roll-up adalah proses sebaliknya, yaitu berpindah dari detail tingkat rendah ke

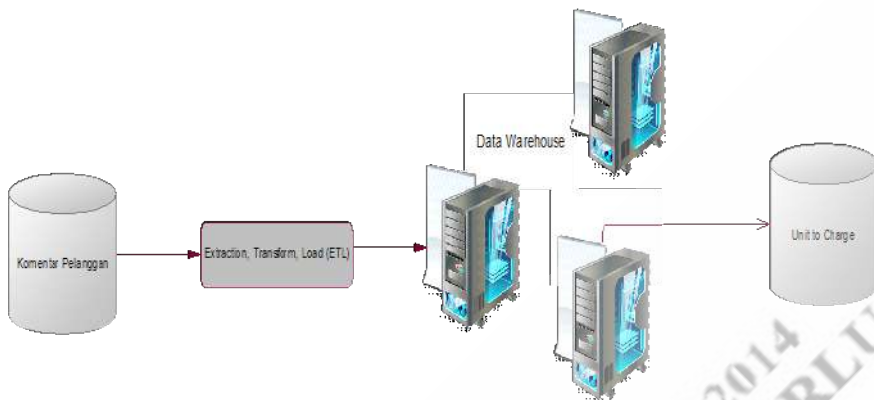
ringkasan tingkat tinggi. Fitur-fitur ini memungkinkan pengguna untuk menjelajahi data secara lebih rinci dan memperoleh wawasan tentang pola yang mendasarinya.

- f. **Hirarki:** Hirarki adalah cara mengatur dimensi ke dalam tingkat detail. Misalnya, dimensi waktu dapat diatur ke dalam tahun, kuartal, bulan, dan hari. Hirarki menyediakan cara untuk menelusuri data dan melakukan operasi drill-down dan roll-up.
- g. **OLAP (Online Analytical Processing):** OLAP adalah jenis model data multidimensi yang mendukung pemrosesan kueri yang cepat dan efisien terhadap kumpulan data besar. Sistem OLAP dirancang untuk menangani kueri yang kompleks dan memberikan waktu respons yang cepat.



Gambar 139. OLAP (Fitriana,2018)

Contoh Model arsitektur sistem intelijensia bisnis dapat dilihat pada Gambar 139 yaitu pemodelan data komentar pelanggan yang kemudian diekstrak, transformasi dan loading dibuat menjadi meta data repository yang merupakan pembetulan data warehouse.



Gambar 140. Contoh Arsitektur Sistem Intelijensia Bisnis (Fitriana et.al.,2019)

3. Keuntungan dan Kekurangan Model Data Multi Dimensi

Keuntungan Model Data Multi Dimensi

Berikut ini adalah keuntungan dari model data multidimensi:

- Model data multidimensi mudah ditangani.
- Mudah dirawat.
- Kinerjanya lebih baik daripada basis data normal (misalnya basis data relasional).
- Representasi data lebih baik daripada basis data tradisional. Hal ini dikarenakan basis data multidimensi memiliki banyak tampilan dan memuat berbagai jenis faktor.
- Ini dapat diterapkan pada sistem dan aplikasi yang kompleks, berbeda dengan sistem basis data satu dimensi yang sederhana.
- Kompatibilitas dalam jenis pangkalan data ini merupakan peningkatan bagi proyek yang memiliki bandwidth lebih rendah untuk staf pemeliharaan.

Kekurangan Model Data Multi Dimensi

Berikut ini adalah kelemahan dari Model Data Multi Dimensi:

1. Model Data multidimensi sedikit rumit sifatnya dan memerlukan profesional untuk mengenali dan memeriksa data dalam basis data.
2. Selama pengerjaan Model Data Multi-Dimensi, ketika sistem melakukan cache, ada pengaruh yang besar terhadap kinerja sistem.
3. Sifatnya yang rumit membuat desain basis data umumnya dinamis.
4. Jalan untuk mencapai hasil akhir seringkali rumit.
5. Karena Model Data Multi Dimensi memiliki sistem yang rumit, basis data memiliki sejumlah besar basis data yang menyebabkan sistem menjadi sangat tidak aman jika terjadi pelanggaran keamanan.

Model RFM (Recency, Frekuensi, Moneter)

Model yang paling memiliki kekuatan dan mudah untuk mengimplementasikan CRM adalah model RFM – Nilai Recency, Frekuensi, Nilai moneter.

Definisi dari RFM adalah:

- (1) R (Recency) nilai minimum dari periode terakhir dilakukan pembelian sampai pembelian berikutnya;
- (2) F (Frekuensi) jumlah pembelian yang dilakukan dalam periode tertentu, frekuensi tertinggi mengindikasikan loyalitas tertinggi;
- (3) M (Monetary) jumlah uang yang dikeluarkan dalam periode tertentu, nilai tertinggi mengindikasikan perusahaan sebaiknya lebih fokus terhadap konsumen tersebut (Mahboubeh et al., 2010).

Diagram input output sub model CRM disajikan pada Tabel 6. Penelitian ini dibagi menjadi 3 fase adalah sebagai berikut: (Fitriana et.al.,2012).

Tabel 6. Diagram Input Output Pemodelan CRM

No	INPUT	AKTIVITAS	OUTPUT
1	Data tanggal penjualan harian ke industri	Retailer Menghitung Recency terkecil	Data Recency penjualan terkecil
2	Retailer Data penjualan harian ke industri	Menghitung Frequency terbesar	Data Frequency harian
3	Retailer Data rupiah penjualan ke industri	Menghitung Monetary terbesar	Data Monetary Penjualan terbesar

Fase 1: Pengertian Bisnis Pada langkah pertama, data dikumpulkan dari koperasi susu. Data adalah basis data transaksi Industri Pengolahan Susu dan Retailer dari bulan September Oktober. Konsumen dari koperasi susu ini dibagi dua yaitu industri dan Retailer, tapi pada makalah ini dibatasi hanya data industri. Tujuan dari sub model CRM ini adalah membuat segmentasi konsumen berdasarkan RFM. Setelah membuat segmentasi konsumen rata-rata, rangking CLV berdasarkan tiap segmen.

Fase 2 : Pengertian Data Fase ini melihat data yang tersedia untuk ditambang meliputi pengumpulan data, deskripsi data dan menverifikasi kualitas data. Database Koperasi meliputi varietas data yang sangat besar.

Fase 3: Persiapan Data Persiapan Data adalah hal yang paling penting. Pada fase pengertian bisnis, data dari koperasi susu dieliminasi menjadi dataset.

Berikut adalah formulasi matematika untuk sub model CRM dengan RFM (Mahboubeh, 2010):

$$\text{Nilai CLV}_k = (NR_k \times WR_k) + (NF_k \times WF_k) + (NM_k \times WM_k)$$

Dimana:

CLV : Customer Lifetime Value=Nilai Siklus Hidup Konsumen

NR: Normalisasi Recency (tanggal penjualan terakhir – tanggal penjualan sebelumnya)

WR_k : Bobot recency

: Normalisasi Frekuensi transaksi

1. Fact Dimension Table

= Bobot Monetary

K = perusahaan

Tabel 7. Transformasi data

Data Awal	Transformasi Data
Tanggal penjualan terakhir (type: date)	<i>Recency</i> (type: int)
<i>Count</i> penjualan	Frequency
Total uang	<i>Monetary</i> : Total uang /count penjualan

Contoh Penerapan *K-Means* dan *K-Medoids* dilakukan melalui tahapan proses *data mining*, (Hidayat M.K dan Fitriana R, 2023) yaitu:

1. Pengumpulan Data

Perhatian utama untuk akurasi informasi adalah sumber data (Jassim & Abdulwahid, 2021). Data yang digunakan pada penelitian adalah data produksi daging ayam yang diperoleh dari Dinas Ketahanan Pangan dan Peternakan Provinsi Jawa Barat yang dipublikasikan pada situs web <https://opendata.jabarprov.go.id>.

2. Data Preprocessing

Teknik *data preprocessing* adalah langkah penting untuk *data mining*, hasil analisis akan baik selama kualitas data baik (Joshi

& Patel, 2020). Tahapan *data preprocessing* dilakukan untuk menghindari masalah yang timbul karena format data yang tidak konsisten sehingga dapat mengganggu ketika pemrosesan data berlangsung dan menyebabkan berkurangnya tingkat akurasi hasil dari proses data. Teknik *data preprocessing* yaitu *data cleaning* untuk membersihkan *missing value*, *data transformation* untuk mengubah bentuk data yang sesuai pada proses data mining, *data integration* untuk menggabungkan beberapa dataset dalam suatu dataset.

3. Implementasi Data Mining

Proses *data mining* dilakukan pada tahap ini dengan menerapkan algoritma *K-Means* dan *K-Medoids* menggunakan data hasil *preprocessing*. Memilih algoritma yang sesuai dalam proses *data mining* sangat penting karena menentukan hasil yang akan mempengaruhi keputusan bisnis (Chikohora, 2014). Algoritma yang dipilih ditentukan berdasarkan pada nilai *Davies Bouldin Index* (DBI) yang dihasilkan.

4. Evaluasi Jumlah Cluster

Performa algoritma diukur dengan menghitung DBI untuk mengevaluasi *cluster* secara umum berdasarkan jumlah dan kedekatan antar elemen klaster. DBI adalah salah satu teknik pengukuran validitas *clustering* (Aggarwal & Reddy, 2014).

DBI mengukur kekompakan cluster yang dihasilkan dan seberapa baik *cluster* ini dipisahkan (Ghany et al., 2022).

5. Penentuan Algoritma Terbaik

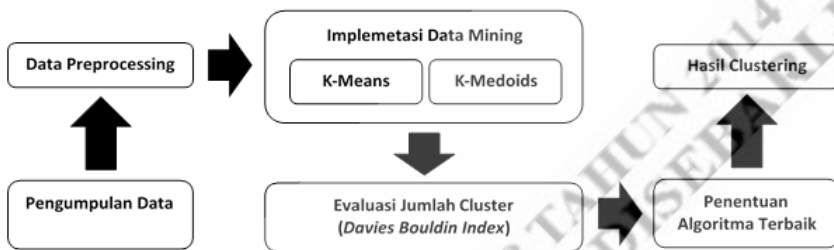
Penentuan performa algoritma terbaik berdasarkan pada nilai DBI, performa algoritma dikatakan terbaik jika memiliki DBI lebih kecil. Semakin kecil nilai DBI memberikan hasil pengelompokan yang lebih baik (Aggarwal & Reddy, 2014; Sivaguru & Punniyamoorthy, 2021). Penelitian ini membandingkan kinerja algoritma *clustering K-Means* dan *K-Medoids* sebanyak 6 *cluster*. Nilai *k* dimasukan secara bertahap mulai dari 2, 3, 4, 5 dan 6 untuk mendapatkan nilai DBI dari

setiap jumlah *cluster* dan hasilnya dibandingkan.

6. Hasil *Clustering*

Pada tahap akhir diperoleh hasil dari proses *data mining* dari penerapan algoritma terbaik yaitu pengelompokan wilayah potensial produksi daging ayam.

Alur proses data mining dapat dilihat pada Gambar 141.



Gambar 141. Proses *Data Mining*

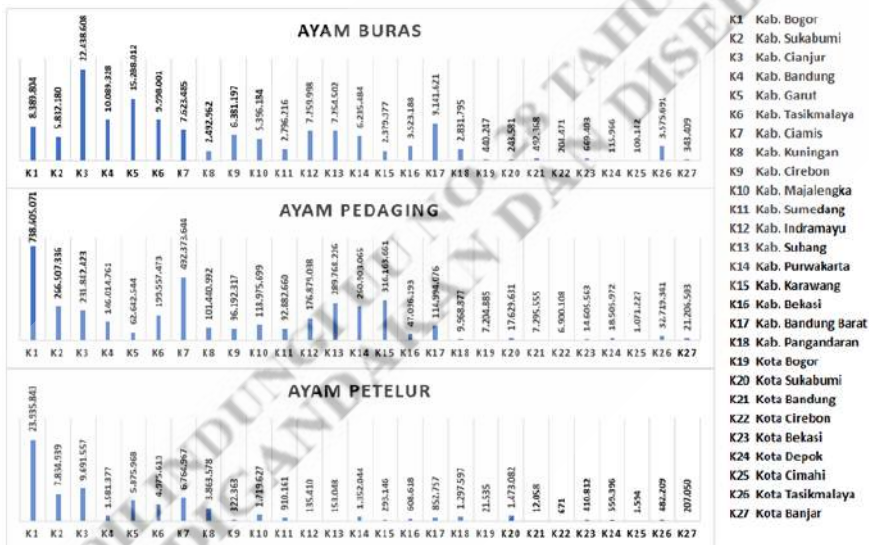
Data untuk keperluan penelitian diperoleh dari Dinas Ketahanan Pangan dan Peternakan Provinsi Jawa Barat yang dipublikasikan melalui situs web <https://opendata.jabarprov.go.id> seperti pada Gambar 142 dan Gambar 143.

Dataset yang diperoleh tentang jumlah produksi daging untuk masing-masing jenis ayam terdiri dari delapan atribut yang menjelaskan jumlah produksi daging ayam per tahun di tiap wilayah kabupaten/kota.

Data preprocessing dilakukan terhadap dataset, kumpulan data diperiksa untuk memastikan data sudah bersih serta melakukan transformasi dan integrasi agar data yang akan digunakan dapat diolah dengan baik pada tahap proses data mining. Langkah selanjutnya setelah data preprocessing yaitu melakukan import data dari format Microsoft Excel ke RapidMiner Studio. Hasil import data pada RapidMiner dapat dilihat pada Gambar 144 dan data jumlah produksi divisualisasikan pada Gambar 145.

Row No.	nama_kab_kota	buras	pedaging	petelur	Row No.	nama_kab_kota	buras	pedaging	petelur
1	Kab. Bogor	8389804	738605071	23935843	15	Kab. Karawang	2379377	316163661	293146
2	Kab. Sukabumi	5832180	266507336	7834939	16	Kab. Bekasi	3523188	47036193	608618
3	Kab. Cianjur	22438608	231842423	9691557	17	Kab. Bandung Barat	9141621	114994076	852757
4	Kab. Bandung	10089328	146014761	1581377	18	Kab. Pangandaran	2831795	9968877	1297597
5	Kab. Garut	15288012	62642544	5875968	19	Kota Bogor	440247	7204885	21535
6	Kab. Tasikmalaya	9998001	199557473	4975613	20	Kota Sukabumi	243561	17629631	1473082
7	Kab. Ciamis	7623485	492373644	6764967	21	Kab. Bandung	492368	7295555	12058
8	Kab. Kuningan	2492962	101440992	3863578	22	Kota Cirebon	204471	6900108	671
9	Kab. Cirebon	6381197	96192317	322363	23	Kota Bekasi	669403	14605563	410812
10	Kab. Majalengka	5396184	118975699	1719627	24	Kota Depok	115966	18505972	559396
11	Kab. Sumedang	2796216	92882660	910161	25	Kota Cimahi	100142	1071227	1594
12	Kab. Indramayu	7259998	176879038	135410	26	Kota Tasikmalaya	3575691	32719341	482209
13	Kab. Subang	7254502	289768226	153048	27	Kota Banjar	343409	21205503	207050
14	Kab. Purwakarta	6235484	260903065	1352044	ExampleSet (27 examples, 0 special attributes, 4 regular attributes)				

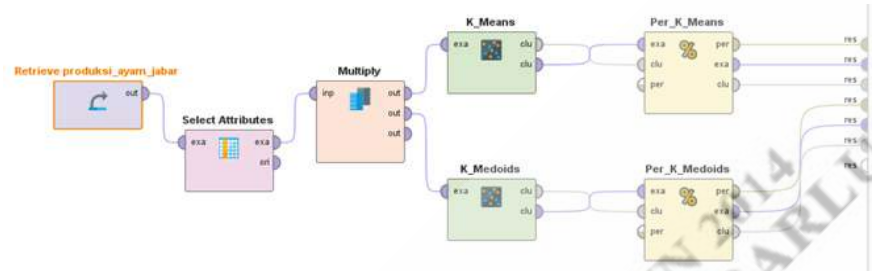
Gambar 142. Import Data Pada RapidMiner



Gambar 143. Visualisasi Data Jumlah Produksi

Pembuatan rancangan proses *data mining* dilakukan sebelum menjalankan proses data lebih lanjut seperti pada Gambar 6. Beberapa operator untuk pengolahan data digunakan pada RapidMiner Studio, yaitu *Select Attributes*, *Multiply*, *K-Means*, *K-Medoids* dan *Cluster Distance Performance*. Operator *Select Attributes* digunakan untuk memilih atribut yang akan digunakan pada pengolahan *clustering data* yaitu buras, pedaging dan petelur. *Multiply* berfungsi untuk membuat salinan obyek pada RapidMiner.

Operator *k-Means* dan *K-Medoids* adalah operator yang akan menjalankan proses *clustering* sesuai algoritma masing-masing. *Cluster Distance Performance* diperlukan untuk evaluasi *centroid* pada metode *clustering*.



Gambar 144. Rancangan RapidMiner Studio

Nilai DBI dari algoritma *clustering K-Means* dan *K-Medoids* sebanyak 6 *cluster* dengan menggunakan RapidMiner dibandingkan untuk mengukur kinerja algoritma terbaik. Perbandingan hasil DBI dapat dilihat pada Tabel 2.

Tabel 8. Davies Bouldin Index (DBI)

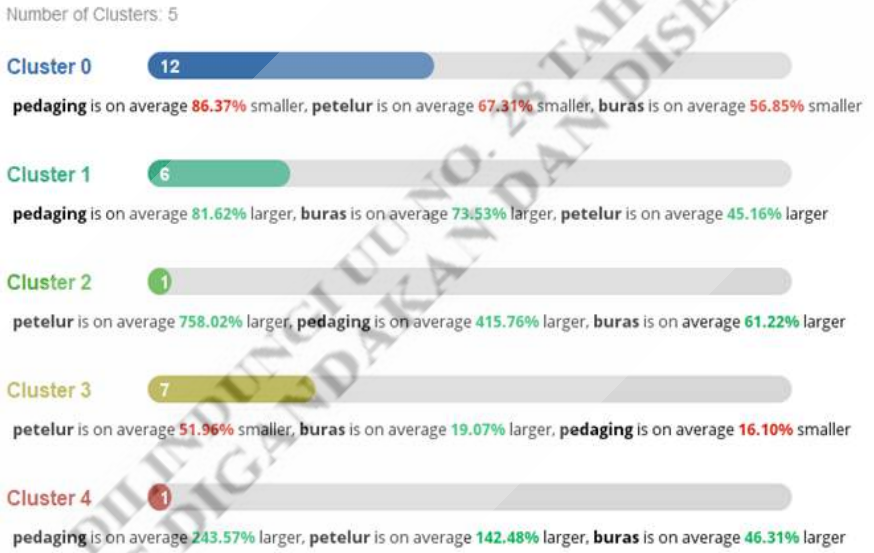
<i>K</i>	<i>K-Means</i>	<i>K-Medoids</i>
2	0,632	0,540
3	0,454	0,338
4	0,411	1,519
5	0,273	0,598
6	0,311	0,918

Dari perbandingan nilai DBI dapat dilihat bahwa nilai DBI terkecil terdapat pada *cluster* 5 menggunakan algoritma *K-Means* yaitu 0,273, hal ini menunjukkan kinerja algoritma *clustering* terbaik yang dipilih. Jumlah elemen setiap kelompok hasil proses *clustering* dengan 5 *cluster* menggunakan algoritma *K-Means* dapat dilihat pada Tabel 3.

Tabel 9. Jumlah Elemen Cluster

Cluster	Jumlah Elemen
0	12
1	6
2	1
3	7
4	1

Visualisasi data jumlah elemen dan ikhtisar setiap kelompok hasil proses clustering dapat dilihat pada Gambar 145.



Gambar 145. Grafik Jumlah Elemen Cluster

Tujuan dari *K-Means Clustering* adalah untuk menemukan sebuah *centroid* untuk setiap *cluster*, semua objek data kemudian diarahkan ke *centroid* terdekat, yang kemudian membentuk sebuah *cluster*. *Centroid* adalah rata-rata dari semua objek data dalam sebuah *cluster* atau objek data yang paling representatif (Umargono et al., 2020). Nilai *centroid* dari setiap *cluster* yang terbentuk hasil proses data dapat dilihat pada Tabel 10.

Tabel 10. Nilai Centroid

Cluster	Buras	Pedaging	Petelur
0	2.319.022,75	20.565.533,25	912.549,17
1	9.023.025,33	260.790.364,00	4.050.057,83
2	8.389.804,00	738.605.071,00	23.935.843,00
3	6.222.500,86	121.054.220,40	1.340.753,29
4	7.623.485,00	492.373.644,00	6.764.967,00

Berdasarkan hasil proses *data mining* menunjukkan bahwa elemen pada kelompok yang terbentuk sudah terpisah pada masing-masing kelompok, dengan demikian tidak terdapat elemen kelompok yang masuk dalam dua kelompok. Suatu *cluster* dapat dinyatakan konvergen jika tidak terjadi perpindahan atau perubahan elemen dari satu *cluster* ke *cluster* yang lain. Hal ini membuktikan bahwa elemen hasil *clustering* bisa mewakili setiap *cluster*. Nilai pusat *cluster* menunjukkan hasil analisis pengelompokan menggunakan algoritma *K-Means* yaitu:

- Cluster 0 berisi produksi ayam buras yang sangat rendah, produksi ayam pedaging yang sangat rendah dan produksi ayam petelur yang sangat rendah.
- Cluster 1 berisi produksi ayam buras yang sangat tinggi, produksi ayam pedaging yang sedang dan produksi ayam petelur yang sedang.
- Cluster 2 berisi produksi ayam buras yang tinggi, produksi ayam pedaging yang sangat tinggi dan produksi ayam petelur yang sangat tinggi.
- Cluster 3 berisi produksi ayam buras yang rendah, produksi ayam pedaging yang rendah dan produksi ayam petelur yang rendah.
- Cluster 4 berisi produksi ayam buras yang sedang, produksi ayam pedaging yang tinggi dan produksi ayam petelur yang tinggi.

Hasil pengelompokan wilayah produksi daging ayam bisa dilihat pada Tabel 5.

Tabel 11. *Cluster* Wilayah Produksi Daging Ayam

Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Kab. Garut	Kab. Sukabumi	Kab. Bogor	Kab. Bandung	Kab. Ciamis
Kab. Bekasi	Kab. Cianjur		Kab. Kuningan	
Kab. Pangandaran	Kab. Tasikmalaya		Kab. Cirebon	
Kota Bogor	Kab. Subang		Kab. Majalengka	
Kota Sukabumi	Kab. Purwakarta		Kab. Sumedang	
Kota Bandung	Kab. Karawang		Kab. Indramayu	
Kota Cirebon			Kab. Bandung Barat	
Kota Bekasi				
Kota Depok				
Kota Cimahi				
Kota Tasikmalaya				
Kota Banjar				

Cluster wilayah produksi daging ayam menunjukkan pengelompokan wilayah produksi daging ayam di provinsi Jawa Barat menjadi 5 *cluster*. *Cluster* 0 merupakan *cluster* yang memiliki anggota kelompok paling banyak dibandingkan dengan *cluster*

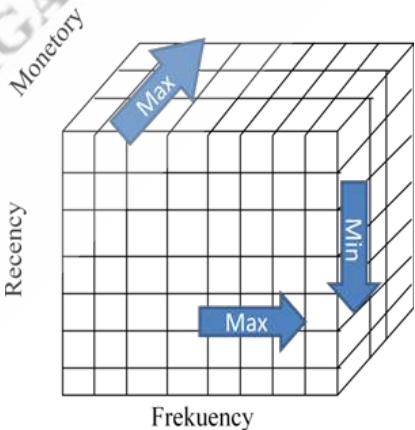
yang lain. Hasil *clustering* dapat dimanfaatkan dalam proses bisnis terkait informasi jumlah produksi daging ayam di wilayah Jawa Barat sebagai acuan dalam pola pembinaan untuk meningkatkan produksi pangan hewani, pengembangan potensi budidaya ayam, dan pengembangan potensi distribusi pakan ternak.

Tabel 38 adalah Tabel Kategori RFM, dibagi menjadi 3 kategori yaitu *low*, *medium*, *high*.

Tabel 12. Kategori RFM

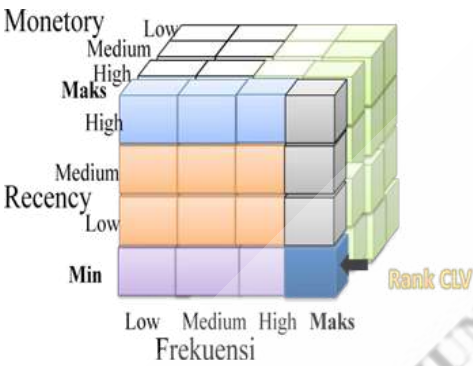
	<i>Low</i>	<i>Medium</i>	<i>High</i>
<i>Recency</i>	1	2-7	7-15
<i>Frequency</i>	10-30	31-50	50-62
<i>Monetary</i>	1.000.000- 50.000.000	51.000.000- 100.000.000	100.000.000- 500.000.000

Kemudian diaggregasi sebagai logical *cube* dengan bentuk sebagai basis dari OLAP. Mininimasi *Recency*, Maksimasi *Frequency*, Maksimasi *Monetary* berkorespondensi untuk melakukan aggregate melalui tiga dimensi yaitu CLV (*Customer Lifetime Value*) terbaik dari bulan September – Oktober 2012. Gambar 146 adalah gambar cube RFM, dimana *recency* minimum, *monetary* maksimum dan frekuensi maksimum.



Gambar 146. *Cube* RFM

Gambar 147 adalah gambar cube RFM tiga dimensi, dimana RFM masing- masing dibagi menjadi *low*, *medium* dan *high*.



Gambar 147. *Cube* RFM tiga dimensi

Model Matematika Pemodelan CRM

Berikut adalah formulasi matematika untuk pemodelan Customer Relationship Management (CRM) dengan RFM dan CLV (Mahboubeh,2010):

$$\text{Nilai } CLV_k = (NR_k \times WR_k) + (NF_k \times WF_k) + (NM_k \times WM_k)$$

CLV_k = *Customer Lifetime Value* = Nilai Siklus Hidup Konsumen ke k

NR_k = Normalisasi *Recency* (tanggal penjualan terakhir – tanggal penjualan sebelumnya)

WR_k = Bobot *recency*

NF_k = Normalisasi Frequency transaksi penjualan per periode

WF_k = Bobot Frequency

NM_k = *Monetary* = Normalisasi uang hasil penjualan per periode/ Frequency

WM_k = Bobot *monetary*

K = perusahaan

Berikut adalah perhitungan nilai CLV:

$WR_k=0,3$, $WF_k=0,35$, $WM_k=0,35$. Nilai bobot didapatkan dari pakar.

NR_k = untuk (*high, medium, low*) nilai NR_k adalah (1, 2, 3) NF_k = untuk (*high, medium, low*) nilai NF_k adalah (3, 2, 1) NM_k = untuk (*high, medium, low*) nilai NF_k adalah (3, 2, 1) NR_k Nilai $CLV_{FFI} = (3 \times 0,3) + (3 \times 0,35) + (3 \times 0,35) = 3$

$$\text{Nilai } CLV_{INDOLAKTO} = (3 \times 0,3) + (2 \times 0,35) + (3 \times 0,35) = 2,65$$

$$\text{Nilai } CLV_{DIAMOND} = (2 \times 0,3) + (2 \times 0,35) + (2 \times 0,35) = 2$$

$$\text{Nilai } CLV_{ALFA} = (2 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,65$$

$$\text{Nilai } CLV_{DI} = (2 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,65$$

$$\text{Nilai } CLV_{DANONE ISAM} = (2 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,65$$

$$\text{Nilai } CLV_{DDI} = (1 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,35$$

$$\text{Nilai } CLV_{UNICAN} = (2 \times 0,3) + (1 \times 0,35) + (1 \times 0,35) = 1,30$$

Tabel 39 adalah ranking CLV berdasarkan nilai CLV berdasarkan data bulan September dan Oktober 2013. Dimana PT. FFI dengan kelas Frequency *High*, kelas monetary *high* dan kelas *recency Low* menduduki peringkat pertama. Diikuti PT. Indolakto dengan kelas Frequency *high*, kelas monetary *medium* dan kelas recent *low* menduduki peringkat kedua. Dan Peringkat ketiga diraih oleh PT. Diamond dengan kelas Frequency *medium*, kelas monetary *medium* dan kelas *recency medium*.

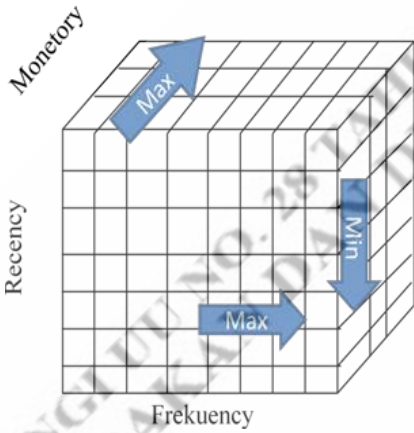
Tabel 38 adalah Tabel Kategori RFM, dibagi menjadi 3 kategori yaitu *low, medium, high*.

Tabel 13. Kategori RFM

	Low	Medium	High
Recency	1	2-7	7-15
Frequency	10-30	31-50	50-62

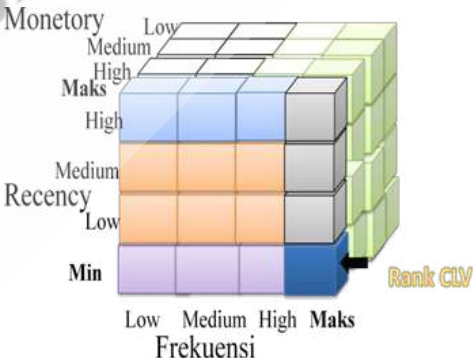
Monetary	1.000.000- 50.000.000	51.000.000- 100.000.000	100.000.000- 500.000.000
----------	--------------------------	----------------------------	-----------------------------

Kemudian diaggregasi sebagai logical *cube* dengan bentuk sebagai basis dari OLAP. Mininimasi *Recency*, Maksimasi *Frequency*, Maksimasi *Monetary* berkorespondensi untuk melakukan aggregate melalui tiga dimensi yaitu CLV (*Customer Lifetime Value*) terbaik dari bulan September – Oktober 2012. Gambar 41 adalah gambar *cube* RFM, dimana *recency* minimum, *monetary* maksimum dan *frekuensi* maksimum.



Gambar 148. *Cube* RFM

Gambar 44 adalah gambar *cube* RFM tiga dimensi, dimana RFM masing- masing dibagi menjadi *low*, *medium* dan *high*.



Gambar 149. *Cube* RFM tiga dimensi

Model Matematika Pemodelan CRM

Berikut adalah formulasi matematika untuk pemodelan Customer Relationship Management (CRM) dengan RFM dan CLV (Mahboubeh,2010):

$$\text{Nilai CLV}_k = (NR_k \times WR_k) + (NF_k \times WF_k) + (NM_k \times WM_k)$$

CLV_k = *Customer Lifetime Value* = Nilai Siklus Hidup
Konsumen ke k

NR_k = Normalisasi *Recency* (tanggal penjualan terakhir – tanggal penjualan sebelumnya)

WR_k = Bobot *recency*

NF_k = Normalisasi Frequency transaksi penjualan per periode

WF_k = Bobot Frequency

NM_k = *Monetary* = Normalisasi uang hasil penjualan per periode/
Frequency

WM_k = Bobot *monetary*

K = *perusahaan*

Berikut adalah perhitungan nilai CLV:

$WR_k=0,3$, $WF_k=0,35$, $WM_k=0,35$. Nilai bobot didapatkan dari pakar.

NR_k = untuk (*high, medium, low*) nilai NR_k adalah (1, 2, 3) NF_k = untuk (*high, medium, low*) nilai NF_k adalah (3, 2, 1) NM_k = untuk (*high, medium, low*) nilai NM_k adalah (3, 2, 1) NR_k Nilai $CLV_{FFI} = (3 \times 0,3) + (3 \times 0,35) + (3 \times 0,35) = 3$

$$\text{Nilai CLV}_{INDOLAKTO} = (3 \times 0,3) + (2 \times 0,35) + (3 \times 0,35) = 2,65$$

$$\text{Nilai CLV}_{DIAMOND} = (2 \times 0,3) + (2 \times 0,35) + (2 \times 0,35) = 2$$

$$\text{Nilai CLV}_{ALFA} = (2 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,65$$

$$\text{Nilai CLV}_{DI} = (2 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,65$$

$$\text{Nilai CLV}_{\text{DANONE ISAM}} = (2 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,65$$

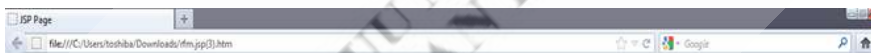
$$\text{Nilai CLV}_{\text{DDI}} = (1 \times 0,3) + (1 \times 0,35) + (2 \times 0,35) = 1,35$$

$$\text{Nilai CLV}_{\text{UNICAN}} = (2 \times 0,3) + (1 \times 0,35) + (1 \times 0,35) = 1,30$$

Tabel 39 adalah ranking CLV berdasarkan nilai CLV berdasarkan data bulan September dan Oktober 2013. Dimana PT. FFI dengan kelas Frequency *High*, kelas monetary *high* dan kelas recency *Low* menduduki peringkat pertama. Diikuti PT. Indolakta dengan kelas Frequency *high*, kelas monetary *medium* dan kelas recent *low* menduduki peringkat kedua. Dan Peringkat ketiga diraih oleh PT. Diamond dengan kelas Frequency *medium*, kelas monetary *medium* dan kelas recency *medium*.

Tabel . Transformasi data Data

Awal Transformasi Data Tanggal penjualan terakhir (type: date) Count penjualan Total uang Recency (type: int) Frequency Monetary: Total uang /count penjualan



Ranking CLV Berdasarkan RFM BULAN SEPTEMBER DAN OKTOBER

Nama IPS	Frekuensi	Monetary	Recency	Kelas Frekuensi	Kelas Monetary	Kelas Recency	Ranking
PT FFI	61	244.290,012	1	High	High	Low	1
PT Indolakta	61	71.017,051	1	High	Medium	Low	2
PT Diamond	55	62.434,618	3	Medium	Medium	Medium	3
PT ALFA	50	7.098,000	2	Medium	Low	Medium	4
PT DI	44	37.896,523	4	Medium	Low	Medium	5
PT Danone Isam	52	42.118,458	2	Medium	Low	Medium	6
PT DDI	30	31.545,225	4	Low	Low	Medium	7
PT Unican	19	69.195,387	8	Low	Medium	High	8

Tabel 7 adalah tabel ranking 3 besar ranking Customer Lifetime Value dari pelanggan perusahaan.

Tabel 14. Ranking CLV

Nama perusahaan	Ranking CLV
PT.F	1
PT.I	2
PT. D	3

Berdasarkan Ranking CLV (Customer Lifetime Value) untuk Industri, dimana didapatkan Ranking tertinggi adalah PT.F, diikuti PT. I dan PT. D sebagai peringkat kedua dan ketiga. Hal ini berarti PT F merupakan konsumen industri prioritas pertama yang merupakan konsumen potensial.

C. Rangkuman

Pemodelan multidimensional mengatur data menjadi fakta (ukuran numerik seperti penjualan) dan dimensi (perspektif kontekstual seperti waktu, lokasi, dan produk) untuk memfasilitasi analisis data yang cepat. Struktur kubus data ini memungkinkan pengguna untuk memotong, memisahkan, dan mengagregasi data dari berbagai sudut, menjawab pertanyaan bisnis yang kompleks seperti Berapa total penjualan per jenis produk dan per bulan?. Ini umum digunakan dalam Gudang Data (*Data Warehouse*) dan Pemrosesan Analitis Online (OLAP) untuk meningkatkan kinerja kueri dan mempermudah penggunaan dalam tugas-tugas analitis.

D. Soal Latihan

1. Sebutkan dan jelaskan tahapan Model Data Multi Dimensi.
2. Sebutkan dan jelaskan fitur model data multidimensi.
3. Sebutkan dan jelaskan keuntungan dan kekurangan model data multi dimensi.
4. Jelaskan konsep RFM (*Recency Frequency Monetary*) dan berikan contoh studi kasusnya.

BAB VII

Data Analitik

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu mengidentifikasi, menformulasikan dan menganalisis masalah dengan menggunakan metodel analitika data

Setelah membaca buku ini diharapkan mahasiswa mampu menerapkan materi yang diberikan :

1. Analitika Data.
2. Klasifikasi.
3. Clustering.
4. Asosiasi.
5. Support Vector Machine.

Capaian Pembelajaran:

Capaian pembelajaran pada bab ini adalah mahasiswa mampu mengidentifikasi, memformulasikan dan menganalisis masalah menggunakan prinsip rekayasa kompleks dengan menggunakan metode analitika data.

Capaian pembelajaran pada bab ini adalah mampu menguasai prinsip dan teknik perancangan sistem terintegrasi dengan pendekatan sistem.

B. Analitika Data

Statistika yang selalu berevolusi. Kemajuan pesat di bidang komputer. Fenomena Big data, permasalahan riset yang baru, menjadi lebih besar dan lebih kompleks.

Analitika Data mendukung proses pengambilan keputusan yang lebih cepat dan akurat, dan mendukung fungsi-fungsi berikut:

1. *Customer Acquisition and Retention.*
2. *Focused dan Targeted Promotions.*
3. *Product Recommendation.*
4. *Fraud Detection.*
5. *Cost Reduction in storing, processing and analyzing large volumes of data.*

Analitika Data terdiri atas teknik Statistika standar, seperti visualisasi data, pemilihan atribut, membuang outlier, dst. untuk mengkonstruksi set data awal, yang selanjutnya sebagai input dalam algoritma machine learning dan uji Statistika untuk memvalidasi (Hastie et al., 2008). Analitika data adalah pengumpulan, transformasi, dan pengorganisasian data untuk menarik kesimpulan, membuat prediksi, dan mendorong pengambilan keputusan yang tepat.

C. Klasifikasi

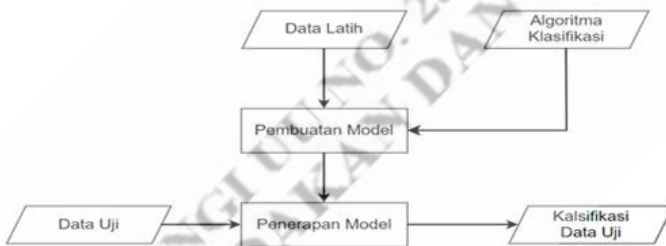
Klasifikasi merupakan salah satu predictive analysis untuk menemukan sebuah model yang mampu menjelaskan dan Ymemisahkan kelas data. Proses ini diturunkan berdasarkan data training dimana label untuk setiap data telah diketahui (Jiawei H, 2012). Tugas utama algoritma klasifikasi adalah membuat suatu

prototipe yang bisa digunakan untuk melakukan pengenalan dan klasifikasi dari suatu data yang belum diketahui kelasnya.

Beberapa contoh permasalahan yang bisa diselesaikan dengan klasifikasi antara lain:

1. Menentukan kelayakan suatu nasabah dalam mengajukan pinjaman.
2. Melakukan deteksi penyakit dari suatu pasien berdasarkan gejala-gejala yang dimiliki.
3. Membagi kelas dari masyarakat berdasarkan pendapatan masing-masing.

Gambar 150 berikut ini menunjukkan metodologi proses klasifikasi yang umundilakukan pada setiap algoritma klasifikasi.



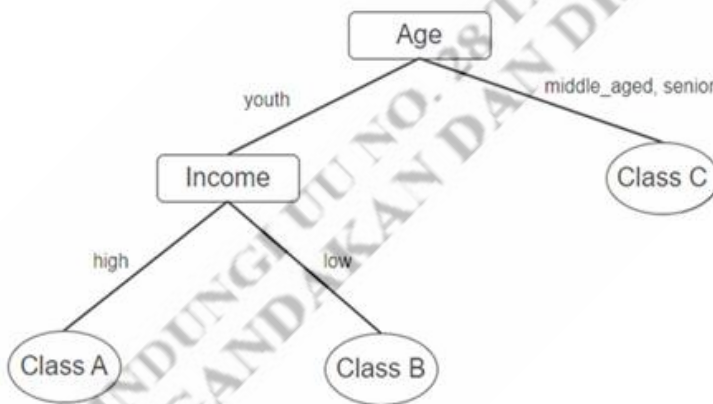
Gambar 150. Metodologi klasifikasi

Model yang dihasilkan akan diterapkan untuk memprediksi target kelas (y) pada data uji (x, y) sehingga dihasilkan data uji (x, y). Umumnya model yang dihasilkan dari teknik klasifikasi dapat memprediksi seluruh data latih dengan benar. Tetapi belum tentu mampu memprediksi dengan benar terhadap seluruh data uji. Maka kita bisa menentukan kelayakan penggunaan model dari nilai accuracy dan error rate (tingkat kesalahan). Suatu model layak digunakan apabila nilai akurasinya lebih tinggi dari nilai tingkat kesalahannya. Jadi semakin nilai akurasinya tinggi, maka model tersebut semakin layak (berfungsi) untuk diimplementasikan. Berikut persamaan/formula untuk menentukan nilai akurasi dan tingkat kesalahan.

Akurasi = $\frac{\text{Jumlah data yang prediksinya benar}}{\text{Jumlah data yang diprediksi}}$

Tingkat Kesalahan = $\frac{\text{Jumlah data yang prediksinya salah}}{\text{Jumlah data yang diprediksi}}$

Salah satu algoritma yang sering digunakan untuk melakukan proses klasifikasi adalah Decision Tree. Flowchart pada algoritma ini berbentuk mirip dengan struktur pohon, dimana setiap node menunjukkan pengujian pada nilai atribut, setiap cabang mewakili pengujian, dan daun pohon mewakili kelas atau distribusi kelas. Gambar 151 berikut ini menggambarkan bagaimana suatu pohon keputusan membagi data yang dimiliki ke dalam beberapa kelas ekonomi berdasarkan usis dan pendapatan.



Gambar 151. Struktur model Decision Tree

Algoritma Decision Tree memiliki beberapa model antara lain ID.3 dan C4.5. Kedua model tersebut memiliki kelebihan dan kekurangan masing-masing. Perbedaan diantara ketiga model tersebut terletak pada bagaimana menentukan Gain pada masing-masing model. Persamaan berikut digunakan untuk menentukan *Gain* dan *Entropy* dari Decision Tree ID.3 (Fitriana,et.al,2022).

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{S_v}{S} Entropy(S_v)$$

$$Entropy(S) = \sum_{i=1}^c -p_i \log_2 p_i$$

$$Entropy(S) = \sum_{i=1}^c -p_i \log_2 p_i$$

$$p_i = \frac{Z_i}{N}$$

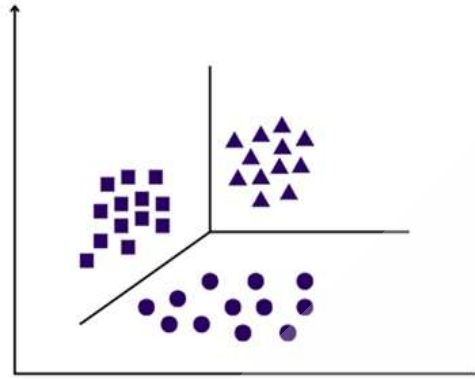
Z_i = contoh positif + contoh negatif

N = jumlah data

Sedangkan pada model C4.5, penentuan nilai Gain tidak berdasarkan nilai Entropy, melainkan nilai Split Info.

$$GainRatio(S, A) = \frac{Gain(S, A)}{SplitInfo(S, A)}$$

$$SplitInfo(S, A) = -\sum_{i=1}^c \frac{S_i}{S} \log_2 \frac{S_i}{S}$$



Gambar 152. Ilustrasi pengelompokan menggunakan metode classification

Contoh kasus

Disini akan diterapkan teknik klasifikasi untuk pembuatan model pohon keputusan dan rule dalam melakukan prediksi ciri-ciri baju yang paling laris dibeli.. Pada kasus ini penentuan keputusan suatu baju akan laku atau tidak, sementara ditentukan oleh 4 atribut, yaitu diantaranya: 1. Atribut lengan (domain: panjang, pendek); 2. Atribut ukuran (domain: besar, medium, kecil); 3. Atribut warna (domain: hitam, merah, biru); 4. Atribut model (domain: slim fit, regular). Pada kasus ini yang menjadi target kelas adalah atribut laku, sehingga terdapat dua target kelas yaitu Tidak/No (S1) dan Ya/Yes (S2). Jumlah objek/kasus yang kita jadikan sebagai data latih adalah sebanyak 14 kasus. Data latih yang akan kita gunakan dapat dilihat pada tabel 1 di bawah ini.

Tabel 15. Data Latih

No	Lengan	Ukuran	Warna	Model	Laku
1	Panjang	Besar	Hitam	Slim Fit	Ya
2	Panjang	Medium	Hitam	Regular	Ya
3	Panjang	Besar	Merah	Regular	Ya
4	Pendek	Kecil	Merah	Regular	Ya
5	Pendek	Besar	Biru	Slim Fit	Ya
6	Pendek	Kecil	Hitam	Slim Fit	Ya
7	Pendek	Medium	Hitam	Regular	Ya

8	Pendek	Kecil	Biru	Regular	Ya
9	Pendek	Besar	Merah	Regular	Ya
10	Pendek	Kecil	Merah	Slim Fit	Ya
11	Panjang	Medium	Biru	Regular	Tidak
12	Panjang	Medium	Biru	Slim Fit	Tidak
13	Panjang	Besar	Biru	Regular	Tidak
14	Panjang	Besar	Merah	Slim Fit	Tidak

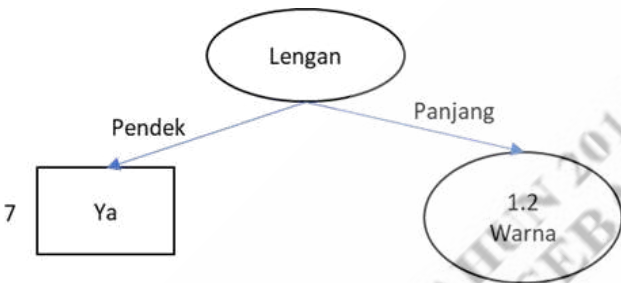
Node 1

Pembentukan model pohon keputusan pada kasus di atas menggunakan algoritma C4.5. Dalam algoritma C4.5, pemilihan atribut sebagai akar didasarkan pada nilai gain tertinggi dari atribut-atribut yang ada. Untuk memudahkan perhitungan gain data latih dipetakan terlebih dahulu seperti pada tabel 2 di bawah ini.

Tabel 16. Perhitungan Entropy dan Gain pada Node 1

Node	Model	Ukuran	Jumlah	Tida	Ya	Entropy	Gain
1	Total		14	4	10	0,86312	
	Lengan						0,37051
		Panjang	7	4	3	0,98523	
		Pendek	7	0	7	0	
	Ukuran						0,18385
		Kecil	4	0	4	0	
		Medium	4	2	2	1	
		besar	6	2	4	0,9183	
	Warna						0,25852
		Hitam	4	0	4	0	
		Merah	5	1	4	0,72193	
		Biru	5	3	2	0,97095	
	Model						0,00598
		Slim Fit	8	2	6	0,81128	
		Regular	6	2	4	0,9183	

Dari tabel di atas, nilai Gain dari lengan merupakan yang terbesar dibanding yang lain, sehingga pada pembentukan pohon keputusan, ukuran lengan menjadi node 1 seperti yang ditunjukkan pada gambar 153 berikut. Sedangkan warna yang memiliki nilai Gain tertinggi kedua menjadi Node 1.2



Gambar 153. Pohon keputusan pada Node I

Node 1.2

Pembentukan Node 1.2 dilakukan menggunakan data-data yang memiliki lengan panjang. Terdapat 7 data yang memiliki nilai lengan panjang. Tabel 17 di bawah ini menunjukkan data-data yang akan digunakan untuk menghitung Entropy dan Gain pada Node 1.2.

Tabel 17. Data Node 1.2

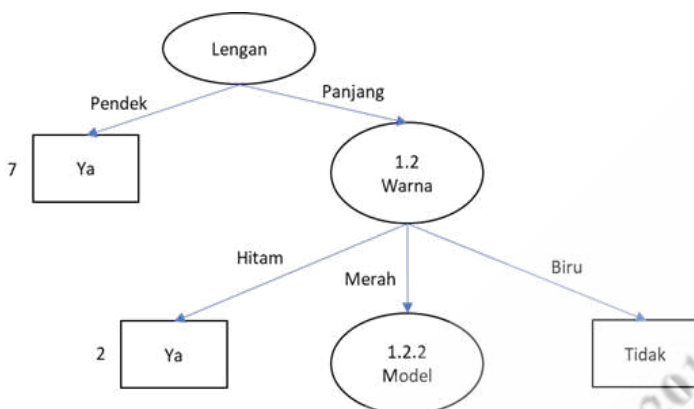
No	Lengan	Ukuran	Warna	Model	Laku
1	Panjang	Besar	Hitam	Slim Fit	Ya
2	Panjang	Medium	Hitam	Regular	Ya
3	Panjang	Besar	Merah	Regular	Ya
4	Panjang	Medium	Biru	Regular	Tidak
5	Panjang	Medium	Biru	Slim Fit	Tidak
6	Panjang	Besar	Biru	Regular	Tidak
7	Panjang	Besar	Merah	Slim Fit	Tidak

Dari hasil perhitungan Entropy dan Gain yang dilakukan pada Node 1.2, didapatkan hasil pada masing-masing model seperti tabel berikut ini.

Tabel 18. Perhitungan Entropy dan Gain pada Node 1.2

Node	Model	Uku- ran	Jum- lah	Tida	Ya	Entro- py	Gain
1.2	Lengan Panjang		7	4	3	0,98523	
	Ukuran						0,38063
		Kecil	0	0	0	0	
		Medi- um	3	2	1	0,9183	
		besar	4	2	2	1	
	Warna						0,72026
		Hitam	2	0	2	0	
		Merah	2	1	1	1	
		Biru	3	3	0	0	
	Model						0,38063
		Slim Fit	3	2	1	0,9183	
		Regu- lar	4	2	2	1	

Dari Tabel 18, nilai Gain dari warna sebesar 0,72026 merupakan yang terbesar dibanding yang lain, sehingga pada pembentukan pohon keputusan, warna menjadi node 1.2 seperti yang ditunjukkan pada Gambar 154. Sedangkan ukuran dan model memiliki nilai Gain yang sama, yaitu 0,38063. Hal itu membuat pemilihan Node berikutnya bisa menggunakan model ataupun ukuran. Pembentukan pohon keputusan pada Node 1.2.2 dapat dilihat pada Gambar 154.



Gambar 154. Pohon keputusan Node 1.2

Nodel 1.2.2

Pembentukan Node 1.2.2 dilakukan menggunakan data-data yang memiliki warna merah. Terdapat 2 data yang memiliki warna merah. Tabel 19 di bawah ini menunjukkan data-data yang akan digunakan untuk menghitung Entropy dan Gain pada Node 1.2.2.

Tabel 19. Data Node 1.2.2

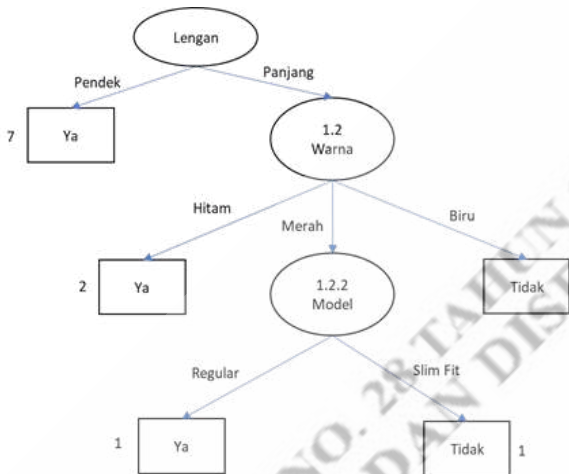
No	Lengan	Ukuran	Warna	Model	Laku
1	Panjang	Besar	Merah	Regular	Ya
2	Panjang	Besar	Merah	Slim Fit	Tidak

Dari hasil perhitungan Entropy dan Gain yang dilakukan pada Node 1.2, didapatkan hasil pada masing-masing model seperti tabel berikut ini.

Tabel 20. Perhitungan Entropy dan Gain pada Node 1.2.2

Node	Model	Ukuran	Jumlah	Tidak	Ya	Entropy	Gain
1.2	Warna Merah		2	1	1	1	
	Model						0,86312
		Slim Fit	1	1	0	0	
		Regular	1	0	1	0	

Dari Tabel 20 didapatkan bentuk pohon keputusan akhir seperti yang terlihat pada Gambar 155.



Gambar 155. Pohon keputusan

Perhitungan sebelumnya merupakan gambaran dasar bagaimana menentukan pohon keputusan pada algoritma Decision Tree menggunakan contoh sederhana dengan data yang tidak terlalu banyak. Dengan kemudahan akses informasi saat ini, mendapatkan suatu data dengan jumlah sample yang sangat besar tidaklah terlalu sulit sehingga perhitungan sebelumnya perlu dilakukan dengan bantuan bahasa pemrograman. Saat ini Python menjadi salah satu tools yang cukup populer digunakan di berbagai industri karena lebih mudah dipelajari dibandingkan dengan bahasa pemrograman yang lain. Di bawah ini merupakan contoh program algoritma Decision Tree menggunakan data bunga Iris yang sudah tersedia pada Python itu sendiri.

```

!pip install scikit-learn matplotlib
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn import tree
from sklearn.metrics import accuracy_score,
classification_report, confusion_matrix

# Load the Iris dataset
iris = load_iris()
X = iris.data
y = iris.target

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X,
y, test_size=0.3, random_state=42)

# Initialize the Decision Tree Classifier
clf = DecisionTreeClassifier(random_state=42)

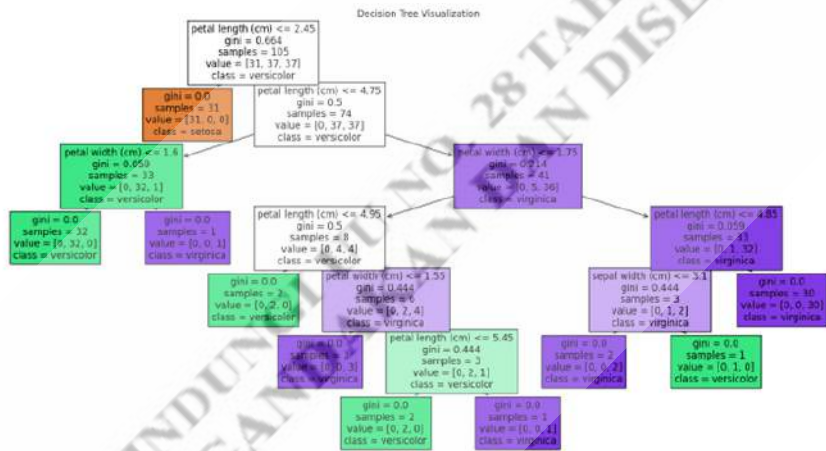
# Train the classifier
clf.fit(X_train, y_train)

# Make predictions
y_pred = clf.predict(X_test)
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy:.2f}")
print("\nClassification Report:")
print(classification_report(y_test, y_pred))
print("\nConfusion Matrix:")
print(confusion_matrix(y_test, y_pred))

```

```
# Visualize the Decision Tree
plt.figure(figsize=(20,10))
tree.plot_tree(clf, feature_names=iris.feature_
names, class_names=iris.target_names, filled=True)
plt.title("Decision Tree Visualization")
plt.show()
```

Program diatas digunakan untuk menentukan spesies dari bunga Iris pada dataset yang digunakan. Dataset tersebut dibagi ke dalam tiga spesies bunga Iris yaitu setosa, versicolor, dan virginica. Gambar 156 menunjukkan pohon keputusan yang berhasil dibuat.

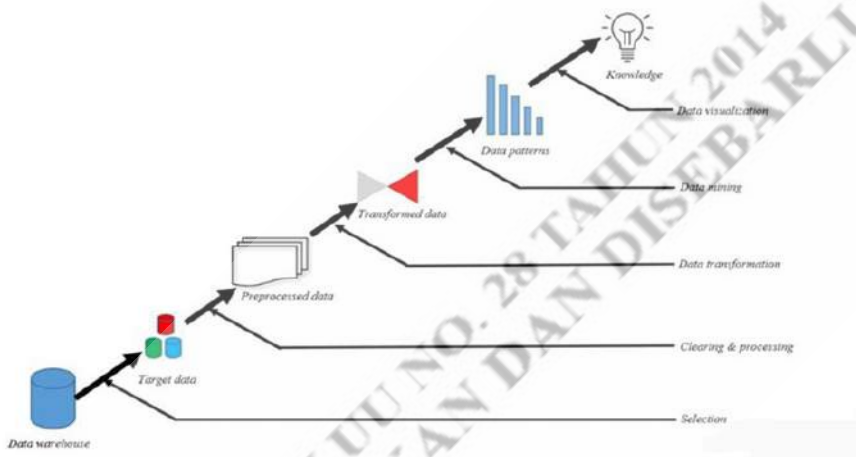


Gambar 156. Pohon keputusan

D. Clustering

Data mining merupakan sebuah teknik untuk melakukan ekstraksi data yang berguna dari *big data* yang bertujuan untuk menemukan suatu informasi atau pengetahuan (Sindhu et al., 2017). Pendapat ini didukung oleh penelitian lain yang menyatakan bahwa *data mining* digunakan untuk memperoleh pola dan trend yang sedang terjadi di dalam data yang dikumpulkan (Syadzali et al., 2020). *Data mining* sendiri merupakan salah satu bagian dari

keseluruhan metode *knowledge discovery in databases* (KDD) yang diawali dari pemilihan data yang sesuai dari *database*, *pre-processing* dan *data cleaning* untuk menghilangkan duplikasi, *error*, dan konflik, tranformasi data ke dalam bentuk yang sesuai untuk *data mining*, proses *data mining*, evaluasi data, dan diakhiri dengan visualisasi hasil yang didapatkan (Krishnamoorthy et al., 2022)(Indraputra et al., 2020). Gambar 157 menunjukkan diagram proses knowledge discovery dalam menemukan suatu informasi dari big data.



Gambar 157. Proses *knowledge discovery*

Terdapat beberapa jenis metode *data mining* yang dapat digunakan untuk mengolah suatu data, antara lain *classification*, *regression*, *clustering*, *summarization*, *dependency modelling*, dan *change and deviation detection*. *Clustering* merupakan salah satu metode *unsupervised learning* yang bertujuan untuk mengidentifikasi suatu set terbatas dari kategori atau *cluster* untuk menggambarkan suatu data (M. Kantardzic., 2020). *Clustering* merupakan suatu *tools* yang sangat berguna dalam *data science* dimana metode ini mengelompokkan suatu kumpulan data yang dikelompokkan berdasarkan kesamaan terbesar dalam satu *cluster* dan perbedaan terbesar diantara *cluster* yang berbeda (K. P. Sinaga et al., 2020). *Clustering* mencoba untuk mengumpulkan data-data yang memiliki kemiripan ke dalam satu kelompok dan tidak dimunculkan lagi di kelompok yang lain. Tingkat kesulitan *clustering* ini sangat tergantung dari bentuk data yang digunakan. Untuk data dua

dimensi, penglihatan manusia masih cukup baik untuk melakukan pengelompokan. Namun saat jumlah dimensi meningkat, maka dibutuhkan algoritma *clustering* untuk melakukannya (S. A. Abbas et al., 2020). *Clustering* melakukan analisa data tanpa adanya peran dari pelabelan kelas dari data tersebut. Hal ini yang membedakan *clustering* dengan klasifikasi ataupun regresi yang membutuhkan pelabelan awal untuk pelatihan. Prinsip utama dari algoritma *clustering* adalah memaksimalkan kemiripan yang dimiliki data-data yang diolah dan meminimalkan persamaan antar *cluster* yang terbentuk.

Jumlah algoritma-algoritma *clustering* dibedakan cukup banyak dan terkadang terdapat *overlap* antara satu algoritma dengan algoritma yang lain sehingga cukup sulit untuk membedakannya. Secara umum terdapat beberapa kategori algoritma *clustering* antara lain:

1. Metode partisi: pengelompokan yang dilakukan berbasis partisi. Metode ini memiliki *center of gravity* (titik pusat) untuk meminimalisir *dissimilarity* (jarak). Contoh algoritma dengan metode partisi antara lain *K-Means* dan *K-Medoids*.
2. Metode hirarki: pengelompokan ini mengelompokkan data ke dalam suatu hirarki. Contoh algoritma dengan metode hirarki antara lain *BIRCH* dan *Chameleon*.
3. Metode *Density*: metode ini memanfaatkan kepadatan suatu data dalam melakukan *clustering*. Contohnya antara lain *DBSCAN*, *OPTICS*, dan *DENCLUE*.
4. Metode *Grid Base*: metode ini menggunakan struktur data *grid* multi-resolusi. Contohnya antara lain *STING* dan *CLIQUE*.

Clustering merupakan suatu teknik yang digunakan dalam penyusunan sekelompok data ke dalam kelas-kelas dimana setiap anggota yang ada di dalam satu kelompok memiliki tingkat similaritas yang tinggi, sedangkan data-data yang berada di kelompok yang berbeda memiliki karakteristik yang berbeda (Tech Difference, 2018). Algoritma untuk melakukan clustering sendiri ada beberapa jenis antara lain *K-Means*, *KNN*, *Naive Bayes Classifier*, dan sebagainya. Terdapat beberapa perbedaan dasar antara metode

klasifikasi yang telah dijelaskan sebelumnya dengan metode clustering. Perbedaan tersebut dapat dilihat pada tabel berikut ini:

Tabel 21. Perbedaan metode clustering dan classification (Pengantar Machine Learning, 2018)

Clustering	Classification
Data tidak berlabel	Data memiliki label
Unsupervised learning	Supervised learning
Pembentukan kelompok didasarkan pada pola kemiripan "pattern similarities" antar anggota	Pembentukan kelas berdasarkan feature yang disediakan dataset
Berfungsi untuk melakukan perkiraan class selanjutnya	Berfungsi untuk eksplorasi data
Jumlah cluster tidak dapat dipastikan	Jumlah kelompok sudah ditentukan

Algoritma K-Means mengelompokkan data berdasarkan kedekatan/kemiripan data sehingga data-data yang memiliki karakteristik yang sama akan dimasukkan kedalam cluster yang sama. Kedekatan data tersebut bisa diukur dari seberapa dekat jarak antar data dan jarak data dengan centroid data. Prinsip kerja algoritma K-Means yaitu menentukan jumlah k cluster terlebih dahulu kemudian menentukan titik centroid data secara random/acak. Setelah itu mengalokasikan data ke cluster terdekat dan proses tersebut akan diulang hingga menemukan centroid yang stabil. Keluaran algoritma K-Means sangat bergantung pada penentuan jumlah cluster dan pemilihan centroid awal yang ditentukan secara random/acak. Permasalahan yang ada pada algoritma K-Means adalah menghasilkan centroid akhir yang tidak benar-benar menjadi pusat cluster yang sesungguhnya. Dalam prakteknya algoritma ini harus dijalankan berkali-kali dengan centroid awal yang berbeda-

beda untuk mendapatkan centroid akhir yang dianggap paling baik. Metode evaluasi cluster dapat menyelesaikan masalah tersebut. Metode evaluasi cluster Seperti metode Elbow, Davies Bouldin Index Dan Silhouette Index merupakan metode internal yang dapat membantu untuk mendapatkan klasterisasi ideal pada algoritma K-Means.(Orisa M.,2022)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.datasets import load_diabetes
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import silhouette_score
# Load the Diabetes dataset
diabetes = load_diabetes()
X = diabetes.data
feature_names = diabetes.feature_names

# Create a DataFrame for easier manipulation
df = pd.DataFrame(X, columns=feature_names)

# Display the first few rows of the dataset
print(df.head())
# Preprocessing
# Standardize the features
scaler = StandardScaler()
scaled_features = scaler.fit_transform(X)

# Determine the optimal number of clusters using the
Elbow method
distortions = []
K = range(1, 11)
for k in K:
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(scaled_features)
    distortions.append(kmeans.inertia_)

# Plot the elbow graph
```

```

plt.figure(figsize=(8, 5))
plt.plot(K, distortions, 'bx-')
plt.xlabel('Number of clusters')
plt.ylabel('Distortion')
plt.title('Elbow Method for Optimal k')
plt.show()

# Choose the optimal number of clusters (e.g., 3
# from the elbow method)
optimal_k = 3
kmeans = KMeans(n_clusters=optimal_k, random_
state=42)
kmeans.fit(scaled_features)
labels = kmeans.labels_

# Calculate silhouette score
silhouette_avg = silhouette_score(scaled_features,
labels)
print(f"Silhouette Score: {silhouette_avg:.2f}")

# Add cluster labels to the original DataFrame
df['Cluster'] = labels
# Plotting the clusters
# We will use PCA to reduce the dimensions to 2D
# for visualization
from sklearn.decomposition import PCA

pca = PCA(n_components=2)
principal_components = pca.fit_transform(scaled_
features)
df_pca = pd.DataFrame(data=principal_components,
columns=['PC1', 'PC2'])
df_pca['Cluster'] = labels

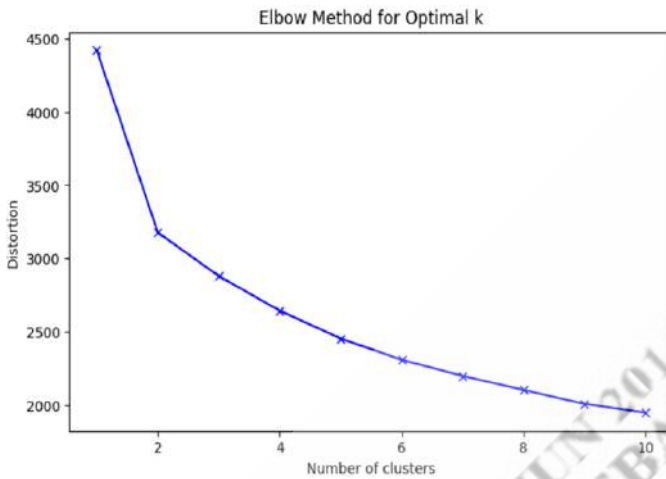
plt.figure(figsize=(12, 8))
sns.scatterplot(x='PC1', y='PC2', hue='Cluster',
data=df_pca, palette='viridis', s=100)
plt.title('K-Means Clustering of Diabetes Dataset
(PCA-reduced)')

```

```
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.legend()
plt.show()
```

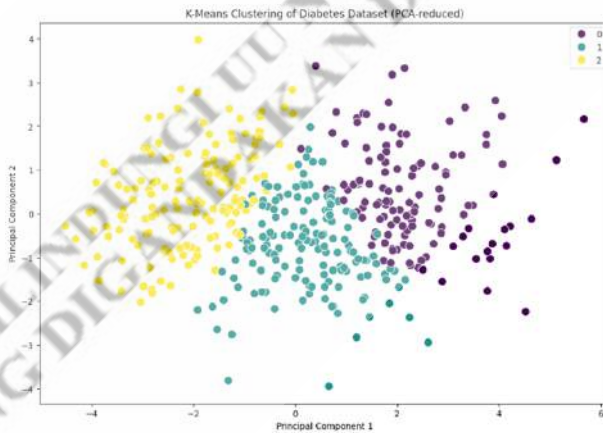
Dalam metode clustering, jumlah cluster (k) harus ditentukan sebelum melakukan proses clustering. Terdapat beberapa metode yang dapat digunakan untuk menentukan jumlah k yang akan digunakan, salah satunya adalah metode elbow. Pada contoh algoritma di atas metode elbow yang digunakan menghasilkan nilai k sebesar 3 sebagai jumlah cluster terbaik pada dataset yang digunakan.

Validasi *cluster* merupakan suatu tahap kegiatan yang digunakan untuk memberikan penilaian hasil dari analisis suatu *cluster* dengan berdasarkan jumlah atau kuantitatif dan sesuai dengan keadaan yang sebenarnya atau objektif. Selain hal tersebut validasi *cluster* juga salah satu metode yang digunakan untuk melakukan evaluasi hasil dari sebuah algoritma *clustering* yang telah dilakukan dengan tujuan untuk memperoleh jumlah *cluster* terbaik. Validasi hasil *clustering* dapat menggunakan beberapa indeks diantaranya yaitu Indeks *Dunn*, Indeks *Silhouette* dan Indeks *Connectivity*. Pada Indeks *Connectivity* membentuk jumlah *cluster* terbaik apabila nilai yang dihasilkan semakin kecil dibandingkan dengan nilai dari *cluster* yang lain. Sedangkan untuk Indeks *Dunn* menghasilkan *cluster* terbaik jika nilai *Dunn* yang diperoleh semakin besar dan untuk nilai Indeks *Dunn* yang besar atau tinggi tersebut menerangkan jika *cluster* yang terbentuk telah terpecah antara *cluster* yang satu dengan yang lainnya dengan teratur dan penuh atau padat. Kemudian untuk Indeks *Silhouette* menghasilkan *cluster* terbaik jika nilai yang diperoleh semakin dekat dengan angka 1, dalam indeks *Silhouette* dari hasil perhitungannya memiliki nilai antara -1 sampai dengan nilai 1 (Susilowati et.al,2021). Visualisasi hasil clustering digunakan untuk mempermudah melakukan pengolahan pengelompokan data, dimana untuk visualisasi dilakukan pembuatan dashboard clustering dengan menggunakan R Shiny. Gambar 158 menunjukkan hasil pengelompokan data dengan jumlah cluster sebanyak 3, yaitu cluster 1 digambarkan dengan lingkaran warna biru, cluster 2 dengan lingkaran warna hijau dan cluster 3 berwarna oranye.



Gambar 159. Grafik Elbow method untuk clustering pasien diabetes

Gambar 160 menunjukkan hasil clustering yang dihasilkan dari dataset yang digunakan. Pemetaan dilakukan berdasarkan tingkat kedekatan dari masing-masing data terhadap titik *centroid*.



Gambar 160. Hasil clustering pasien diabetes

Algoritma K-Means, K-Medoids, dan DBSCAN dapat digunakan untuk mengelompokkan data. Penggunaan algoritma K-Means sangat sensitif untuk menginisialisasi pusat cluster karena dilakukan secara acak. Algoritma K-Means menggunakan nilai rata-rata sebagai pusat cluster.

Berikut langkah-langkah algoritma K-Means.

- a. Pilih nilai k sebagai pusat cluster awal secara acak.
- b. Masing-masing data dibagi menjadi k cluster dan pusat cluster yang diperoleh dengan menggunakan Euclidean Distance.
- c. Setiap pusat cluster dihitung ulang berdasarkan nilai rata-rata dalam cluster yang diperoleh.
- d. Ulangi langkah kedua dan ketiga jika ada perubahan pada grup cluster. Proses akan berhenti jika tidak ada perubahan pada cluster.

Algoritma K-Medoids menerapkan objek sebagai perwakilan (medoid) untuk setiap cluster. Penerapan algoritma K-Medoids membutuhkan waktu lebih lama dibandingkan K-Means.

Langkah-langkah untuk menyelesaikan algoritma K-Medoids adalah sebagai berikut.

- a. Inisialisasi pusat cluster dengan jumlah cluster (k).
- b. Setiap data atau objek dialokasikan ke cluster terdekat menggunakan Euclidian Distance.
- c. Pilih secara acak objek di setiap cluster sebagai kandidat medoid baru.
- d. Hitung jarak setiap objek yang terdapat pada setiap cluster dengan calon medoid baru.
- e. Hitung simpangan total (S) dengan menghitung nilai total jarak baru – total jarak lama. Jika $S < 0$ diperoleh, tukarkan objek tersebut dengan cluster data untuk membuat kumpulan k objek baru sebagai medoid.
- f. Ulangi langkah tiga sampai lima hingga tidak terjadi perubahan pada medoid sehingga diperoleh cluster dan anggota cluster. DBSCAN adalah metode pengelompokan yang membangun cluster berdasarkan kepadatan, cluster yang tidak termasuk dalam objek dianggap noise. Praktek

DBSCAN memerlukan waktu yang sangat lama karena penggunaan metode ini dilakukan dengan mencari epsilon dan min point secara acak untuk mendapatkan cluster tertentu.

Langkah-langkah untuk menyelesaikan algoritma DBSCAN adalah sebagai berikut.

- a. Menginisialisasi parameter min, parameter eps.
- b. Tentukan titik awal atau p secara acak.
- c. Ulangi langkah 3 - 5 hingga semua poin telah diproses.
- d. Hitung eps atau semua titik jarak yang kepadatannya dapat dicapai hingga p.
- e. Jika titik yang cocok dengan eps lebih dari satu titik kecil, maka titik p merupakan titik inti, dan terbentuklah gugus.
- f. Jika p merupakan titik batas dan tidak ada titik yang kepadatannya dapat dicapai p, maka proses dilanjutkan ke titik lainnya. (Berahmana et.al,2020)

Contoh Kasus Clustering (Turnip Y.E.H., Fitriana R.,2023)

Dalam melakukan penelitian ini, pengumpulan data dilakukan dengan mengambil data luas lahan berdasarkan jenis hutan melalui Badan Pusat Statistik untuk wilayah Jawa Tengah data yang diambil terdiri dari data luas lahan untuk 4 jenis hutan berdasarkan fungsinya, yaitu hutan lindung, kawasan perlindungan, kawasan untuk produksi, dan kawasan untuk pengguna lain. Pada informasi data yang didapatkan, luasan lahan dibagi untuk masing-masing kabupaten di provinsi Jawa Tengah. Masing-masing kabupaten ini selanjutnya akan dilakukan pengelompokan berdasarkan luas hutan yang dimiliki untuk masing-masing jenis hutan menggunakan metode *K-Means*.

Sebelum melakukan pemilihan atribut yang digunakan, data yang didapatkan perlu diketahui apakah mengandung *missing data* yang perlu diganti. Pada aplikasi Rapidminer Studio, operator

Replace Missing Values dapat digunakan dan diatur untuk mengganti *missing data* yang ada dengan nilai rata-rata. Selanjutnya dilakukan pemilihan data yang akan digunakan, yaitu atribut harga tertinggi, harga terendah, dan harga rata-rata menggunakan operator *Select Attributes*. Tabel 1 menunjukkan contoh data yang dihasilkan setelah melakukan pemilihan atribut yang akan digunakan. Atribut-atribut yang dipilih selanjutnya akan dikelompokkan ke dalam kluster-kluster tertentu sesuai dengan nilai atribut-atribut yang digunakan.

Tabel 22. Data luas hutan yang telah mengalami *preprocessing*

Kabupaten	Hutan Lindung	Kawasan Perlindungan	Kawasan Produksi	Kawasan Pengguna Lain
Kabupaten Cilacap	3655,936	21304,08	24876,49	2508,5
Kabupaten Banyumas	7985,22	4976,08	14928,13	800,27
Kabupaten Purbalingga	10944,03	659	3360,46	110,72
Kabupaten Banjarnegara	2370,93	3549,2	10941,47	233,66
Kabupaten Kebumen	3982,56	1739,14	12654,89	583,14
Kabupaten Purworejo	3655,936	927	7408,03	497,64
Kabupaten Wonosobo	4255,1	3388,5	10623,2	138,19
Kabupaten Magelang	1472,95	725,9	3023,08	53,47

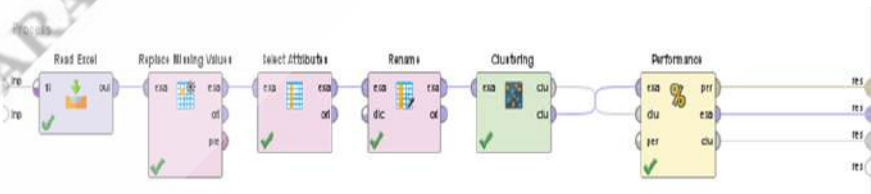
Kabupaten Boyolali	3655,936	1690,4	11207,28	1107,43
Kabupaten Klaten	52,3	4,6	567,4	18,6
Kabupaten Sukoharjo	304,15	0,2	70,1	990,274
Kabupaten Wonogiri	11842,95	1009,47	7002,13	181,91
Kabupaten Karanganyar	7080,2	4,7	118,7	135,6
Kabupaten Sragen	54,5	524,4	4500,7	61,09
Kabupaten Grobogan	3655,936	5621,42	60249,34	3863,1
Kabupaten Blora	3655,936	8818,87	67404,9	5308,81
Kabupaten Rembang	2446,46	2406,02	18215,4	948,91
Kabupaten Pati	1118,61	3153,21	16963,57	1259,38
Kabupaten Kudus	1288,85	783,06	321,9	1241,05
Kabupaten Jepara	2707,81	5104,15	5706,21	388,55
Kabupaten Demak	3655,936	596,42	2720,6	142
Kabupaten Semarang	1898,87	2797,53	5336,02	947,13
Kabupaten Temanggung	3273,32	2730,9	7263,9	238,01
Kabupaten Kendal	1702,29	3484,3	11720,28	1719,08
Kabupaten Batang	3080,74	6691,99	6730,58	1363,04

Kabupaten Pekalongan	1800,87	15886,2	9959,44	358,43
Kabupaten Pemalang	5269,42	2221,68	23104,17	2171,62
Kabupaten Tegal	2901,52	2990,44	17245,93	670,64
Kabupaten Brebes	6252,88	6787,05	36384,93	677,71

Clustering dengan K-Means

Pada penelitian ini, penentuan jumlah *cluster* yang akan diaplikasikan didasarkan pada nilai DB. Tabel 23 memperlihatkan nilai DB untuk clustering menggunakan 2-4 cluster. Pada pembagian menjadi 2 cluste, nilai DB sebesar 0.436 yang merupakan nilai DB terkecil. Sedangkan untuk pembagian 3 dan 4 *cluster* masing-masing menghasilkan nilai DB sebesar 0.635 dan 0.735. Dari nilai DB yang dihasilkan maka dapat disimpulkan bahwa pembagian menjadi 2 *cluster* dengan nilai DB 0.436, menjadi pilihan terbaik dan akan dilanjutkan dengan penentuan jarak dengan titik *centroid* pada masing-masing atribut yang diteliti.

Data yang telah siap untuk digunakan selanjutnya akan diolah menggunakan *K-Means clustering*. Gambar 161 menunjukkan model yang digunakan dalam melakukan proses clustering menggunakan Rapidminer. Proses diawali dengan memasukkan data menggunakan operator *Read Excel* untuk memasukkan data yang akan diolah dan dilanjutkan dengan pemilihan atribut. Data selanjutnya diolah menggunakan operator *Clustering* dan diakhiri dengan operator *Performance*.



Gambar 161. Model analisa *K-Means* menggunakan Rapidminer Studio

Tabel 23. Indeks DB untuk jumlah cluster 2-4

Jumlah cluster	Davies Bouldin
2	0.436
3	0.635
4	0.735

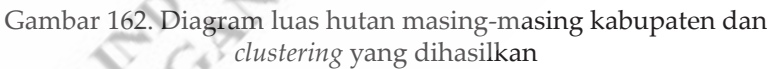
Tabel 24 merupakan tabel jarak antara tiap attribute yang terbentuk dengan titik centroid. Dari tabel tersebut dapat disimpulkan bahwa *cluster 1* memiliki jarak terdekat dengan titik *centroid* untuk keempat jenis hutan dibandingkan *cluster 2*. Hutan lindung memiliki jarak titik *centroid* sebesar 3556,054, sedangkan pada *cluster 2* memiliki jarak titik *centroid* sebesar 4521,584. Begitu pula dengan jenis hutan yang lain, dimana jarak titik *centroid* untuk *cluster 1* masing-masing adalah 3436,483 pada kawasan perlindungan, 9093,848 pada kawasan produksi, dan 725,705 pada kawasan pengguna. Nilai ini lebih kecil dibandingkan *cluster 2* yaitu masing-masing 7075,780 pada kawasan perlindungan, 54679,723 pada kawasan untuk produksi, dan 3283,207 pada kawasan untuk pengguna lain. Tabel 25 menunjukkan pembagian kabupaten ke dalam 2 cluster yang ditentukan Pada tabel tersebut terlihat bahwa hanya ada 3 kabupaten yang masuk ke dalam *cluster 2*, yaitu kabupaten Grobogan, kabupaten Blora, dan kabupaten Brebes, sedangkan sebanyak 26 kabupaten masuk ke dalam cluster 1.

Tabel 24. Tabel jarak titik *centroid* antara *cluster 1* dan *cluster 2*

Attribute	Cluster 1	Cluster 2
Hutan Lindung	3556.054	4521.584
Kawasan Perlindungan	3436.483	7075.780
Kawasan untuk Produksi	9098.848	54679.723
Kawasan untuk Pengguna Lain	725.705	3283.207

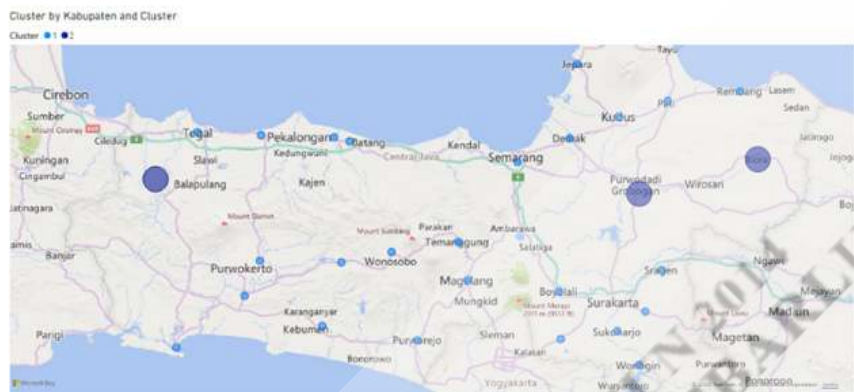
Visualisasi clustering yang dilakukan dapat dilihat pada Gambar 162 dan Gambar 163. Gambar 162 merupakan diagram batang yang menunjukkan luas area hutan berdasarkan jenis hutan

CLUSTERING KABUPATEN BERDASARKAN LUAS HUTAN DI PROVINSI JAWA TENGAH



212 | Intelijensia Bisnis dan Data Analitik

CLUSTERING KABUPATEN BERDASARKAN LUAS HUTAN DI
PROVINSI JAWA TENGAH



Gambar 163. Peta pembagian kabupaten berdasarkan *clustering* yang dilakukan

Tabel 25. Pembagian kabupaten dengan *K-Means*

Kabupaten	Cluster	Kabupaten	Cluster
Kabupaten Cilacap	1	Kabupaten Blora	2
Kabupaten Banyumas	1	Kabupaten Rembang	1
Kabupaten Purbalingga	1	Kabupaten Pati	1
Kabupaten Banjarnegara	1	Kabupaten Kudus	1
Kabupaten Kebumen	1	Kabupaten Jepara	1
Kabupaten Purworejo	1	Kabupaten Demak	1
Kabupaten Wonosobo	1	Kabupaten Semarang	1
Kabupaten Magelang	1	Kabupaten Temanggung	1

Kabupaten Boyolali	1	Kabupaten Kendal	1
Kabupaten Klaten	1	Kabupaten Batang	1
Kabupaten Sukoharjo	1	Kabupaten Pekalongan	1
Kabupaten Wonogiri	1	Kabupaten Pemalang	1
Kabupaten Karanganyar	1	Kabupaten Tegal	1
Kabupaten Sragen	1	Kabupaten Brebes	2
Kabupaten Grobogan	2		

E. Asosiasi

Asosiasi (*Association rule*) merupakan suatu algoritma yang digunakan untuk menemukan suatu hubungan atau pola diantara sekumpulan data. (Sornalakhsmi et al., 2020). Secara umum, teknik ini digunakan untuk mengetahui persentase dari aktivitas pembelian untuk suatu barang dalam suatu periode (Santoso, 2021). Itemset merupakan sekumpulan data yang terdiri dari satu atau beberapa data. Algoritma ini umumnya digunakan untuk kemunculan suatu barang yang dibeli oleh konsumen berdasarkan barang lain yang dibeli.

Pembelajaran aturan asosiasi adalah metode yang terkenal dan banyak digunakan untuk mengungkap hubungan menarik antara variabel dalam basis data besar dalam penambangan data. Penambangan penggunaan web, deteksi intrusi, dan bioinformatika adalah contoh domain di mana aturan asosiasi digunakan akhir-akhir ini. Terlepas dari jenis pola yang ditambang, pengukuran ketertarikan memainkan peran penting dalam penambangan data [2-3]. Kriteria ini digunakan untuk memilih dan memberi peringkat pola berdasarkan minat prospektif pengguna terhadapnya.

Pengukuran yang baik juga memungkinkan pengurangan biaya waktu dan ruang proses penambangan. Aturan asosiasi untuk mendeteksi keteraturan antara produk dalam data transaksi skala besar yang ditangkap oleh sistem point-of-sale (POS) di supermarket diperkenalkan berdasarkan konsep aturan yang kuat. (Krishnamoorthy M, Karthikeyan R, 2022)

```
import pandas as pd
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import apriori, association_rules

# Sample transactional data
transactions = [
    ['Oli', 'Lampu_Belakang', 'Filter'],
    ['Oli', 'Lampu_Belakang'],
    ['Oli', 'Lampu_Belakang', 'Kampas_Rem'],
    ['Oli', 'Kampas_Rem'],
    ['Lampu_Belakang', 'Filter'],
    ['Lampu_Belakang', 'Kampas_Rem'],
    ['Oli', 'Lampu_Belakang', 'Filter', 'Kampas_Rem']
]

# Convert the transactions into a one-hot encoded DataFrame
te = TransactionEncoder()
te_ary = te.fit_transform(transactions)
df = pd.DataFrame(te_ary, columns=te.columns_)

# Apply the Apriori algorithm to get frequent itemsets
frequent_itemsets = apriori(df, min_support=0.4,
                             use_colnames=True)
print("Frequent Itemsets:")
print(frequent_itemsets)

# Generate association rules from the frequent itemsets
rules = association_rules(frequent_itemsets,
                          metric="confidence", min_threshold=0.7)
```

```
print("\nAssociation Rules:")
print(rules)
```

Hasil dari program tersebut adalah sebagai berikut:

support		itemsets	
0	0.428571	(Filter)	
1	0.571429	(Kampas_Rem)	
2	0.857143	(Lampu_Belakang)	
3	0.714286	(Oli)	
4	0.428571	(Lampu_Belakang, Filter)	
5	0.428571	(Lampu_Belakang, Kampas_Rem)	
6	0.428571	(Oli, Kampas_Rem)	
7	0.571429	(Lampu_Belakang, Oli)	
antecedents		consequents	antecedent support
consequent support \			
0	(Filter)	(Lampu_Belakang)	0.428571
0.857143			
1	(Kampas_Rem)	(Lampu_Belakang)	0.571429
0.857143			

2	(Kampas_Rem)	(Oli)	0.571429
0.714286			

3	(Oli)	(Lampu_Belakang)	0.714286
0.857143			

	support	confidence	lift	leverage	conviction
zhangs_metric					

0	0.428571	1.00	1.166667	0.061224	inf
0.250000					

1	0.428571	0.75	0.875000	-0.061224	0.571429
-0.250000					

2	0.428571	0.75	1.050000	0.020408	1.142857
0.111111					

3	0.571429	0.80	0.933333	-0.040816	0.714286
-0.200000					

F. Support Vector Machine

Support Vector Machine (SVM) telah menjadi algoritma pembelajaran mesin yang sangat kuat yang dikembangkan untuk masalah klasifikasi dan prediksi, yang bekerja dengan mengenali pola melalui trik kernel. Karena kinerjanya yang tinggi dan kemampuan generalisasi yang hebat dibandingkan dengan metode klasifikasi lainnya, metode SVM banyak digunakan dalam bidang bioinformatika, pengenalan teks dan gambar, serta keuangan, dan salah satunya pada prediksi. Pada dasarnya, metode ini menemukan batas linier (hyperplane) yang mewakili margin terbesar antara dua kelas (label) dalam ruang masukan. Metode ini dapat diterapkan tidak hanya pada data linier tetapi juga pada data yang bersifat nonlinier dengan menggunakan fungsi kernel (M. Ryu and K. Lee, 2020.). Meskipun pertama kali dibuat untuk memecahkan

masalah klasifikasi, pekerjaan regresi juga dapat ditangani dengan digunakan sebagai analisis prediktif [16]. Support Vector Regression adalah adaptasi dari support machine learning berdasar klasifikasi model regresi dari Support Vector Machine (SVM) [17]. SVR merupakan metode yang dapat menyelesaikan permasalahan estimasi nonlinear. SVR berhasil memprediksi data deret waktu seperti pada data finansial deret waktu. Menurut Cortez dan Vapnik, SVR menggunakan model linear untuk diterapkan untuk kelas nonlinear melalui pemetaan vektor masukan x ke dalam ruang fitur high – dimensional [18]. Cara kerja SVR yaitu dengan mencari maksimum margin dari hyperplane dan mencari error terkecil. Data yang paling dekat dengan maksimum margin hyperplane disebut support vector [19]. Jika dalam plot data yang sudah divisualisasikan layak atau feasible atau $\varepsilon=0$, maka akan diperoleh regresi yang sempurna, akan tetapi jika terdapat ketidaklayakan atau infeasibility, dimana ada beberapa titik yang keluar dari rentang atau ambang batas tersebut, maka akan ada penambahan variabel slack ξ dan ξ^* untuk melakukan optimasi error yang sudah ada. Penjelasan ini bisa dilihat pada gambar 2.2. Data Warehouse with Kettle.Multimatics.

Secara umum persamaan SVM ditunjukkan oleh persamaan [19]:

$$f(x)=w^T\varphi(x)+b \quad (1)$$

Dimana w adalah faktor bobot (weight), b adalah ambang batas atau threshold, dan $\varphi(x)$ adalah fungsi yang memetakan input x ke dalam ruang feature. Untuk melakukan minimasi fungsi risiko yang sudah didefinisikan dapat dilakukan dengan melakukan estimasi terhadap koefisien w dan b dengan persamaan berikut:

$$\min \frac{1}{2} \|w\|^2 + C \sum (\xi + \xi^*) \quad \lambda_i = 1 \quad (2)$$

Dengan batasan:

$$y_i - \langle w\varphi(x_i) \rangle - b \leq \varepsilon + \xi \quad (3)$$

$$\langle w\varphi(x_i) \rangle + b - y_i \leq \varepsilon + \xi^* \quad (4)$$

$$\xi, \xi^* \geq 0 \quad (5)$$

Pada persamaan di atas konstanta C memiliki syarat $C > 0$,

yaitu menentukan trade off antara kerataan fungsi f dan batas ε (epsilon), sedangkan faktor $\|w\|_2$ disebut juga sebagai regularisasi [20]. Setiap titik atau koordinat nilai yang berada di luar batas ε akan dikalikan C . Penetapan nilai C yang besar berarti menekankan pentingnya faktor ε dibandingkan faktor kerataan fungsi. Gambar 2.3 menggambarkan situasi secara grafis penambahan peubah slack ξ dan ξ^* . Peubah slack yang terletak dalam selang ε akan bernilai nol, kemudian semakin jauh jarak peubah slack terhadap batas ε maka nilainya semakin besar. Support vector adalah titik-titik dalam data latih (training) yang terletak tepat garis batas dan di luar batas ε . Semakin kecil nilai ε akurasi semakin tinggi, jumlah support vector semakin banyak, dan nilai peubah slack juga semakin tinggi.

Untuk mengatasi masalah regresi ini, produk internal persamaan (1) dapat diganti dengan fungsi kernel $K()$. Hal ini memungkinkan melakukan operasi seperti itu dalam dimensi lebih besar dengan input data ruang dimensi rendah tanpa mengetahui transformasi ϕ [21], seperti yang ditunjukkan pada persamaan (6). Ini disebut fungsi kernel:

$$f(x) = \sum (\alpha_i - \alpha) \cdot K(x_i, x_j) + b \quad N_i=1 \quad (6)$$

Masalah kuadrat optimasi dikaitkan dengan pengali Lagrange α^* dan α . SVR menggunakan kernel yang dapat menghubungkan data ke ruang fitur yang lebih besar, memungkinkan pemrosesan data yang lebih terstruktur dan komputasi yang lebih efisien. Fungsi kernel dalam SVR memiliki lebih banyak ketepatan daripada SVR linear standar dalam regresi nonlinear [22]. Pada penelitian ini, menggunakan 3 jenis fungsi kernel yaitu kernel linear, kernel polinomial, dan kernel Radial Basis Function (RBF). Formula dari tiga fungsi Kernel tersebut adalah sebagai berikut (A. Hermawan, *et. al.*, 2022):

1. Kernel Linear

$$K(x_i, x_j) = x_i^T x_j \quad (7)$$

Dengan x_i, x_j pasangan dua data dalam training set

2. Kernel Polinomial

$$K(x_i, x_j) = (x_i^T x_j + 1)^d, d > 1 \quad (8)$$

d adalah derajat parameter yang diperlukan

3. Kernel Radial Basis Function

$$K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right) \quad (9)$$

Algoritma Grid Search

Algoritma Grid Search dapat digunakan untuk menemukan parameter yang ideal. Algoritma ini membagi jangkauan parameter yang akan dioptimalkan ke dalam grid dan melintasi setiap titik untuk mendapatkan parameter yang ideal. Dalam aplikasinya, algoritma grid search harus dikontrol oleh berbagai metrik kinerja, yang biasanya diukur melalui uji silang validasi (cross-validation) pada data pelatihan (training). Oleh karena itu, disarankan untuk mencoba beberapa pasangan parameter pada hyperplane SVR. Pasangan parameter yang menghasilkan uji cross-validation yang paling akurat adalah yang terbaik. Metode cross-validation yaitu metode yang digunakan saat memilih kombinasi model dan hyperparameter yang secara otomatis memvalidasi setiap kombinasi (S. Xie, *et.al*, 2023). Pasangan hyperparameter yang optimal memiliki nilai error terkecil dan akurasi tertinggi.

G. Rangkuman

Analitika data adalah pengumpulan, transformasi, dan pengorganisasian data untuk menarik kesimpulan, membuat prediksi, dan mendorong pengambilan keputusan yang tepat.

Klasifikasi merupakan salah satu predictive analysis untuk menemukan sebuah model yang mampu menjelaskan dan memisahkan kelas data. Tugas utama algoritma klasifikasi adalah membuat suatu prototipe yang bisa digunakan untuk melakukan pengenalan dan klasifikasi dari suatu data yang belum diketahui kelasnya.

Clustering merupakan suatu *tools* yang sangat berguna

dalam analitika data dimana metode ini mengelompokkan suatu kumpulan data yang dikelompokkan berdasarkan kesamaan terbesar dalam satu *cluster* dan perbedaan terbesar diantara *cluster* yang berbeda. Algoritma K-Means, K-Medoids, dan DBSCAN dapat digunakan untuk mengelompokkan data. Penggunaan algoritma K-Means sangat sensitif untuk menginisialisasi pusat cluster karena dilakukan secara acak. Algoritma K-Means menggunakan nilai rata-rata sebagai pusat cluster.

Berikut langkah-langkah algoritma K-Means:

1. Pilih nilai k sebagai pusat cluster awal secara acak.
2. Masing-masing data dibagi menjadi k cluster dan pusat cluster yang diperoleh dengan menggunakan Euclidean Distance.
3. Setiap pusat cluster dihitung ulang berdasarkan nilai rata-rata dalam cluster yang diperoleh.
4. Ulangi langkah kedua dan ketiga jika ada perubahan pada grup cluster. Proses akan berhenti jika tidak ada perubahan pada cluster.

Algoritma K-Medoids menerapkan objek sebagai perwakilan (medoid) untuk setiap cluster. Penerapan algoritma K-Medoids membutuhkan waktu lebih lama dibandingkan K-Means.

Langkah-langkah untuk menyelesaikan algoritma K-Medoids adalah sebagai berikut:

- a. Inisialisasi pusat cluster dengan jumlah cluster (k).
- b. Setiap data atau objek dialokasikan ke cluster terdekat menggunakan Euclidian Distance.
- c. Pilih secara acak objek di setiap cluster sebagai kandidat medoid baru.
- d. Hitung jarak setiap objek yang terdapat pada setiap cluster dengan calon medoid baru.
- e. Hitung simpangan total (S) dengan menghitung nilai total jarak baru – total jarak lama. Jika $S < 0$ diperoleh, tukarkan

objek tersebut dengan cluster data untuk membuat kumpulan k objek baru sebagai medoid.

- f. Ulangi langkah tiga sampai lima hingga tidak terjadi perubahan pada medoid sehingga diperoleh cluster dan anggota cluster. DBSCAN adalah metode pengelompokan yang membangun cluster berdasarkan kepadatan, cluster yang tidak termasuk dalam objek dianggap noise. Praktek DBSCAN memerlukan waktu yang sangat lama karena penggunaan metode ini dilakukan dengan mencari epsilon dan min point secara acak untuk mendapatkan cluster tertentu.

Langkah-langkah untuk menyelesaikan algoritma DBSCAN adalah sebagai berikut.

- a. Menginisialisasi parameter min, parameter eps.
- b. Tentukan titik awal atau p secara acak.
- c. Ulangi langkah 3 - 5 hingga semua poin telah diproses.
- d. Hitung eps atau semua titik jarak yang kepadatannya dapat dicapai hingga p.
- e. Jika titik yang cocok dengan eps lebih dari satu titik kecil, maka titik p merupakan titik inti, dan terbentuklah gugus.
- f. Jika p merupakan titik batas dan tidak ada titik yang kepadatannya dapat dicapai p, maka proses dilanjutkan ke titik lainnya. (Berahmana et.al,2020)

Algoritma asosiasi merupakan suatu algoritma yang digunakan untuk menemukan suatu hubungan atau pola diantara sekumpulan data. Secara umum, teknik ini digunakan untuk mengetahui persentase dari aktivitas pembelian untuk suatu barang dalam suatu periode. Algoritma ini umumnya digunakan untuk kemunculan suatu barang yang dibeli oleh konsumen berdasarkan barang lain yang dibeli.

H. Soal Latihan

1. Jelaskan konsep analitika data.
2. Sebutkan dan jelaskan jenis analitika data.
3. Jelaskan perbedaan klasifikasi dan clustering.
4. Jelaskan contoh kasus clustering.
5. Jelaskan contoh kasus support vector machine.

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

BAB VIII

Visualisasi Data

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu membuat visualisasi data dengan menggunakan *Power Business Intelligence*. Mahasiswa mampu melakukan penelitian di bidang teknik industri untuk meningkatkan produktivitas dan daya saing organisasi dengan *Power Business Intelligence*.

Materi yang diberikan :

1. *Power Business Intelligence*.
2. Penggunaan *Power Business Intelligence*.
3. Visualisasi Data.
4. Persiapan Data Set.
5. Tahapan Percobaan.

Kemampuan akhir yang diharapkan dihapus.

Capaian pembelajaran adalah mampu mengidentifikasi, memformulasikan dan menganalisis masalah rekayasa kompleks pada sistem terintegrasi dengan *Power Business Intelligence*.

B. Format Excel

Format Excel yang baik untuk *Power Business Intelligence*. Data berantakan : tidak bisa divisualisasikan, tidak bisa di drag and drop, karena data excel tidak pada layout yang ideal

Tabel 26. Tabel Data

Order ID	Order Date	Customer Name	Segment	City	State	Category	Sub-Category	Product Name	Sales	Quantity	Discount	Profit
CA-2016-152156	08/11/2016	Claire Gule	Consumer	Henderson	Kentucky	Furniture	Bookcases	Bush Somerset ES	261.96	3	0	41.9138
CA-2016-152156	08/11/2016	Claire Gule	Consumer	Henderson	Kentucky	Furniture	Chairs	Hon Deluxe Fabric Uph	731.94	3	0	219.362
CA-2016-138688	12/09/2016	Darrin Van Huff	Corporate	Los Angeles	California	Office Supply	Labels	Self-Adhesive Ad	14.62	2	0	6.874
US-2015-120966	11/16/2015	Sean O'Donnell	Consumer	Fort Lauderdale	Florida	Furniture	Tables	Bretford CH4500 S	957.5775	5	0.45	-189.811
US-2015-130966	03/06/2016	Brosina Hoffman	Consumer	Los Angeles	California	Office Supply	Storage	Eldon Fold N Roll	22.368	2	0.1	2.5.594
CA-2014-115612	09/06/2014	Brosina Hoffman	Consumer	Los Angeles	California	Furniture	Furnishings	Eldon Expressions	48.86	7	0	14.184
CA-2014-115612	09/06/2014	Brosina Hoffman	Consumer	Los Angeles	California	Office Supply	Art	Newell 322	7.28	4	0	1.906
CA-2014-115612	09/06/2014	Brosina Hoffman	Consumer	Los Angeles	California	Technology	Phones	Mittel S13010 Pho	907.153	6	0.3	40.7152
CA-2014-115612	09/06/2014	Brosina Hoffman	Consumer	Los Angeles	California	Office Supply	Binders	DOL Angle-View I	14.504	3	0.4	5.7925
CA-2014-115612	09/06/2014	Brosina Hoffman	Consumer	Los Angeles	California	Office Supply	Appliances	Belkin F5C200VTE	154.9	3	0	34.87
CA-2014-115612	09/06/2014	Brosina Hoffman	Consumer	Los Angeles	California	Furniture	Tables	Chromcraft Recta	1506.184	9	0.2	85.3092
CA-2014-115612	09/06/2014	Brosina Hoffman	Consumer	Los Angeles	California	Technology	Phones	Korffel 250 Confe	117.424	4	0.2	48.1568
CA-2017-141412	10/04/2017	Andrew Allen	Consumer	Seasport	North Carolina	Office Supply	Paper	Repro 1407	11.552	1	0.2	5.4463
CA-2016-101189	05/12/2016	Irene Maddox	Consumer	Seattle	Washington	Office Supply	Binders	Fellowes P1800 P	401.976	3	0.2	132.9122
US-2015-118981	22/11/2015	Harold Paulan	Home Office	Fort Worth	Texas	Office Supply	Appliances	Holmes R4200N	61.82	5	0.8	-129.858
US-2015-118981	22/11/2015	Harold Paulan	Home Office	Fort Worth	Texas	Office Supply	Binders	Storco Bulk-Tech I	2.544	3	0.8	-1.814
CA-2016-105093	11/11/2016	Pete Kitz	Consumer	Madison	Wisconsin	Office Supply	Storage	Stump-Stor 200 L	140.88	6	0	13.3178
CA-2014-167164	13/05/2014	Alejandro Greve	Consumer	West Jordan	Utah	Office Supply	Storage	Pellico 85 Super S	55.5	2	0	3.39
CA-2014-141836	27/09/2014	Zsuzsanna Dornetels	Consumer	San Francisco	California	Office Supply	Art	Newell 341	954	2	0	2.4024
CA-2014-141836	27/09/2014	Zsuzsanna Dornetels	Consumer	San Francisco	California	Technology	Phones	Apple 116 16GB L	213.48	3	0.2	16.011
CA-2014-141836	27/09/2014	Zsuzsanna Dornetels	Consumer	San Francisco	California	Office Supply	Binders	Wilson Jones 300	12.72	4	0.2	7.384
CA-2014-141836	27/09/2014	Zsuzsanna Dornetels	Consumer	San Francisco	California	Office Supply	Art	Newell 312	10.84	5	0	6.904

Tabel yang baik memiliki judul kolom. Hal ini bisa dilakukan dengan menggunakan Pivot.

Data yang baik adalah:

- 1. Memiliki judul kolom

Berikut adalah contoh tabel yang memiliki judul kolom

Tabel 27. Contoh Tabel yang memiliki judul kolom

	Segment	City	State	Product ID	Category	Sub-Category	Product Name	Product	Quantity
Claire Gule	Consumer	Henderson	Kentucky	FUR-BO-1000	Furniture	Bookcases	Bush Somerset Collecti	261.96	2
Claire Gule	Consumer	Henderson	Kentucky	FUR-CH-1000	Furniture	Chairs	Hon Deluxe Fabric Uph	731.94	3
Darrin Van Huff	Corporate	Los Angeles	California	OFF-LA-1000	Office Supplies	Labels	Self-Adhesive Address	14.62	2
Sean O'Donnell	Consumer	Fort Lauderdale	Florida	FUR-TA-1000	Furniture	Tables	Bretford CH4500 Series	957.5775	5
Sean O'Donnell	Consumer	Fort Lauderdale	Florida	OFF-ST-1000	Office Supplies	Storage	Eldon Fold N Roll Cart	22.368	2
Brosina Hoffman	Consumer	Los Angeles	California	FUR-FU-1000	Furniture	Furnishings	Eldon Expressions Woo	48.86	7
Brosina Hoffman	Consumer	Los Angeles	California	OFF-AR-1000	Office Supplies	Art	Newell 322	7.28	4



- 2. Tidak ada merged cell

Berikut adalah contoh tabel tidak ada merged cell

Tabel 28. Tidak ada merged cell

Data Customer				Data Product					
Name customer	Segment	City	State	Product ID	Category	Sub-Category	Product Name	Product	Quantity
Claire Gule	Consumer	Henderson	Kentucky	FUL-BO-1000	Furniture	Bookcases	Bush Somerset Collecti	261,96	2
Claire Gule	Consumer	Henderson	Kentucky	FUL-CH-1000	Furniture	Chairs	Hon Deluxe Fabric Uph	731,94	3
Darrin Van Huff	Corporate	Los Angeles	California	OFF-JA-1000	Office Supplies	Labels	Self-Adhesive Address	14,62	2
Sean O'Donnell	Consumer	Fort Lauderdale	Florida	FUL-TA-1000	Furniture	Tables	Brexford CMA500 Series	957,5775	5
Sean O'Donnell	Consumer	Fort Lauderdale	Florida	OFF-ST-1000	Office Supplies	Storage	Eldon Fold 'N Roll Cart	22,368	2
Brosina Hoffman	Consumer	Los Angeles	California	FUL-FU-1000	Furniture	Furnishings	Eldon Expressions Wood	48,96	7
Brosina Hoffman	Consumer	Los Angeles	California	OFF-AR-1000	Office Supplies	Art	Newell 322	7,28	4

3. Tidak ada duplikat judul kolom

Tabel 29. Tidak ada duplikat judul kolom

Purchased	Expended		
Date	Date	Type	Amount
01/02/2021	01/02/2021	Food	1
02/02/2021	02/02/2021	Medical	2
03/02/2021	03/02/2021	Food	3
04/02/2021	04/02/2021	Medical	4

4. Tidak ada data yang kosong

Berikut adalah contoh tidak ada data yang kosong

Tabel 30. Contoh data tidak kosong



Tahun	Bulan	Qty
2021	Jan	1
		2
		3
	Feb	4
		5
		6
2021	Jan	1
		2
		3
	Feb	4
		5
		6

Tahun	Bulan	Qty
2021	Jan	1
2021	Jan	2
2021	Jan	3
2021	Feb	4
2021	Feb	5
2021	Feb	6
2019	Jan	1
2019	Jan	2
2019	Jan	3
2019	Feb	4
2019	Feb	5
2019	Feb	6

5. Satu kolom memiliki 1 arti

Berikut adalah contoh satu kolom memiliki 1 arti

Region	OS - Type+	Qty
Regional A	Android - Tablet	11
Regional A	Android - Phone	12
Regional A	Windows - Tablet	12
Regional B	Android - Tablet	23
Regional B	Android - Phone	34
Regional B	Windows - Tablet	11
Regional C	Android - Tablet	45
Regional C	Android - Phone	34
Regional C	Windows - Tablet	14



Region	OS	Type	Qty
Regional A	Android	Tablet	11
Regional A	Android	Tablet	12
Regional A	Windows	Phone	12
Regional B	Android	Tablet	23
Regional B	Android	Tablet	34
Regional B	Windows	Phone	11
Regional C	Android	Tablet	45
Regional C	Android	Tablet	34
Regional C	Windows	Phone	14

6. Tidak ada grand total/sub total

Berikut adalah contoh tabel memiliki grand total/sub total

Nama customer	Segment	City	State	Product ID	Category	Sub-Category	Product Name	Product	Quantity
Claire Gute	Consumer	Henderson	Kentucky	FUR-BO-1000	Furniture	Bookcases	Bush Somerset Collecti	261,96	2
Claire Gute	Consumer	Henderson	Kentucky	FUR-CH-1000	Furniture	Chairs	Hon Deluxe Fabric Uph	732	3
Darrin Van Huff	Corporate	Los Angeles	California	OFF-LA-1000	Office Supplies	Labels	Self-Adhesive Address	13	2
Sean O'Donnell	Consumer	Fort Lauderdale	Florida	FUR-TA-1000	Furniture	Tables	Bretford CR4500 Series	958	5
Sean O'Donnell	Consumer	Fort Lauderdale	Florida	OFF-ST-1000	Office Supplies	Storage	Eldon Fold 'N Roll Cart	22	2
Brosina Hoffman	Consumer	Los Angeles	California	FUR-FU-1000	Furniture	Furnishings	Eldon Expressions Woo	49	7
Brosina Hoffman	Consumer	Los Angeles	California	OFF-AA-1000	Office Supplies	Art	Newell 322	7	4
TOTAL								2.045	25



7. Update ke bawah bukan ke kanan

Berikut adalah contoh tabel update ke bawah

Tabel Update ke bawah (bukan ke kanan)

Product	2019					2020					2021		
	Amount	Price	Sales	Discount		Amount	Price	Sales	Discount		Amount	Price	Sales
TV	1	Rp12	Rp12	5%		1	Rp12	Rp12	5%		1	Rp12	Rp12
Blu-ray	2	Rp13	Rp26	5%		2	Rp26	Rp26	5%		2	Rp13	Rp26
Fridge	3	Rp14	Rp42	5%		3	Rp42	Rp42	5%		3	Rp14	Rp42
Console	4	Rp15	Rp60	5%		4	Rp15	Rp60	5%		4	Rp15	Rp60

Product	Tahun	Amount	Price	Sales
TV	2019	1	Rp12	Rp12
Blu-ray	2019	2	Rp13	Rp26
Fridge	2019	3	Rp14	Rp42
Console	2019	4	Rp15	Rp60
TV	2020	1	Rp12	Rp12
Blu-ray	2020	2	Rp13	Rp26
Fridge	2020	3	Rp14	Rp42
Console	2020	4	Rp15	Rp60
TV	2021	1	Rp12	Rp12
Blu-ray	2021	2	Rp13	Rp26

Power Businee Intelligence

Microsoft Power BI adalah *platform* visualisasi data yang populer dan memiliki berbagai fitur yang memungkinkan pengguna untuk menggabungkan data dari berbagai sumber, membersihkan dan mentransformasikan data, serta membuat visualisasi yang interaktif dan dinamis. Power BI adalah platform business intelligence dan visualisasi data mutakhir yang dikembangkan oleh Microsoft. Ini memberdayakan organisasi dan individu untuk mengubah data mentah menjadi wawasan berharga dan laporan serta *dashboard* yang menarik secara visual (Metre,2024). Power BI memiliki keunggulan dalam hal kemudahan penggunaan, integrasi dengan berbagai sumber data, dan fleksibilitas dalam pembuatan visualisasi yang menarik. Dalam konteks penelitian ini, visualisasi data dengan Power BI menjadi salah satu fokus utama untuk menggambarkan hasil analisis data dengan cara yang menarik dan informatif. Penggunaan Power BI diharapkan dapat memberikan kontribusi yang signifikan dalam memberikan wawasan yang lebih dalam terhadap data yang dianalisis serta mendukung proses pengambilan keputusan yang efektif dalam konteks studi ini.

Power Business Intellegent atau Power BI merupakan *platform* analisis yang mendukung penggunaan dari berbagai sumber data

untuk keperluan analisis dan pelaporan. *Microsoft* mengembangkan *Power BI* pada bulan September 2013 sebagai bagian dari *Office 365*. *Power BI* adalah platform visualisasi data yang populer dan memiliki berbagai fitur yang memungkinkan pengguna untuk menggabungkan data dari berbagai sumber, membersihkan dan mentransformasikan data, serta membuat visualisasi yang interaktif dan dinamis. *Power BI* adalah platform business intelligence dan visualisasi data mutakhir yang dikembangkan oleh *Microsoft*. Ini memberdayakan organisasi dan individu untuk mengubah data mentah menjadi wawasan berharga dan laporan serta dashboard yang menarik secara visual (K. V. Metre et al., 2024). Versi pertama *Power BI* dirilis secara luas pada tanggal 24 Juli 2015. *Business intelligence* adalah kegiatan yang umum dilakukan oleh pengusaha dalam mengelola data serta mendapatkan informasi yang lebih luas berdasarkan data tersebut. Data yang terkumpul kemudian disimpan dalam data *mart* selanjutnya dilakukan analisis menggunakan berbagai *tools*. Hal ini bertujuan untuk memperoleh informasi yang relevan bagi pengusaha dalam pengambilan keputusan bisnis (Febiyanti, 2022). *Power BI* memiliki keunggulan dalam hal kemudahan penggunaan, integrasi dengan berbagai sumber data, dan fleksibilitas dalam pembuatan visualisasi yang menarik. Menerapkan model sistem intelijen bisnis memberikan keuntungan bagi perusahaan karena teknik ini mengubah data mentah menjadi informasi berharga untuk kedepannya. Selain itu, intelijen bisnis dapat mengenali data yang tidak terstruktur, meningkatkan mutu produk, dan menciptakan peluang baru dalam strategi bisnis [30]. Dalam konteks penelitian ini, visualisasi data dengan *Power BI* menjadi salah satu fokus utama untuk menggambarkan hasil analisis data dengan cara yang menarik dan informatif. Penggunaan *Power BI* diharapkan dapat memberikan kontribusi yang signifikan dalam memberikan wawasan yang lebih dalam terhadap data yang dianalisis serta mendukung proses pengambilan keputusan yang efektif dalam konteks studi ini.

Manfaat dari pengaplikasian *Power BI* adalah sebagai berikut (Febiyanti, 2022):

1. Perbaharuan *dashboard* secara *real-time* memastikan bahwa informasi yang ditampilkan selalu terkini, memberikan

pengguna akses cepat ke data yang paling baru.

2. Keamanan diperkuat oleh koneksi langsung ke sumber data dan penyimpanan di *cloud*, menjamin bahwa data tetap aman dan tersedia dengan aksesibilitas yang tinggi.
3. Penggunaan fungsionalitas kueri bahasa alami dalam pengambilan data memungkinkan pengguna untuk berinteraksi dengan *dashboard* dengan cara yang lebih intuitif, meningkatkan efisiensi dalam mendapatkan informasi yang diinginkan.

Penggunaan BI menawarkan beberapa manfaat yang dapat mempercepat pengambilan keputusan (Widjaja dan T. Mauritsius, 2019):

1. Mengotomatiskan dan meningkatkan aspek intensif informasi seperti perencanaan bisnis rutin, manajemen kinerja, analisis varians, analisis akar penyebab, dan perencanaan tindakan perbaikan.
2. Membuat *dashboard* dan metrik pengukuran kinerja di era otomatisasi dan akselerasi. Fokus terhadap perhatian bisnis pada pelanggan utama dan saluran yang mendorong hasil yang diinginkan, mengukur kinerja terhadap sasaran pengukuran kinerja yang sudah ditetapkan, serta dapat memengaruhi manajemen dengan strategi pengecualian.
3. Mengotomatiskan analisis efektivitas promosi perdagangan untuk meningkatkan dukungan periklanan dan pemasaran konsumen dan mencapai hasil yang optimal.
4. Memberikan informasi historis bisnis yang standar dan komprehensif dalam waktu singkat mengenai masukan informasi/fakta ke dalam berbagai rencana, anggaran, dan pengendalian suatu perusahaan sehingga memungkinkan perencanaan yang efisien dan pengelolaan yang efektif.
5. Mendapatkan wawasan yang tepat waktu dan hemat biaya mengenai kinerja bisnis dan keuangan perusahaan melalui pandangan yang terstandarisasi dan dinamis mengenai profitabilitas dan kinerja pelanggan, segmen, kategori, merek, produk, dan jaringan yang berdampak signifikan terhadap

penjualan, biaya, dan layanan yang memungkinkan untuk dilakukan pemantauan secara efisien.

6. Menyediakan informasi dan analisis bisnis dalam layar yang sederhana dan mudah digunakan. Hal ini memungkinkan untuk menentukan variabel yang diminati dan data dasar untuk pelaporan dan analisis yang cepat. Profesional bisnis dapat melaporkan berbagai pengaturan penggunaan, termasuk laporan standar, *dashboard*, serta analisis prediktif.
7. Mengubah proses penyimpanan dan pelaporan saat ini untuk mengurangi atau menghilangkan masalah kinerja di lingkungan. Hal ini dirancang untuk memberikan kerangka kerja yang komprehensif untuk membantu organisasi mencapai tujuan tata kelola perusahaan dan manajemen sistem informasi.

C. Penggunaan Power Business Intelligence

1. Visualisasi Data

Visualisasi data berhubungan erat dengan pemilihan grafik yang tepat untuk menyajikan informasi dari data secara jelas dan mudah dimengerti. proses visualisasi ditujukan untuk merepresentasikan informasi yang terkandung dalam data secara efektif dan atraktif. Terdapat 4 aspek penting yang menjadi pertimbangan ketika memilih tipe grafik dalam visualisasi yaitu **Perbandingan, Distribusi, Relasi dan Komposisi**.

- a. **Visualisasi Perbandingan** menggambarkan perbedaan atau kesamaan antara komponen data yang berbeda. Perbandingan dapat dibagi antara unit, elemen atau kelompok serta deret waktu (*time series*). Grafik yang dapat digunakan untuk perbandingan antara lain adalah grafik batang (*bar chart*). Ini adalah grafik paling umum untuk perbandingan data yang bersifat kategorikal. Grafik batang menunjukkan perbedaan antar kategori secara jelas dengan menggunakan panjang batang. Grafik batang yang vertikal disebut juga grafik kolom (*column chart*). Dapat digunakan grafik garis (*line chart*) apabila terdapat elemen waktu dalam perbandingannya. Grafik garis cocok untuk

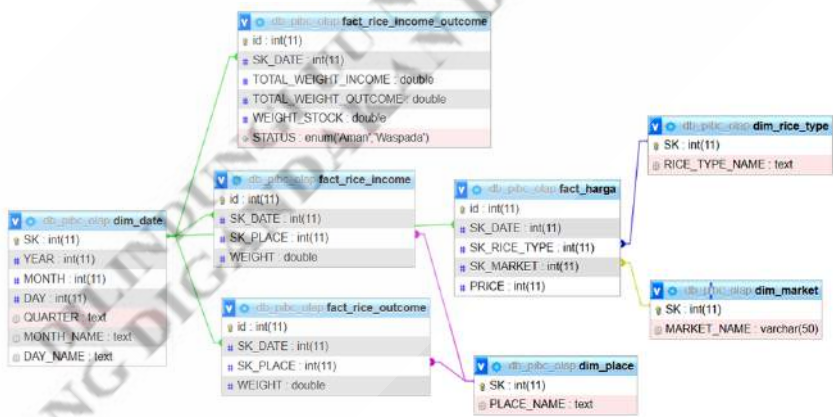
menunjukkan tren dari berbagai kategori atau kelompok selama periode waktu tertentu. Dalam buku ini, visualisasi perbandingan diimplementasikan menggunakan Power BI, Looker Studio dan Python.

- b. **Visualisasi Distribusi** menggambarkan bagaimana data didistribusikan dalam suatu dataset. Tujuan utama visualisasi distribusi adalah untuk melihat bagaimana frekuensi atau jumlah perubahan suatu peristiwa di seluruh rentang data. Grafik yang digunakan antara lain adalah *histogram*, *boxplot*, dan *density plot*. Histogram sangat cocok untuk menunjukkan distribusi frekuensi dari satu variabel numerik. Histogram memvisualisasikan seberapa sering nilai-nilai tertentu muncul dalam rentang tertentu. *Box plot* digunakan untuk menampilkan distribusi data dalam bentuk lima nilai ringkasan (minimum, kuartil pertama, median, kuartil ketiga, maksimum). Ini sangat berguna untuk mendeteksi adanya pencilan atau *outlier*. *Density plot* menunjukkan distribusi probabilitas data numerik. Ini mirip dengan histogram, tetapi lebih halus dengan pendistribusian menggunakan kurva.
- c. **Visualisasi Relasi** menggambarkan bagaimana dua atau lebih variabel terkait satu sama lain. Visualisasi relasi dapat membantu dalam memahami korelasi atau interaksi antar variabel. Grafik yang digunakan antara lain diagram pencar (*scatter plot*) dan *bubble chart*. Grafik ini cocok digunakan untuk melihat hubungan antara dua variabel numerik. *Scatter plot* menunjukkan apakah ada korelasi positif, negatif, atau tidak ada korelasi antara dua variabel. *Bubble chart* mirip dengan *scatter plot*, namun menggunakan ukuran lingkaran untuk menampilkan dimensi tambahan, sehingga cocok untuk menampilkan tiga variabel.
- d. **Visualisasi Komposisi** menggambarkan bagaimana seluruh bagian dibagi. Tujuannya untuk memperlihatkan persentase atau porsi dari total yang dimiliki oleh berbagai kategori atau elemen. Grafik yang dapat digunakan antara lain diagram lingkaran (*pie chart*) dan grafik batang bertumpuk (*stacked bar chart*). Diagram lingkaran dapat

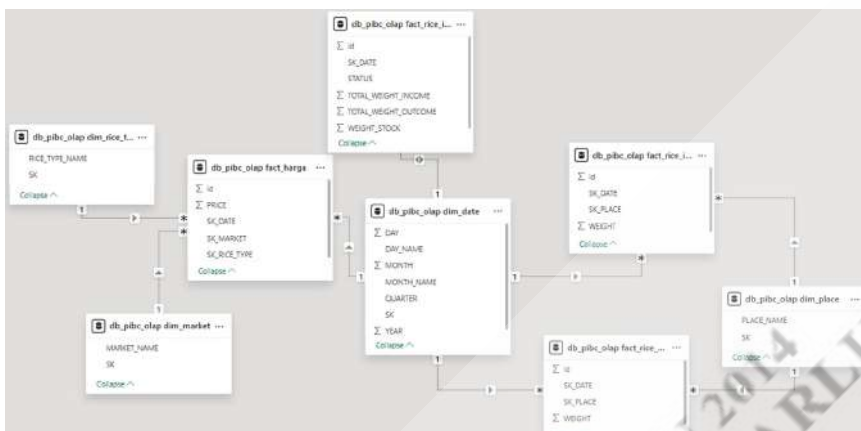
digunakan untuk menampilkan bagian dari keseluruhan dalam bentuk lingkaran. Grafik batang bertumpuk dapat digunakan untuk menampilkan persentase beberapa kategori dalam satu batang. Cocok untuk membandingkan kategori yang berbeda secara bersamaan.

2. Persiapan dataset

Sebagai ilustrasi di buku ini digunakan data yang berasal dari penelitian rantai pasok beras di DKI Jakarta. Secara umum rantai pasok beras ke DKI Jakarta memiliki pola yaitu usaha penggilingan beras dari berbagai sentra produksi mengirimkan berasnya kepada pedagang besar di Pasar Induk Beras Cipinang (PIBC) untuk kemudian disalurkan kembali kepada pengecer atau agen pemasaran di Jakarta untuk kemudian didistribusikan lagi kepada konsumen. Dataset tersimpan dalam database dengan relasi yang dapat dilihat pada Gambar 164. Relasi antar table yang tampil dalam power BI dapat dilihat pada Gambar 165.



Gambar 164. Disain relasi antar table dalam database OLAP



Gambar 165. Relasi antar tabel di Power BI

3. Tahapan Percobaan

Proses koneksi database MySQL ke powerBI dapat dilihat pada Langkah-langkah berikut ini :

- Install Power BI di laptop (disarankan tidak melalui Microsoft Store) <https://www.microsoft.com/en-us/download/details.aspx?id=58494>
- Pastikan Database MySQL sudah terpasang (XAMPP), apabila belum terpasang bisa melalui link berikut <https://www.apachefriends.org/> atau MySQL standalone di link berikut <https://dev.mysql.com/downloads/windows/installer/8.0.html> (jika memilih versi kecil/32 MB maka akan melakukan download tambahan di saat proses penginstalan berlangsung) lalu ikuti langkah penginstalan secara default.
- Install MySQL Connector <https://dev.mysql.com/downloads/connector/net/> lalu ikuti langkah penginstalan secara default.

General Availability (GA) Releases Archives

Connector/ODBC 8.4.0

Select Version:
8.4.0

Select Operating System:
Microsoft Windows

Platform	Version	Size	Action
Windows (x86, 64-bit), MSI Installer (mysql-connector-odbc-8.4.0-winx64.msi)	8.4.0	14.8M	Download
Windows (x86, 64-bit), ZIP Archive (mysql-connector-odbc-8.4.0-winx64.zip)	8.4.0	16.9M	Download
Windows (x86, 64-bit), ZIP Archive Debug Binaries & Test Suite (mysql-connector-odbc-8.4.0-winx64-debug.zip)	8.4.0	53.3M	Download

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

d. Hubungkan Power BI dengan MySQL Database

▼ Select a data source or start with a blank report

Blank report OneLake catalog Excel workbook SQL Server Learn with sample data **Get data from other sources**

▼ Recommended

Getting started

Intro—What is Power BI?

Get Data

Database

- SQL Server database
- Access database
- SQL Server Analysis Services database
- Oracle database
- IBM Db2 database
- IBM Informatica database (Beta)
- IBM Netezza
- MySQL database**
- Import data from a MySQL database
- Teradata database
- SAP HANA database
- SAP Business Warehouse Application Server
- SAP Business Warehouse Message Server
- Amazon Redshift
- Impala

Visualizations

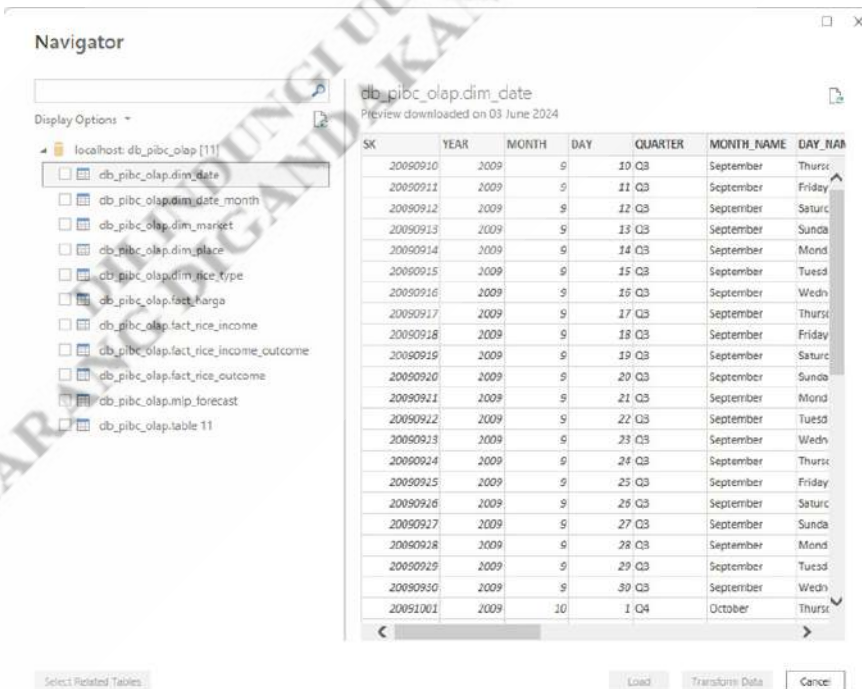
Fields



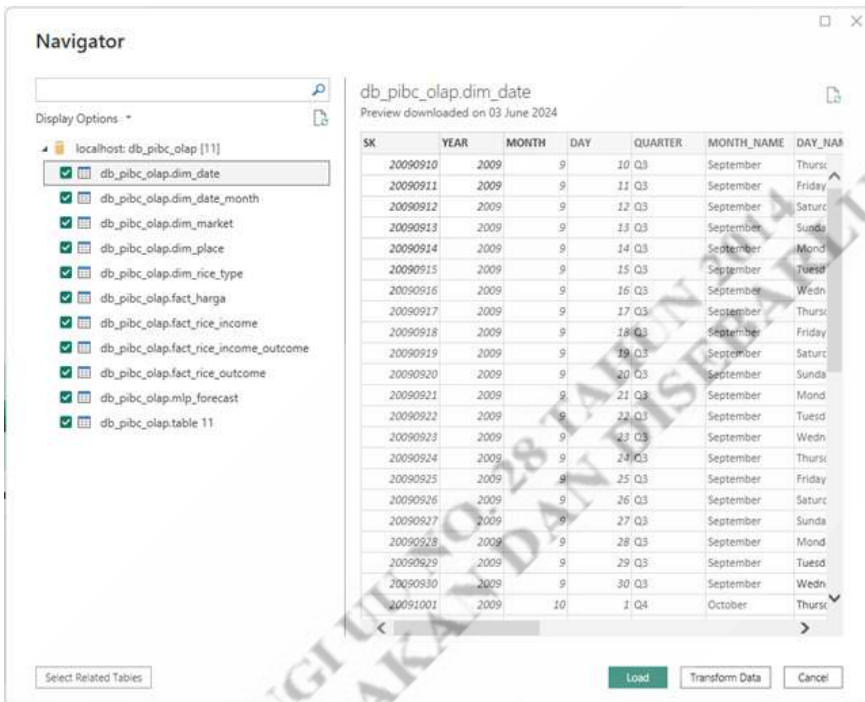
Gambar 166. Hubungkan Power BI dengan MySQL Database

*Notes: jika server MySQL tidak terletak pada port default (ada di 3306 biasanya), maka tambahkan port database server MySQL pada kolom 'Server'. Jika tidak, maka cukup tuliskan localhost saja. Contoh: localhost:3307. Dan apabila baru pertama kali mengkoneksi gunakan username 'root' dan kosongkan password.

e. Tampilan awal setelah Tabel di muat kedalam Power BI



- f. Pilih tabel pada database yang ingin dimuat kedalam Power BI (dalam hal ini menggunakan db_pibc_olap) lalu klik load.

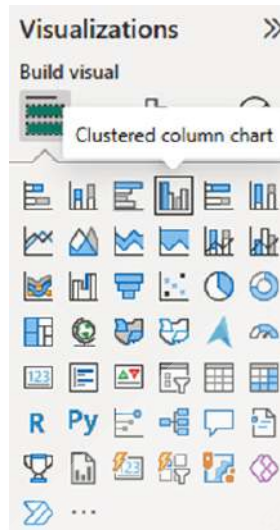


Gambar 167. Tampilan Power BI

4. Visualisasi Perbandingan

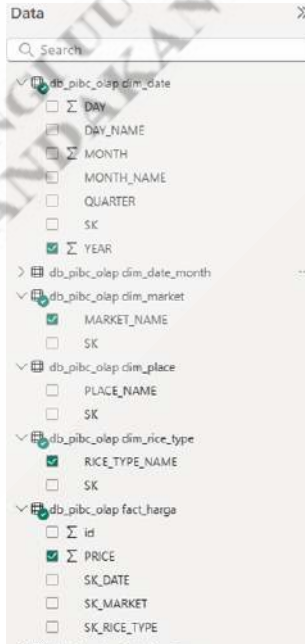
a. Clustered Column Chart

- 1) Untuk melakukan visualiasi data, pilih jenis visualisasi Clustered Column Chart



Gambar 168. Visualisasi Clustered Column Chart

- 2) Pada tab 'Fields' pilih kolom-kolom yang akan divisualisasikan



Gambar 169. Pilih Kolom yang akan divisualisasi

- 3) Pada tab 'Visualizations', bagian Axis, Legend dan Value di isikan sebagai berikut:

X-axis

MARKET_NAME

RICE_TYPE_NAME

YEAR

Y-axis

Average of PRICE

Legend

Add data fields here

Small multiples

Add data fields here

Tooltips

Add data fields here

Drill through

Cross-report

Keep all filters

Add drill-through fields here

Gambar 170. Tab Visualisasi

*Notes: Mengisikannya dengan cara *drag* kolom yang ada dalam tab 'Fields' ke dalam bagian yang diinginkan. Serta khusus untuk kolom 'PRICE' klik panah bawah dan pilih average

- 4) Terakhir, pada kolom filter, masukan variabel yang akan di visualisasikan, dalam hal ini pilih pasar yang akan dibandingkan di dalam tab 'MARKET_NAME', dan jenis beras yang akan divisualisasikan dalam tab 'RICE_TYPE_NAME'

Average of PRICE

is (All)

YEAR

is 2018

Filter type ①

Basic filtering

<input type="checkbox"/>	2012	366
<input type="checkbox"/>	2013	365
<input type="checkbox"/>	2014	365
<input type="checkbox"/>	2015	365
<input type="checkbox"/>	2016	366
<input type="checkbox"/>	2017	365
<input checked="" type="checkbox"/>	2018	365
<input type="checkbox"/>	2019	42

☐ Require single selection

MARKET_NAME

is Pasar Grogol or Pas...

Filter type ①

Basic filtering

Search

<input type="checkbox"/>	Pasar Cipete	1
<input type="checkbox"/>	Pasar Ciplak	1
<input type="checkbox"/>	Pasar Gondangdia	1
<input checked="" type="checkbox"/>	Pasar Grogol	1
<input checked="" type="checkbox"/>	Pasar Induk Beras Ci...	1
<input checked="" type="checkbox"/>	Pasar Jatinegara	1
<input checked="" type="checkbox"/>	Pasar Jembatan Lima	1

☐ Require single selection

RICE_TYPE_NAME

is IR-64 I

Filter type ①

Basic filtering

Search

<input checked="" type="checkbox"/>	Select all	
<input type="checkbox"/>	Cianjur Kepala (Pand...	1
<input type="checkbox"/>	Cianjur Slyp (Pandan...	1
<input type="checkbox"/>	IR-42 (Pera)	1
<input checked="" type="checkbox"/>	IR-64 I	1
<input type="checkbox"/>	IR-64 II	1
<input type="checkbox"/>	IR-64 III	1

☐ Require single selection

Gambar 171. Tambahkan data field

5) Hasil akhirnya seperti berikut

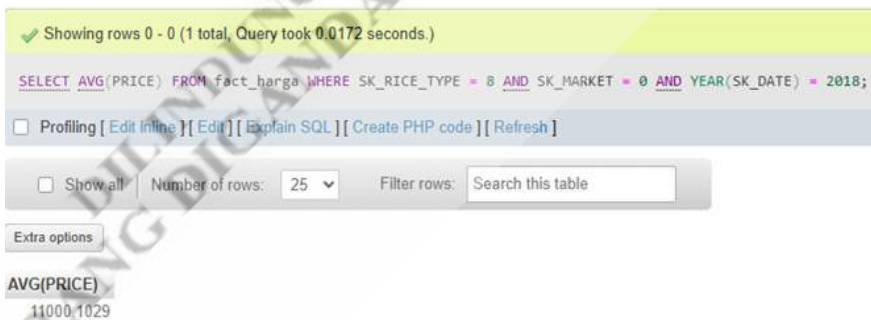


Gambar 172. Hasil Clustered Chart

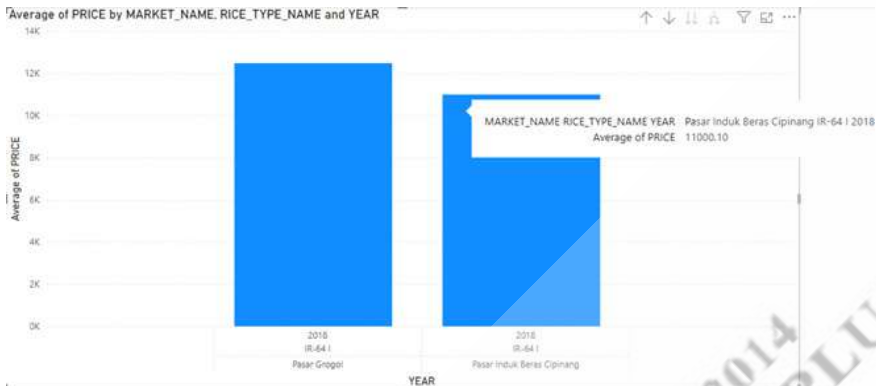
- 6) Untuk membuktikan kebenaran data, lakukan pengecekan secara manual dengan melakukan query langsung ke database

*Terlihat bahwa hasil query menunjukkan visualisasi data sudah akurat

Query=SELECT AVG(PRICE) FROM fact_harga WHERE SK_RICE_TYPE = 8 AND SK_MARKET = 0 AND YEAR(SK_DATE) = 2018;



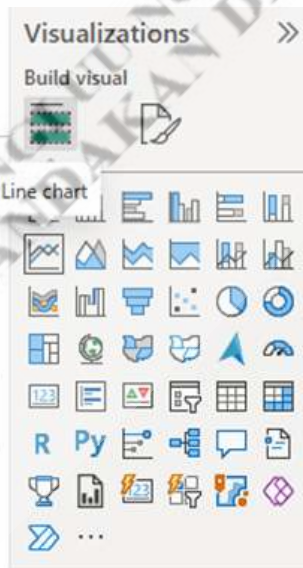
Gambar 173. Query Database



Gambar 174. Tampilan rata-rata harga di Power BI

b. Line Chart

- 1) Untuk melakukan visualiasi line chart, pilih jenis visualisasi Line Chart

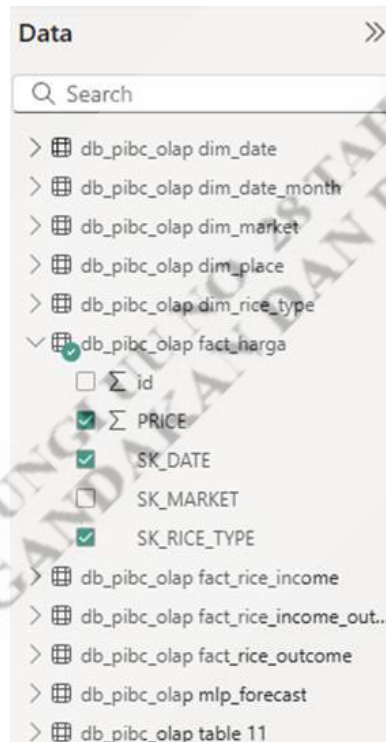


Gambar 175. Visualisasi Linechart

- 2) Untuk menampilkan line chart, pada tab 'Fields' dalam table db_pibc_olap fact_harga

- masukkan kolom 'SK_DATE' kedalam tab 'Axis' di bagian 'Visualizations'
- masukkan kolom 'SK_RICE_TYPE' ke dalam tab 'Legend' di bagian 'Visualizations'
- masukkan kolom 'PRICE' ke dalam tab 'Values'.

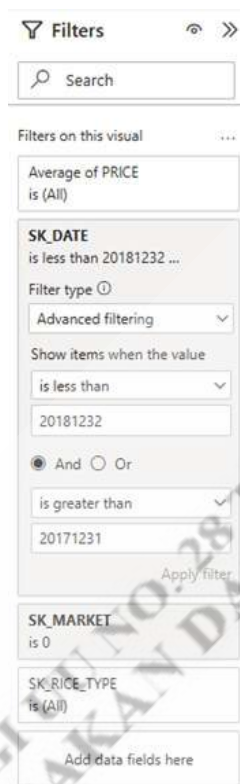
*Kolom PRICE dikonfigurasi menjadi Average of Price dengan memilih pilihan 'average' di tombol panah kebawah yang ada di tab 'Values'.



Gambar 176. Memasukkan kolom Visualisasi

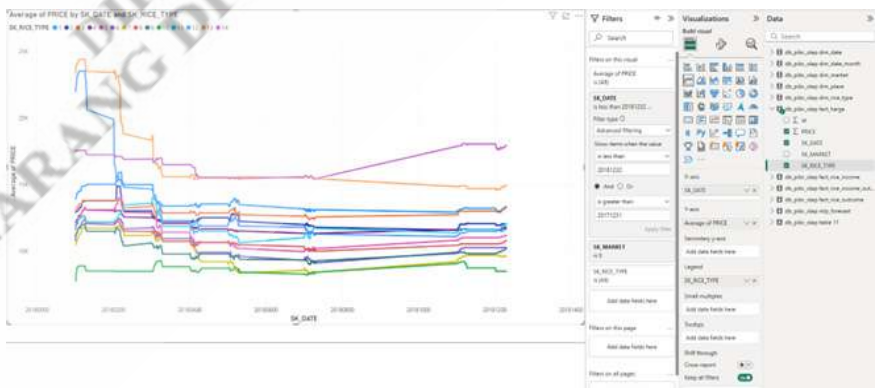
- 3) Di bagian 'Filters', tambahkan kolom 'SK_MARKET' ke bagian paling bawah dan filter hanya untuk baris yang memiliki SK_MARKET 0 saja (yang berarti pasar induk beras cipinang yang hanya dipilih). Kemudian pada kolom 'SK_DATE', gunakan advanced filtering dan tetapkan nilai 'is less than 20181232 And is greater than 20171231' untuk

mengambil grafik rentang waktu khusus di tahun 2018 saja



Gambar 177. Filter untuk Visual Line Chart

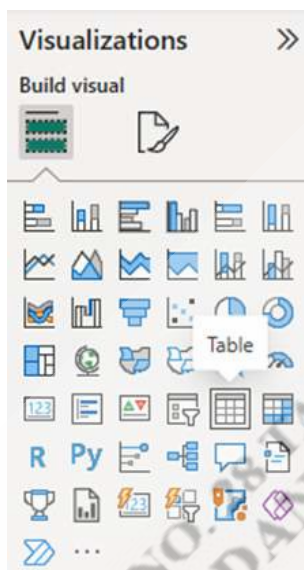
4) Hasil line chart sebagai berikut:



Gambar 178. Hasil Visual Line chart

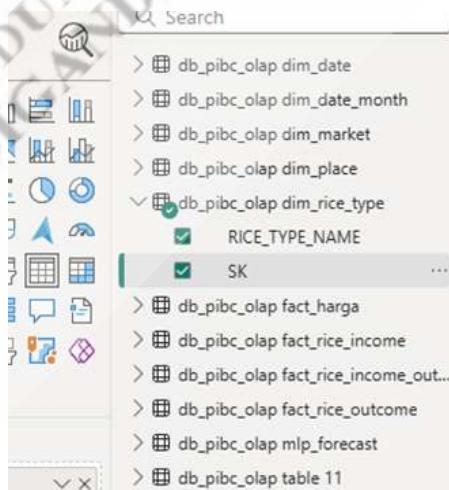
c. Table

1) Pada tab Visualizations, pilih Table



Gambar 179. Visualisasi pada Power BI

2) Pada tab 'Fields' pilih kolom-kolom yang akan divisualisasikan



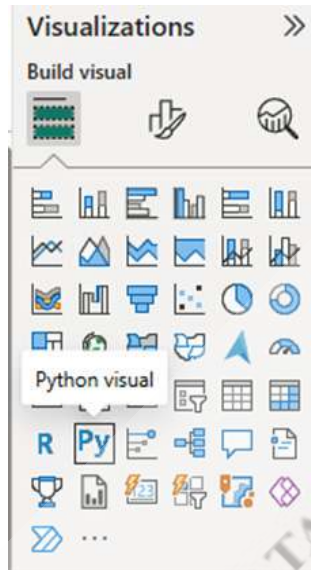
Gambar 180. Fields pada Power BI

5. Visualisasi Distribusi

- <https://www.python.org/downloads/release/python-31210/>, jika sudah terinstall buka terminal atau command prompt lalu ketik perintah berikut:

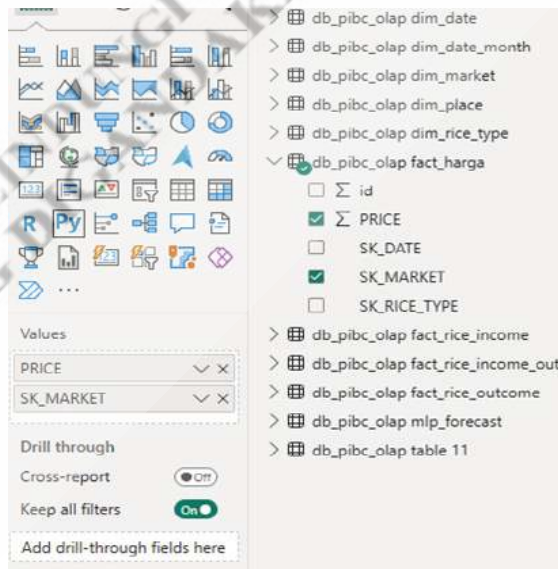
```
'pip install pandas matplotlib seaborn numpy --user'
```

- Intelijensia Bisnis dan Data Analitik | 247



Gambar 182. Membuka Python script

- c. Pindahkan tab pada bagian Fields (dengan men-drag) kedalam Values untuk dilakukan kalkulasi menggunakan Python

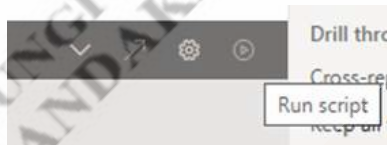


Gambar 183. Pindahkan tab pada bagian Fields

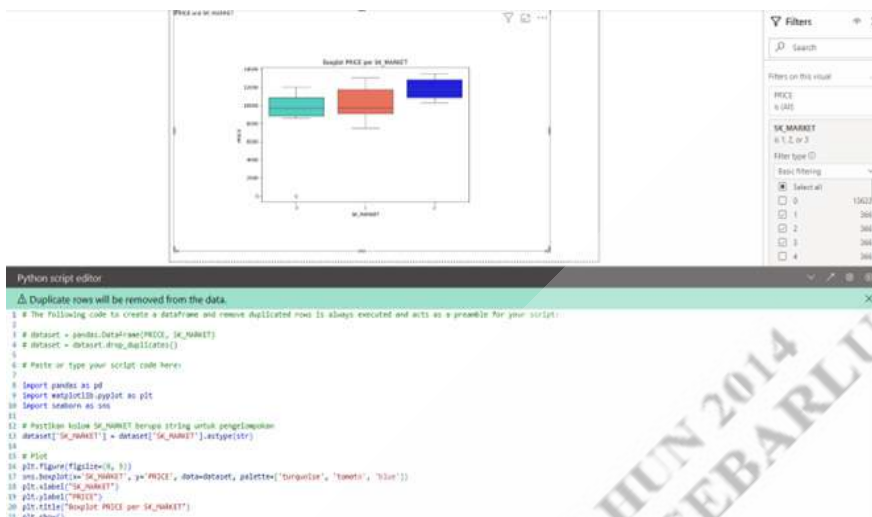
- d. Untuk menampilkan boxplot yang dapat membandingkan average price dari tiap market yang dipilih, berikut dengan pemberian warna yang berbeda tiap boxplotnya, ketikkan script sepeprti berikut:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
# Pastikan kolom SK_MARKET berupa string untuk pengelompokan
dataset['SK_MARKET'] = dataset['SK_MARKET'].astype(str)
# Plot
plt.figure(figsize=(8, 5))
sns.boxplot(x='SK_MARKET', y='PRICE', data=dataset,
palette=['turquoise', 'tomato', 'blue'])
plt.xlabel("SK_MARKET")
plt.ylabel("PRICE")
plt.title("Boxplot PRICE per SK_MARKET")
plt.show()
```

kemudian klik Run Script



Gambar 184. Tombol Run script



Gambar 185. Hasil Python script membuat boxplot

6. Visualisasi Relasi

Buka python visual baru lalu masukan data ke kolom 'Values'

Values

RICE_TYPE_NAME ✓ ✕

SK_RICE_TYPE ✓ ✕

SK_MARKET ✓ ✕

PRICE ✓ ✕

Drill through

Cross-report ☐ Off

Keep all filters ☒ On

Add drill-through fields here

db_pibc_olap dim_rice_type

☒ RICE_TYPE_NAME

☐ SK

db_pibc_olap fact_harga

☐ Σ id

☒ Σ PRICE

☐ SK_DATE

☒ SK_MARKET

☒ SK_RICE_TYPE

Gambar 186. Memasukan Data ke Values

Masukan kode python

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Filter data: hanya untuk SK_RICE_TYPE = 10
filtered = dataset[dataset['SK_RICE_TYPE'] == 10]

# Ambil PRICE untuk dua pasar
price_0 = filtered[filtered['SK_MARKET'] == 0]['PRICE'].reset_index(drop=True)
price_10 = filtered[filtered['SK_MARKET'] == 10]['PRICE'].reset_index(drop=True)

# Buat ukuran sama agar bisa diplot berpasangan
min_len = min(len(price_0), len(price_10))
price_0 = price_0[:min_len]
price_10 = price_10[:min_len]

# Plot scatter
plt.figure(figsize=(10, 6))
plt.scatter(price_0, price_10, color='green')

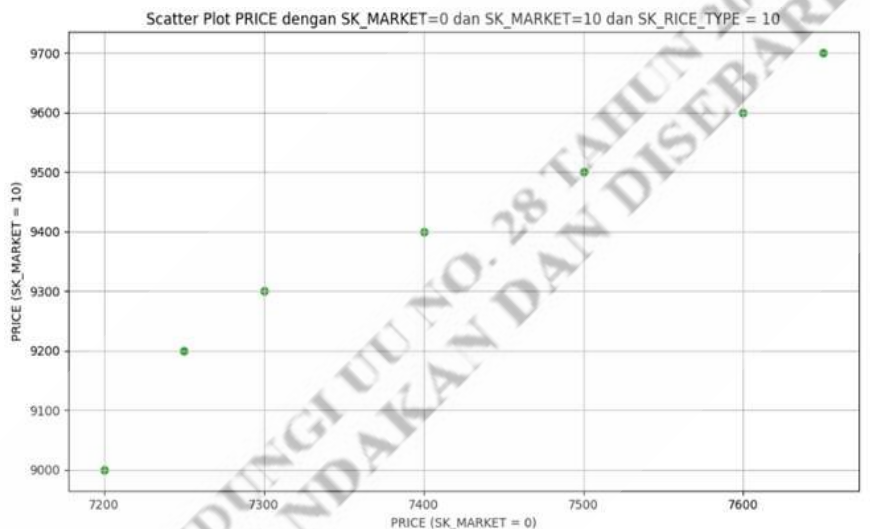
plt.title("Scatter Plot PRICE dengan SK_MARKET=0 dan SK_MARKET=10 dan SK_RICE_TYPE = 10")
plt.xlabel("PRICE (SK_MARKET = 0)")
plt.ylabel("PRICE (SK_MARKET = 10)")
plt.grid(True)
plt.tight_layout()
plt.show()
```

```

Python script editor
Duplicate rows will be removed from the data.
6 import pandas as pd
7 import matplotlib.pyplot as plt
8
9 # Filter data: hanya untuk SK_RICE_TYPE = 10
10 filtered = dataset[dataset['SK_RICE_TYPE'] == 10]
11
12 # Ambil PRICE untuk dua pasar
13 price_0 = filtered[filtered['SK_MARKET'] == 0]['PRICE'].reset_index(drop=True)
14 price_10 = filtered[filtered['SK_MARKET'] == 10]['PRICE'].reset_index(drop=True)
15
16 # Buat ukuran sumbu agar bisa diplot bersamaan
17 min_len = min(len(price_0), len(price_10))
18 price_0 = price_0[:min_len]
19 price_10 = price_10[:min_len]
20
21 # Plot scatter
22 plt.figure(figsize=(10, 6))
23 plt.scatter(price_0, price_10, color='green')
24

```

Gambar 187. Kode Visual Scatter Plot Python Script Editor di Power BI

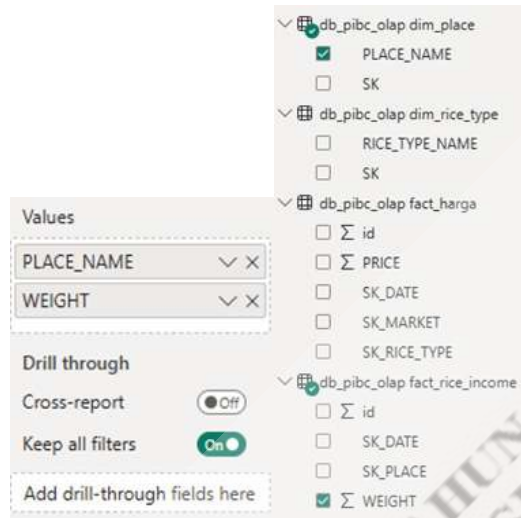


Gambar 188. Hasil Visual Scatter Plot

6. Visualisasi Komposisi

a. Pie Chart

Buka python visual baru lalu masukan data ke kolom 'Values'



Gambar 189. Memasukan Data ke Values

Masukan kode python

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Group by untuk menjumlahkan total WEIGHT per PLACE_NAME
grouped = dataset.groupby('PLACE_NAME')['WEIGHT'].sum().sort_
values(ascending=False)

# Pie chart
plt.figure(figsize=(8, 8))
plt.pie(grouped, labels=grouped.index, autopct='%1.1f%%',
startangle=140)

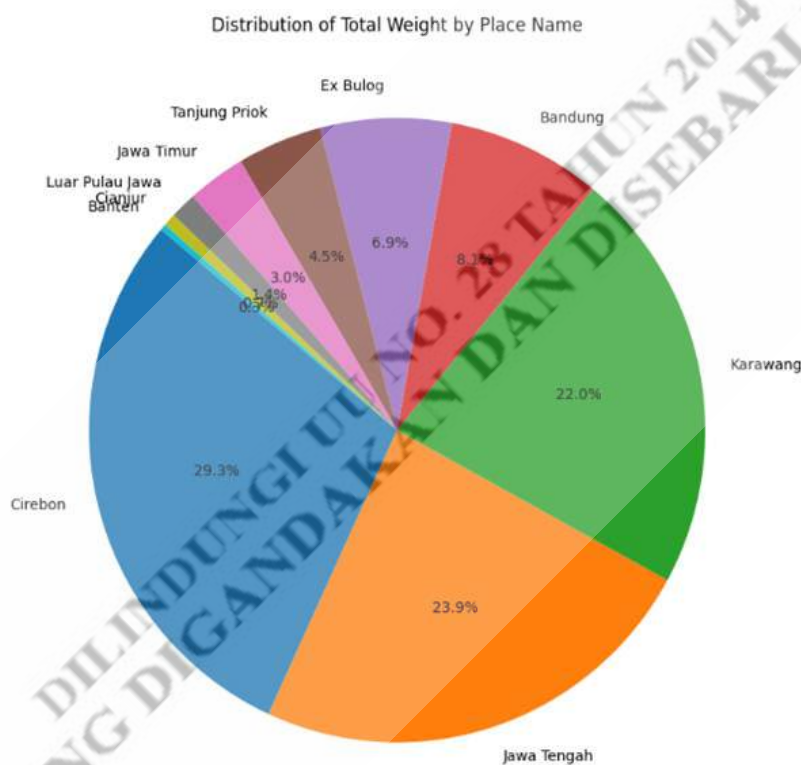
plt.title("Distribution of Total Weight by Place Name")
plt.tight_layout()
plt.show()
```

```

Python script editor
Duplicate rows will be removed from the data.
1 # The following code to create a dataframe and remove duplicated rows is always executed and acts as a preamble for your script:
2
3 # dataset = pandas.DataFrame(PLACE_NAME, HEIGHT)
4 # dataset = dataset.drop_duplicates()
5
6 # Import pandas as pd
7 # Import matplotlib.pyplot as plt
8
9 # Group by untuk menjalankan total HEIGHT per PLACE_NAME
10 grouped = dataset.groupby('PLACE_NAME')['HEIGHT'].sum().sort_values(ascending=False)
11
12 # Pie chart
13 plt.figure(figsize=(8, 8))
14 plt.pie(grouped, labels=grouped.index, autopct='%1.1f%%', startangle=140)
15
16 plt.title("Distribution of Total Weight by Place Name")
17 plt.tight_layout()
18 plt.show()

```

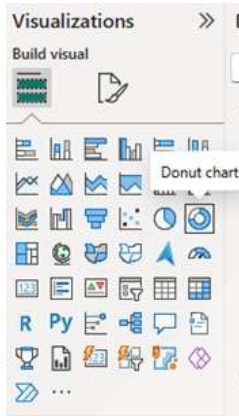
Gambar 190. Kode Visual Pie Chart Python Script Editor di Power BI



Gambar 191. Hasil Visual Pie Chart

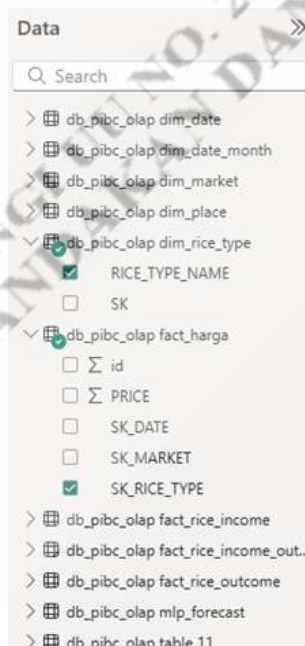
b. Donut Chart

- 1) Untuk melakukan visualiasi donut chart, pilih jenis visualisasi Donut Chart



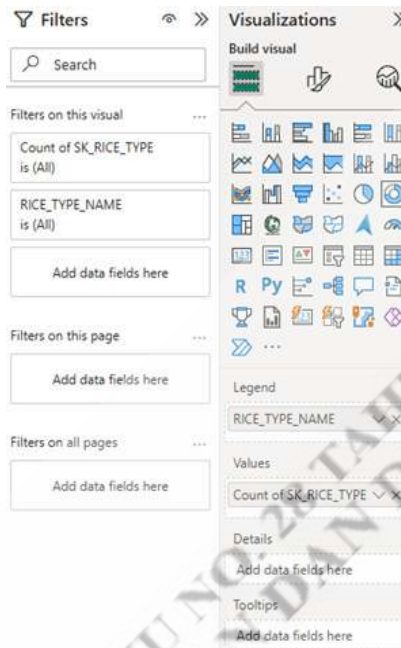
Gambar 192. Membuka Donut chart

- 2) Pada tab 'Fields' pilih kolom-kolom yang akan divisualisasikan



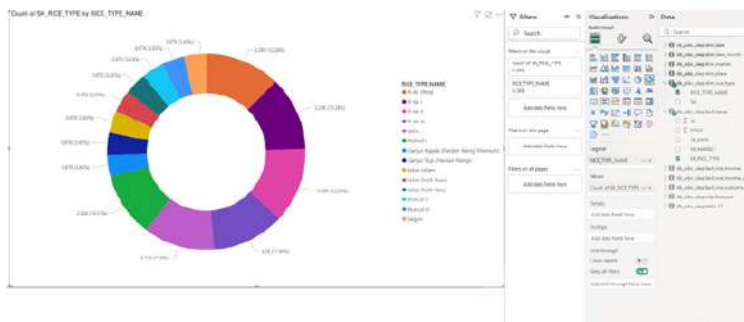
Gambar 193. Memilih tab Fields untuk bagian Value

- 3) Pada tab 'Visualizations', bagian Axis, Legend dan Value di isikan sebagai berikut:



Gambar 194. Memilih tab Fields untuk bagian Value

- a) Masukkan kolom 'RICE_TYPE_NAME' dari table db_pibc_olap dim_rice_type ke dalam tab 'Legend' di bagian 'Visualizations'.
 - b) Masukkan kolom 'SK_RICE_TYPE' dari table db_pibc_olap fact_harga ke dalam tab 'Values' di bagian 'Visualizations'. Pilih tombol panah bawah dan pilih 'count'
- 4) Hasil akhirnya seperti berikut.



Gambar 195. Hasil Visual Donut Chart

Studi Kasus

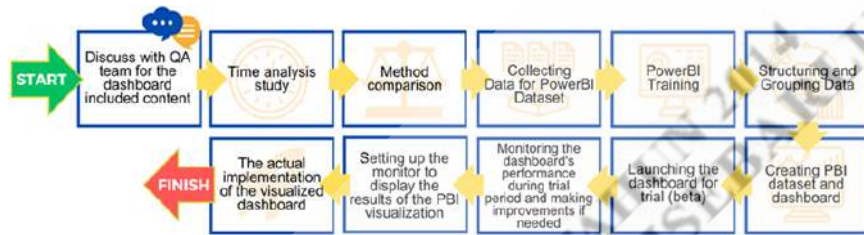
Analisis *software* yang akan digunakan untuk pembuatan dashboard untuk visualisasi data. Dianalisis tiga *interactive data visualization software* diantaranya PowerBI, Tableau, dan Sisense. Pada Tabel didapatkan hasil *advantage* dan juga *disadvantage* dari masing-masing *software*.

Tabel 31. Perbandingan Metode

Software	PowerBI	Tableau	Sisense
(+)	(1) Integration with Microsoft Ecosystem (2) User-Friendly Interface	(1) Wide variety of Visualization options.	(1) In-Chip Technology = Faster data loading
(-)	(1) Limited Custom Visuals	(1) Expensive Cost (2) Complexity for Beginners	(1) Expensive Cost (2) Not integrated with microsoft
Judgement			

Berdasarkan analisis di atas diputuskan untuk menggunakan PowerBI sebagai *countermeasure* dikarenakan PowerBI terintegrasi dengan Microsoft 360 yang merupakan *based platform* yang digunakan oleh perusahaan sehingga nantinya akan lebih memudahkan saat integrasi data.

Implementasi dari improvement yang dibuat terlihat pada beberapa aktivitas berikut ada pada Gambar 196.



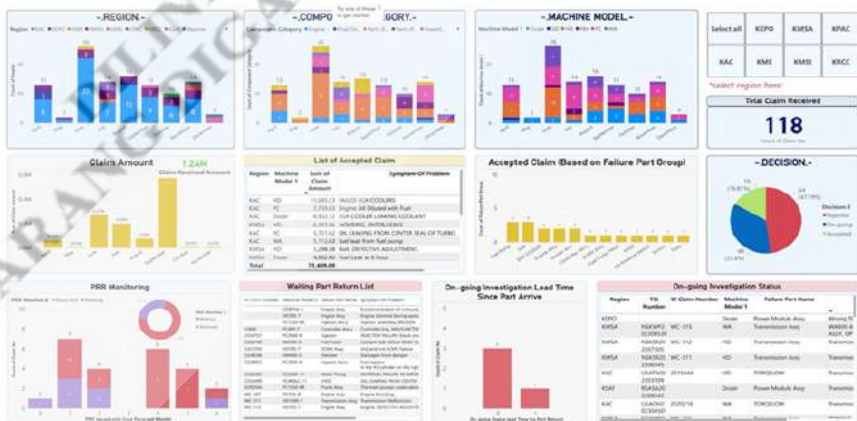
Gambar 196. Improvement Implementation

Dimulai dengan melakukan diskusi dengan Team Quality Assurance untuk konten yang akan dimasukkan pada dashboard PowerBI. Dilanjutkan dengan melakukan time study analysis untuk mengetahui hasil sebelum dan setelah implementasi improvement yang ada. Lalu dilakukan method comparison yaitu pencarian software yang akan digunakan untuk pembuatan dashboard untuk visualisasi data. Selanjutnya, mengumpulkan data untuk pembangunan dataset PowerBI. Selanjutnya, diadakan PowerBI training untuk melatih kemampuan PIC dashboard. Langkah selanjutnya yaitu mengelompokkan data untuk database PowerBI sebelum pembuatan PowerBI dataset dan dashboard. Setelah dashboard terbentuk, akan dilakukan launching untuk fase trial terlebih dahulu. Selanjutnya, melakukan monitoring performa dashboard dan melakukan perbaikan bila diperlukan. Setelah dashboard sudah dapat digun dianggap optimal maka dashboard akan ditampilkan pada monitor di genba (area produksi) agar operator dan team produksi lainnya juga dapat mengetahui informasi mengenai claim warranty yang ada.



Gambar 197. Display PowerBI di Genba

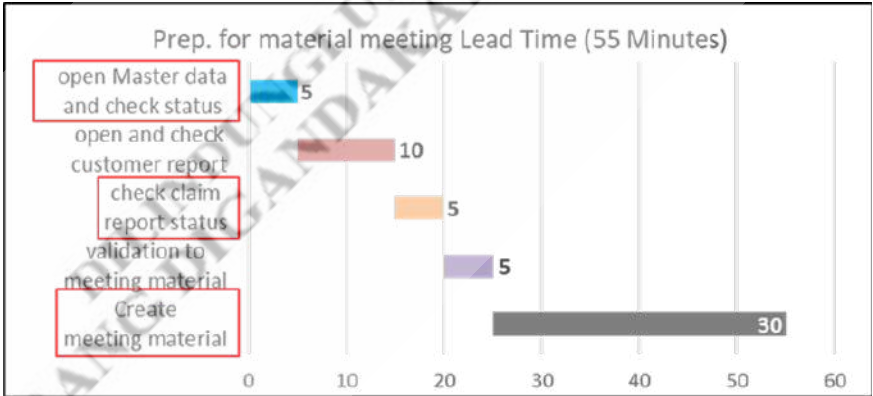
Untuk tahap selanjutnya yaitu tahap keenam Verifying the Effects. Dimulai dari indentifikasi result of improvement. Pada Gambar 198 ditampilkan hasil dashboard PowerBI yang telah dibuat.



Gambar 198. Tampilan Power BI

Melalui konten Power BI yang sudah tervisualisasi dan terintegrasi nantinya Team Quality Assurance tidak perlu membuka master data satu persatu lagi. Mempercepat pengecekan claim status report dengan hanya klik masing-masing konten PowerBI yang tertampil, nantinya PowerBI akan secara langsung memberikan info yang diinginkan user. Terakhir, pastinya akan mempercepat pembuatan material meeting karena yang sebelumnya Team QA KI KBN harus mengolah data Microsoft Excel, melakukan pivot data, dan membuat grafik secara satu persatu. Dengan adanya dashboard PowerBI nantinya material meeting bisa disusun pada Microsoft PowerPoint hanya dengan melampirkan hasil screenshot grafik-grafik dari PowerBI saja.

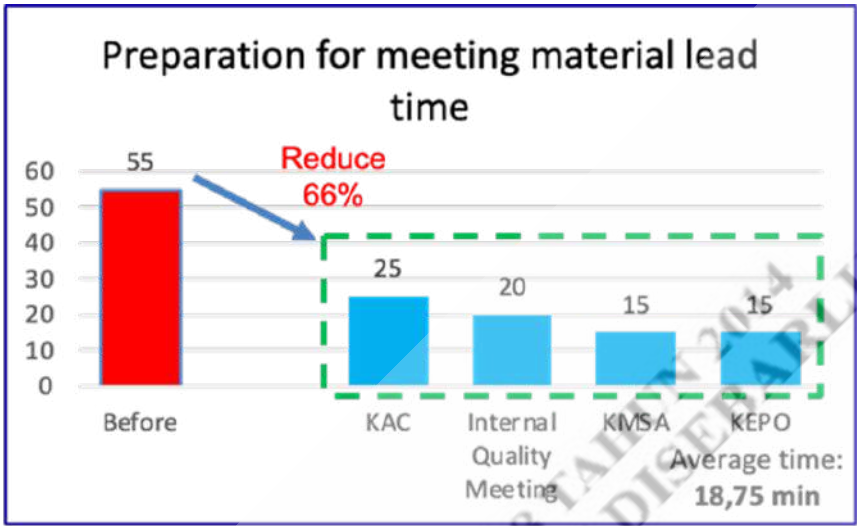
Tahap selanjutnya memverifikasi effects dari improvement yang telah dilakukan. Pada Gambar 156 terlihat grafik menunjukkan untuk aktivitas yang tereliminasi adalah aktivitas open master data and check status dan juga aktivitas check claim report status. Sementara, aktivitas yang ter-reduce waktunya yaitu aktivitas create meeting material di mana yang sebelumnya tiga menit menjadi dapat dilakukan hanya selama 10 menit.



Gambar 199. Verifying the Effects

Berdasarkan time study yang dilakukan, ditampilkan pada Gambar 200 setelah implementasi improvement yang di mana sebelumnya waktu untuk aktivitas preparation for meeting material lead time yaitu selama 55 menit, menjadi 18,75 menit secara rata-rata untuk beberapa creating meeting materia. Sehingga, lead time

aktivitas preparation for meeting material te-reduce sebanyak 66%.






Gambar 200. Before after improvement

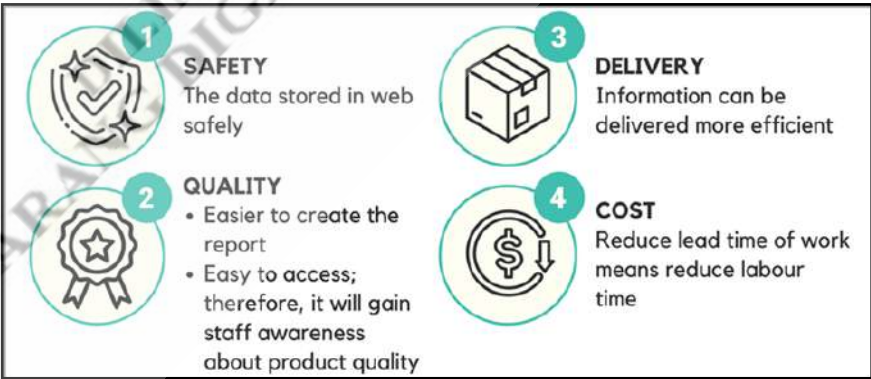
Untuk tahap selanjutnya yaitu tahap ketujuh Standardization and the Establishment of Fix Management Procedure. Pada Tabel 32 terlihat penerapan perbaikan dengan menampilkan dashboard visual di genba (area produksi). Team Quality Assurance sedang melakukan briefing kepada operator dan team produksi lainnya agar mereka mengetahui informasi mengenai claim warranty yang ada. Standarisasi yang diterapkan untuk dashboard claim warranty adalah Team Quality Assurance harus memperbarui database pada setiap awal bulan untuk menjaga data yang divisualisasikan tetap terkini dan dapat diakses oleh semua orang.

Terakhir, tahap kedelapan yaitu Remaining Problems and Future Plan adalah menunjukkan rencana perbaikan kedepannya untuk menunjang improvement berupa dashboard yang sudah dibangun. Terlihat pada Tabel beberapa plans, goals, dan status dari aktivitas improvement yang dilakukan Team QA Plant X.

Tabel 32. Persoalan saat ini dan rencana di masa depan

Rencana	1	2	3
	Membuat PowerBI dashboard	Melaksanakan pelatihan PowerBI untuk karyawan	<i>socialize about the update on warranty claims in genba</i>
Goals	<i>to integrate KI KBN warranty database</i>	<i>to standardize the skills of the powerBI PIC</i>	<i>to make all staff (outside QA Team) knows the update of KI KBN warranties claim</i>
Status			

Pada Gambar 201 ditampilkan benefit improvement yang telah dilakukan dari segi SQDC (safety, quality, delivery, dan cost)



Gambar 201. Keuntungan dalam SQDC (Safety, Quality, Delivery, Cost)

Pertama, keuntungan dari segi safety (keamanan) yaitu data disimpan di web dengan aman. Selanjutnya, dari segi quality (kualitas) yaitu lebih mudah untuk membuat laporan dan juga mudah diakses; oleh karena itu, hal ini akan meningkatkan kesadaran staf tentang kualitas produk. Dari segi delivery (penyampaian) informasi mengenai claim warranty dapat disampaikan dengan lebih efisien. Terakhir, dari segi cost (biaya) yaitu mengurangi lead time kerja berarti mengurangi waktu kerja.

D. Studi Kasus

Dalam hal penyajian data untuk performansi perusahaan, dilakukan meeting dengan manajerial untuk mengetahui *voice of customer* dari *dashboard* yang akan dibuat, berikut merupakan bagian yang perlu dicantumkan untuk melihat performa perusahaan dari sisi biaya:

1. Nama Departemen.
2. Work Cost Center ID.
3. Biaya secara Rupiah.
4. Biaya secara Rupiah/Unit masing masing tipe.
5. Tipe.
6. Progres Rupiah dan Rupiah/Unit setiap bulannya.
7. Persentase penggunaan Biaya untuk masing-masing tipe.
8. Penggunaan biaya masing-masing tipe terhadap item process cost yang ada.



Gambar 202. Dashboard Cost Power BI

Dari setiap item cost yang ada yaitu *salary*, *overtime*, *material consumable* (*fuel lub*, *indirect amterial*, *toll equipment*, *cutting tool*, *consumable*), *spoilage* dan *utility*, khusus untuk *material consumable* dibuatkan dashboard untuk mampu mengevaluasi penggunaan material yang ada apakah sesuai dengan target atau tidak, dan juga dapat melihat pola dari pengambilan yang ada, dengan kalsifikasi nama *controllable item*, nomor material, desripsi, penggunaan secara rupiah, sehingga ditampilkan dengan urutan sebagai berikut:

1. Nama Departemen.
2. Work Cost Center ID.
3. Target Biaya secara Rupiah.
4. Aktual Biaya secara Rupiah.
5. Biaya secara Rupiah/Unit masing masing tipe.
6. Progres Rupiah dan Rupiah/Unit setiap bulannya.
7. Trend pola pengambilan material.
8. Persentase penggunaan Biaya untuk masing-masing tipe.



Gambar 203. Cost Material Consumable

Untuk mengetahui performa dari suatu tipe, dilihat berdasarkan harga Rp/Unit, performa kecacatan setiap bulannya, dan nilai *Overall Equipment Effectiveness* (OEE) dari seksi LPDC. *Dashboard* ini bertujuan untuk mengetahui apakah terdapat pengaruh antara Peningkatan Performa nilai OEE, penurunan kecacatan dan penurunan nilai Rp/Unit dari proses produksi *cylinder head*. Berikut urutan tampilan dashboard Power BI yang digunakan sesuai dengan suara pelanggan (*voice of customer*) dari penelitian ini yaitu manajerial.

1. Supporting type ke plant.

Dalam hal penyajian data untuk performansi perusahaan, dilakukan meeting dengan manajerial untuk mengetahui *voice of customer* dari *dashboard* yang akan dibuat, berikut merupakan bagian yang perlu dicantumkan untuk melihat performa perusahaan dari sisi biaya:

2. Nama Departemen.
3. Work Cost Center ID.
4. Biaya secara Rupiah.
5. Biaya secara Rupiah/Unit masing masing tipe.

6. Tipe.
7. Progres Rupiah dan Rupiah/Unit setiap bulannya.
8. Persentase penggunaan Biaya untuk masing-masing tipe.
9. Penggunaan biaya masing-masing tipe terhadap item process cost yang ada.



Gambar 204 Dashboard Cost Power BI

Dari setiap item cost yang ada yaitu *salary*, *overtime*, *material consumable* (*fuel lub*, *indirect amterial*, *toll equipment*, *cutting tool*, *consumable*), *spoilage* dan *utility*, khusus untuk *material consumable* dibuatkan dashboard untuk mampu mengevaluasi penggunaan material yang ada apakah sesuai dengan target atau tidak, dan juga dapat melihat pola dari pengambilan yang ada, dengan kalsifikasi nama *controllable item*, nomor material, desripsi, penggunaan secara rupiah, sehingga ditampilkan dengan urutan sebagai berikut:

1. Nama Departemen.
2. Work Cost Center ID.
3. Target Biaya secara Rupiah.
4. Aktual Biaya secara Rupiah.

5. Biaya secara Rupiah/Unit masing masing tipe.
6. Progres Rupiah dan Rupiah/Unit setiap bulannya.
7. Trend pola pengambilan material.
8. Persentase penggunaan Biaya untuk masing-masing tipe.



Gambar 205 Cost Material Consumable

Untuk mengetahui performa dari suatu tipe, dilihat berdasarkan harga Rp/Unit, performa kecacatan setiap bulannya, dan nilai *Overall Equipment Effectiveness* (OEE) dari seksi LPDC. *Dashboard* ini bertujuan untuk mengetahui apakah terdapat pengaruh antara Peningkatan Performa nilai OEE, penurunan kecacatan dan penurunan nilai Rp/Unit dari proses produksi *cylinder head*. Berikut urutan tampilan dashboard Power BI yang digunakan sesuai dengan suara pelanggan (*voice of customer*) dari penelitian ini yaitu manajerial.

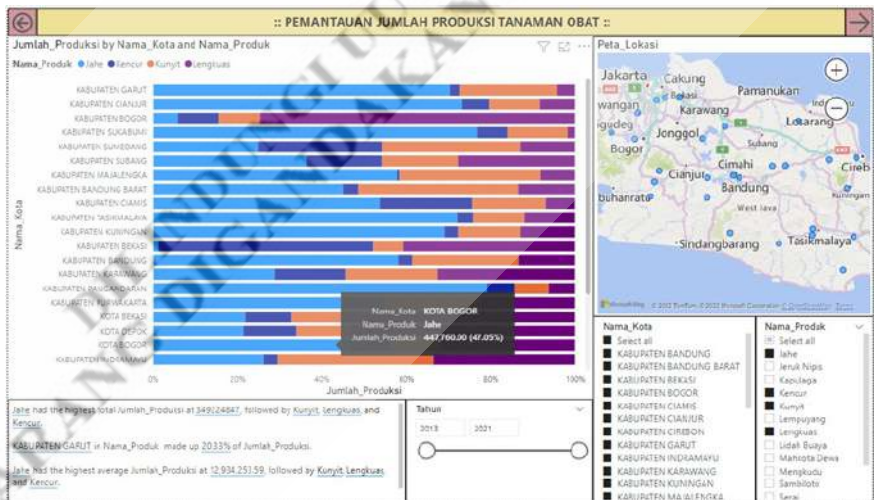
Contoh dashboard dirancang sesuai kebutuhan dan data pada data warehouse dirancang dengan dengan Power BI. (Hidayat M.K dan Fitriana,R, 2022).

Informasi yang disajikan pada *dashboard* terdiri dari beberapa bagian yaitu informasi jumlah produksi berdasarkan nama kota

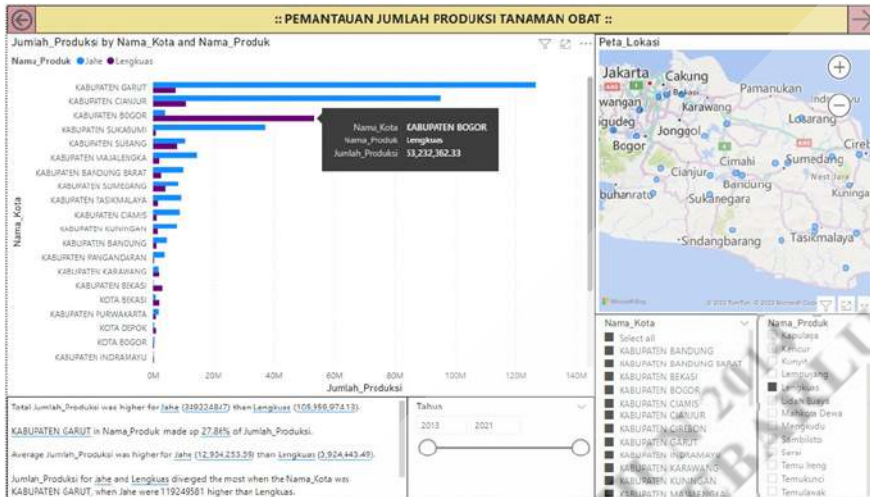
dan nama produk dalam bentuk *stacked bar chart*; informasi jumlah produksi berdasarkan nama kota dan nama produk dalam bentuk *clustered bar chart*; informasi jumlah produksi berdasarkan tahun dan nama produk dalam bentuk *donut chart*; informasi jumlah produksi berdasarkan tahun dan nama kota dalam bentuk *line chart*; informasi hasil proses pengelompokan wilayah produksi dengan menerapkan algoritma *K-Means* dalam bentuk *treemap* dan *line chart*.

Selain menggunakan grafik, tampilan *dashboard* dilengkapi ikhtisar informasi, peta wilayah, fitur filter, dan disertai informasi yang rinci dalam bentuk tabel. Alasan penggunaan grafik dipilih karena menampilkan data dalam bentuk grafik, lebih mudah bagi otak untuk memahami informasi dan memprosesnya.

Tampilan *Stacked Bar Chart* pada Gambar 206 merupakan grafik dengan menggunakan batang untuk menggambarkan perbandingan antar kategori data, dengan membagi serta membandingkan seluruh bagian. Batang-batang pada grafik mewakili wilayah produksi, dan segmen pada batang mewakili jenis produk tanaman obat.



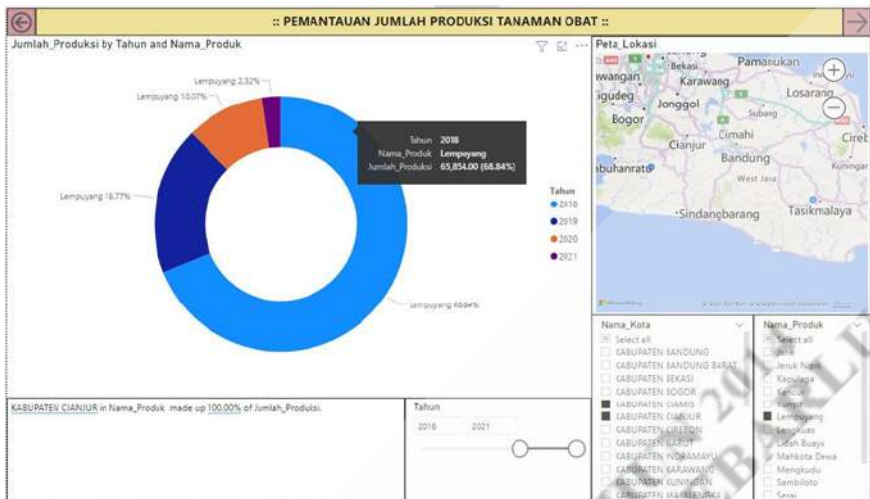
Gambar 206. *Stacked Bar Chart Dashboard*



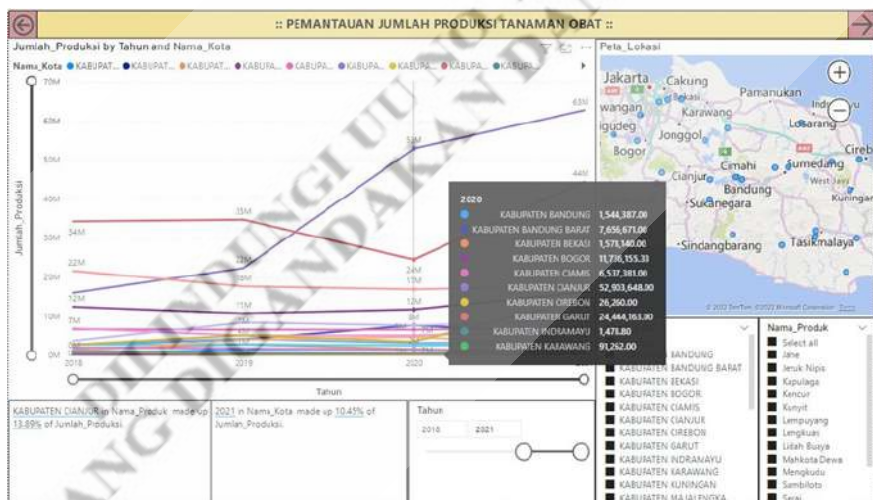
Gambar 207. Clustered Bar Chart Dashboard

Tampilan Clustered Bar Chart pada Gambar 207 merupakan grafik dengan menggunakan batang secara berkelompok yang berbeda ditempatkan berdampingan untuk membandingkan nilai pada semua kategori dengan menggunakan batang horizontal atau vertikal. Setiap kelompok batang pada grafik mewakili wilayah produksi, dan setiap batang pada kelompok mewakili jenis produk tanaman obat. Tampilan Donut Bar Chart pada Gambar 208 merupakan grafik yang digunakan untuk menggambarkan perbandingan data berdasarkan kategori dengan ukuran setiap bagian merupakan representasi proporsi masing-masing kategori. *Donut Bar Chart* digunakan untuk menunjukkan persentase jumlah produksi berdasarkan tahun dan jenis produk tanaman obat.

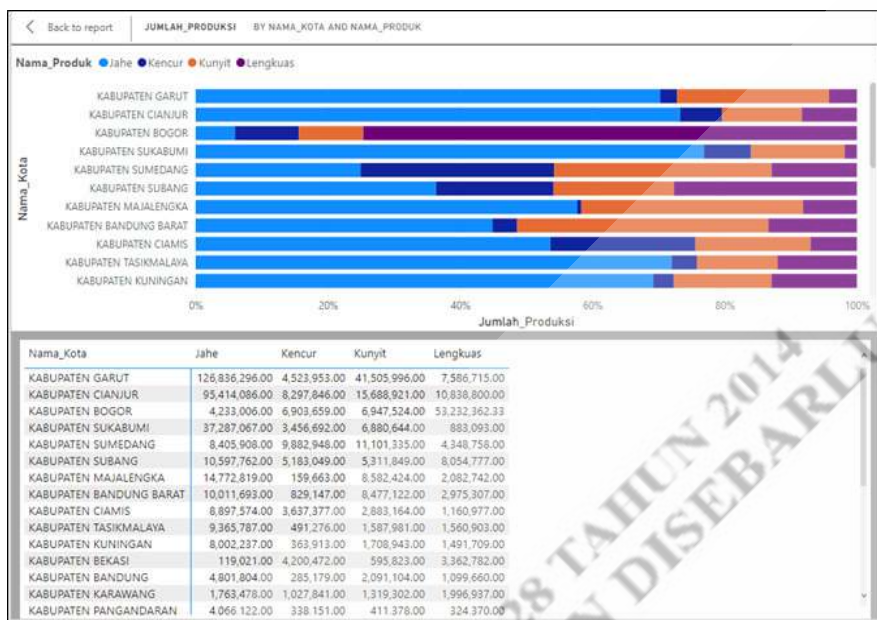
Tampilan Line Chart pada Gambar 209 merupakan grafik yang menggunakan garis untuk menyambungkan titik-titik data yang bernilai kuantitatif dalam rentang waktu tertentu sebagai representasi dari data historis. *Line Chart* digunakan untuk menampilkan jumlah produksi berdasarkan tahun dan wilayah produksi tanaman obat.



Gambar 208. Donut Chart Dashboard



Gambar 209. Line Chart Dashboard



Gambar 210. Table Dashboard



Gambar 211. Clustering Dashboard

Tampilan tabel pada Gambar 210 digunakan untuk menyajikan informasi dengan tampilan yang lebih rinci dalam penelusuran

data. Hal ini memungkinkan untuk menggali informasi lebih dalam secara interaktif sesuai dengan kebutuhan pengguna yang belum tersajikan dalam grafik atau visualisasi data yang lain. *Means* pada proses *data mining* dapat diakses melalui *Clustering Dashboard*. Informasi yang ditampilkan pada *dashboard* adalah *treemap* dan tabel yang menunjukkan jumlah *cluster* dan anggotanya, serta *line chart* dan tabel yang menunjukkan nilai *centroid* setiap jenis produk di masing-masing *cluster*. Tampilan *Clustering Dashboard* dapat dilihat pada Gambar 211.

Kebutuhan informasi terhadap dashboard yang dirancang adalah untuk mampu menampilkan grafik deret waktu (*time series plot*) harga beras berdasarkan 14 jenis beras yang umum diperdagangkan di PIBC (seperti Cianjur Kepala, Cianjur Slop, Setra, Saigon, Muncul I, Muncul II, Muncul III, IR-64 I, IR-64 II, IR-64 III, IR-42, ketan putih biasa, ketan putih paris, dan ketan hitam), jenis pasar serta rentang waktu penyajian grafik. Selain itu, dashboard juga dapat menampilkan grafik batang perbandingan rerata harga dari 14 jenis beras berdasarkan pasar retail atau eceran yang sudah dipilih sebelumnya. Grafik batang yang dihasilkan tidak dibatasi jumlah pasar retail serta jenis beras yang akan diperiksa. Selain daripada itu, dashboard juga diharapkan dapat menampilkan diagram lingkaran jumlah pasokan beras masuk dan keluar PIBC yang dapat dipilih berdasarkan kota asal (beras masuk) dan kota tujuan (beras keluar) serta dapat menampilkan informasi jumlah stok beras di PIBC.

Tabel 33. Sampel data harga dari PT. Food Station dalam format excel

Tgl	Cianjur Kepala (Pandan)	Cianjur Slop (Pandan)	Setra	Saigon	Muncul I	Muncul II	Muncul III	IR-64 I	IR-64 II	IR-64 III	IR-42 (Pera)	Ketan Putih Biasa	Ketan Putih Paris	Ketan Hitam
1	13,475	12,425	13,250	11,825	11,075	10,450	9,875	10,450	9,750	8,650	10,825	13,025	18,725	19,300
2	13,450	12,400	13,250	11,825	11,075	10,425	9,875	10,450	9,750	8,650	10,850	13,025	18,750	19,300
3	13,450	12,400	13,250	11,825	11,075	10,425	9,875	10,450	9,725	8,650	10,850	13,025	18,750	19,300
4	13,425	12,400	13,250	11,825	11,075	10,425	9,875	10,450	9,725	8,650	10,850	13,000	18,750	19,300
5	13,425	12,400	13,250	11,825	11,075	10,425	9,875	10,450	9,725	8,650	10,850	13,000	18,775	19,300

Kegiatan bisnis utama perusahaan terkait dengan posisi perusahaan sebagai pengelola tunggal PIBC yang harus memastikan pasokan beras aman dimana stok beras dijaga tetap di atas 30.000

ton setiap harinya. Secara umum jasa utama yang diselenggarakan perusahaan adalah jasa pertokoan, jasa pergudangan, jasa pengangkutan, resi gudang serta penyediaan informasi perdagangan beras berupa informasi harga dan stok serta bisnis beras yang meliputi pengadaan, pengolahan beras modern serta penjualan beras dengan merek FS. Ada pula bisnis perdagangan lainnya yang dijalankan yaitu penjualan minyak goreng dan gula tebu yang masih dibawah merek FS.

Pemilihan *grain* terkait dengan penentuan yang akan dipresentasikan oleh sebuah tabel fakta atau oleh *record* di dalam tabel fakta. Grain yang digunakan untuk merancang data warehouse adalah harga, kuantitas beras masuk, kuantitas beras keluar dan stok beras.

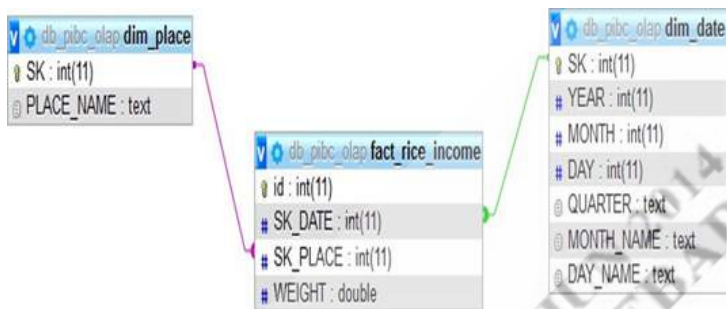
Beragam jenis data yang ada dari beberapa dimensi dapat direlasikan kepada beberapa data *grain* yang sudah didapatkan. Hal ini digunakan untuk membangun struktur *database* yang ternormalisasi. Hasil penghubungan relasi tersebut disimpan kedalam tabel baru didalam database bernama *fact table* atau tabel fakta. Hubungan antara tabel fakta dan tabel dimensi dapat dilihat pada skema bintang pada Gambar 212 dan Gambar 213 di bawah ini.



Gambar 212. Diagram bintang fact_harga

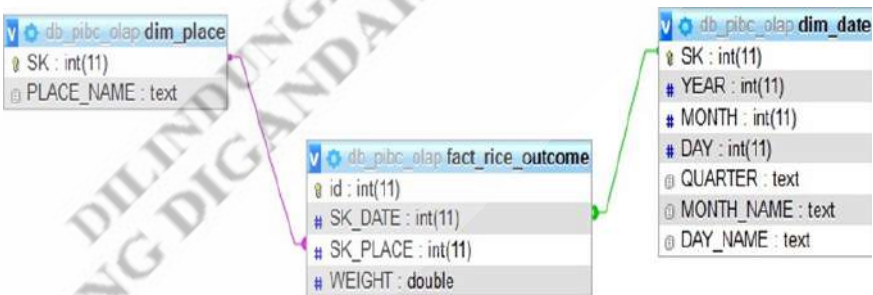
Untuk menyimpan data harga beragam jenis beras berdasarkan

tipe berasnya, tanggalnya, serta lokasi pasarnya maka dapat dilakukan relasi antara tabel `dim_date`, `dim_rice_type` dan `dim_market` untuk membentuk satu tabel baru bernama `fact_harga` (Gambar 3).



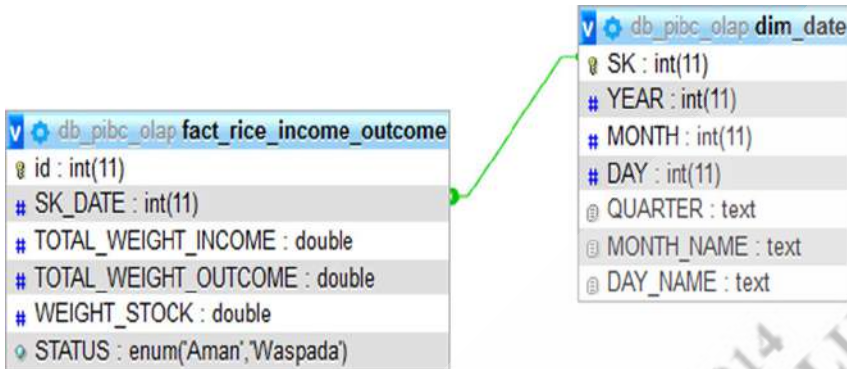
Gambar 213. Diagram bintang `fact_rice_income`

Kemudian untuk menyimpan informasi mengenai stok beras yang masuk berdasarkan tanggalnya dapat dilakukan relasi antara tabel `dim_date` dengan `dim_location` untuk membentuk tabel baru bernama `fact_rice_income` yang dimasuki oleh data grain berupa kuantitas beras masuk (Gambar 214).



Gambar 214. Diagram bintang `fact_rice_outcome`

Selanjutnya, penyimpanan informasi untuk jumlah stok beras yang keluar juga dapat dilakukan. Sama dengan proses pada Gambar 214 yang membedakan hanyalah data grain kuantitas beras keluar yang dimasukan kedalam tabel `fact_rice_outcome` (Gambar 215).



Gambar 215. Diagram bintang fact_rice_income_outcome

Terakhir, untuk melihat kuantitas total dari stok beras yang tersedia (tanpa melihat tempat asal dan tujuan dari beras tersebut berpindah) maka tabel fact_rice_income_outcome dapat dibentuk dengan merelasikan tabel dim_date dan menambahkan grain kuantitas beras masuk, kuantitas beras keluar serta tambahan kolom yang menandakan status kecukupan dari stok beras yang ada (Gambar 216).

Setelah struktur basis data dihasilkan maka proses akuisi bisa dijalankan. Data yang terdapat pada berbagai sheet di file excel PT. Food Station serta data dari info pangan Jakarta diintegrasikan menjadi sebuah database OLAP yang dinamakan *db_pibc_olap*. Proses ETL dalam penelitian ini bermakna migrasi data dari data harga dan pasokan beras dari PIBC dengan format Excel yang berasal dari PT. Food Station dan data berformat JSON dari situs <https://infopangan.jakarta.go.id/> menuju data *warehouse* dimana proses ini akan dilakukan setiap bulan. Durasi waktu dari data yang dimasukkan ke dalam data warehouse adalah data 4 tahun terakhir mulai 2016- 2019. Proses pemindahan data baru ke dalam data warehouse dilakukan setiap awal bulan.

Proses selanjutnya setelah data tersimpan di data warehouse adalah memanggil database tersebut di Google Data Studio serta melakukan kustomisasi sumber data seperti dapat dilihat pada Gambar 217.

Penambahan kolom yang diatur secara kustom seperti kolom SK_RICE_TYPE_string berguna untuk mengintegrasikan atribut

yang sama dari satu tabel dengan tabel lainnya, khususnya atribut jenis beras.

Salah satu contoh tampilan dashboard grafik batang harga rerata beras sesuai dengan pasar yang dipilih, jenis beras dan tanggalnya dapat dilihat pada Gambar 8 (sisi kiri). Pengguna dapat memilih berdasarkan variabel yang akan ditampilkan dan secara interaktif grafik batang akan terbentuk sesuai pilihan parameter yang diberikan. Terlihat pula pada gambar tersebut informasi harga eceran beras IR64- III di Pasar Jembatan Merah bernilai sama dengan data yang bersumber dari situs info pangan Jakarta tertanggal 10 Januari 2019. Hal ini menunjukkan dashboard yang dihasil dari penelitian ini telah mampu mengintegrasikan data harga di tingkat grosir yang berasal dari PT. Food Station (PIBC) serta harga di tingkat eceran yang berasal dari situs info pangan Jakarta.

← EDIT CONNECTION | FILTER BY EMAIL

Field ↓	Type ↓	Default Aggregation ↓	Description ↓
DIMENSIONS (6)			
id	123 Number	Sum	
PRICE	123 Number	Sum	
SK_DATE	Date	None	
SK_MARKET	123 Number	Sum	
SK_RICE_TYPE	123 Number	Sum	
SK_RICE_TYPE_string	ABC Text	Sum	
METRICS (1)			
Record Count	123 Number	Auto	

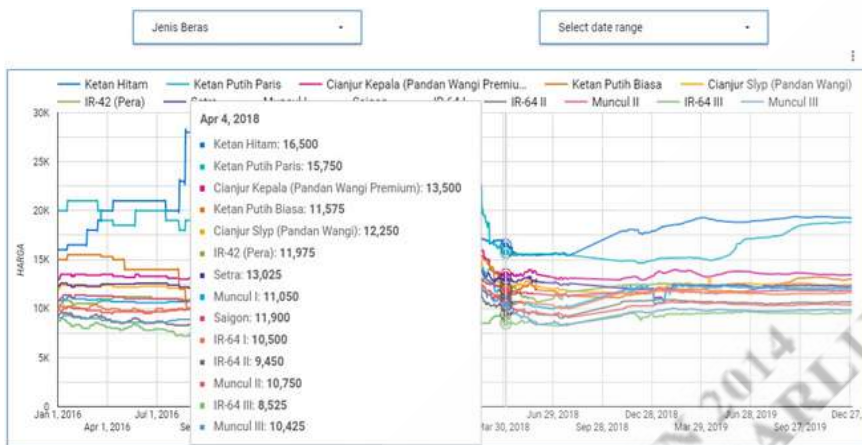
Gambar 216. Kustomisasi sumber data di Google Data Studio



Gambar 217. Perbandingan tampilan grafik batang di Google Data Studio (sisi kiri) dan sumber data (sisi kanan)

Kemudian untuk tampilan dashboard grafik deret waktu harga beras terlihat pada Gambar 219. Pengguna dapat memilih tipe beras yang akan ditampilkan dan secara interaktif grafik garis akan bersisi plot deret waktu perkembangan harga sesuai tipe berasnya. Pengguna juga dapat mengatur rentang waktu yang ingin ditampilkan pada grafik timeseries ini sesuai dengan pilihan yang sudah ditentukan pengguna. Data yang ditampilkan pada Gambar untuk data tanggal 4 April 2018 dapat ditunjukkan bernilai sama dengan data yang dipublikasi oleh PT. Food Station (Gambar 2020).

Tampilan grafik harga dengan pilihan jenis beras dan rentang waktu ini juga dapat digunakan untuk memonitor dampak dari pelaksanaan operasi pasar yang dilakukan oleh PT. Food Station bersama instansi terkait. Sebagai contoh operasi pasar beras medium yang pernah dilakukan adalah pada tanggal 22 November 2018 ketika harga mulai merangkak naik sebelum dilakukannya operasi pasar dan terlihat menurun setelahnya (Gambar 2021).



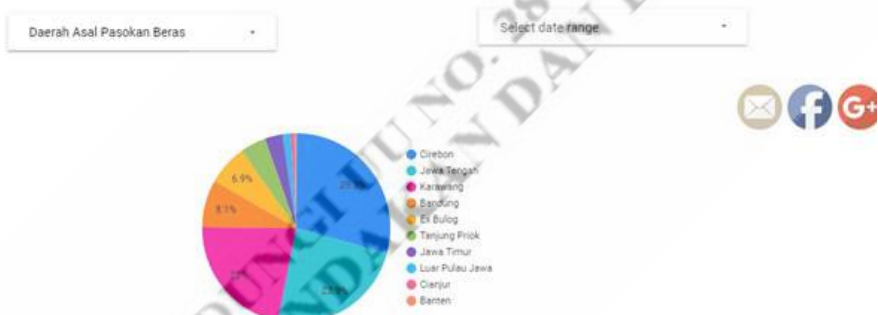
Gambar 218. Tampilan grafik deret waktu di Google Data Studio



Gambar 219. Data harga beras tanggal 4 April 2018 yang bersumber dari PT. Food Station



Gambar 220. Perbandingan tren harga sebelum dan sesudah operasi pasar beras medium tanggal 22 November 2018



Gambar 221. Diagram lingkaran daerah asal pasokan beras

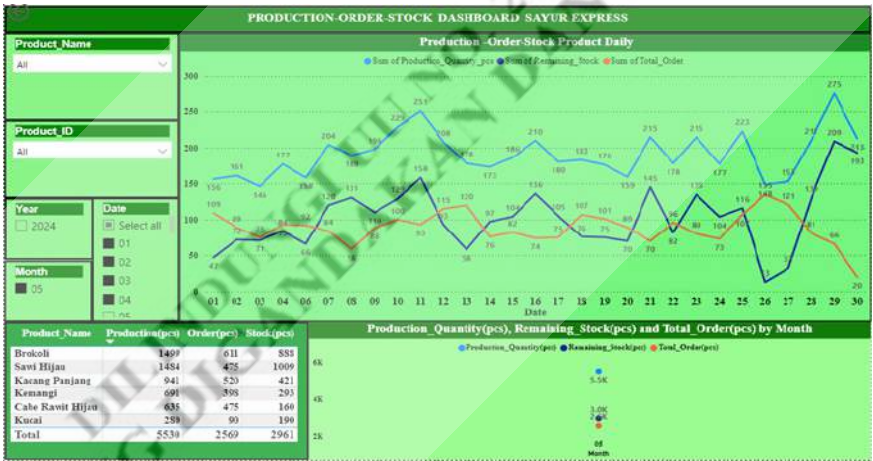
Pasokan harga beras masuk dan beras keluar dari PIBC juga dapat ditampilkan dalam diagram lingkaran. Terlihat pada Gambar 12 adalah diagram lingkaran daerah asal pasokan beras (beras masuk) yang dipilih pada data 4 bulan pertama pada tahun 2017. Terlihat bahwa 4 daerah utama pemasok beras ke DKI Jakarta melalui PIBC adalah beras yang berasal dari Cirebon, Jawa Tengah, Karawang dan Bandung.

Jenis dan kategori produk terlaris berdasarkan nilainya. Analisis produk dan kategori terlaris dapat mengidentifikasi produk dengan nilai margin tertinggi untuk meningkatkan keuntungan. Promosi penjualan silang untuk produk dengan margin rendah. Dashboard menunjukkan fluktuasi harian dalam nilai dan volume penjualan. Tanggal 26 Mei merupakan puncak penjualan dengan nilai 12,9 M (Rp) dan volume 6,5M (gram). Namun, setelah tanggal 26 Mei hingga 30 Mei terjadi penurunan drastis. Wawasan dari penjualan puncak adalah strategi pemasaran atau promosi pada waktu tertentu ketika permintaan tinggi. Lonjakan penurunan volume mengindikasikan adanya inefisiensi distribusi atau penurunan permintaan. Untuk itu, perlu dilakukan perbaikan rantai pasok ketika terjadi tren penurunan permintaan.



280 | Intelijensia Bisnis dan Data Analitik

membantu mengidentifikasi kekurangan atau kelebihan produksi. Misalnya, pada tanggal 27-29, terjadi lonjakan total produksi sebesar 275 sementara total pesanan 66 sehingga menghasilkan stok yang besar yaitu 209. Hal ini menunjukkan inefisiensi produksi. Data ini dapat digunakan untuk menentukan jumlah dan jenis sayuran yang harus dibeli untuk produksi agar tidak terjadi kelebihan stok yang dapat menyebabkan kerugian. Dengan data ini, pemilik UMKM dapat meramalkan dan menyesuaikan kapasitas produksi sesuai dengan tren permintaan pelanggan. Pemilik UMKM dapat fokus pada produk yang banyak diminati pelanggan dan mengurangi produksi untuk produk yang kurang diminati. Pemilik UMKM dapat menentukan kapasitas distribusi harian yang optimal sesuai dengan tingkat permintaan. Pemilik UMKM dapat membuat strategi promosi atau diskon yang tepat untuk produk dengan tingkat stok yang tinggi.



Gambar 223. Dashboard stok – pesanan - produksi

E. Rangkuman

Visualisasi Data merupakan hasil dari visualisasi data yang digunakan untuk mengeksplorasi, memahami, dan menyampaikan data. Visualisasi ini menampilkan informasi dari data yang sudah dikumpulkan, selanjutnya di ringkasan, dan pemberian konteks pada data. Pada visualisasi ini hanya memperlihatkan inti informasi, bukan keseluruhan data mentah. Visualisasi data memiliki hubungan dengan grafik informasi, visualisasi ilmiah, dan statistik grafis. Visualisasi ini biasanya disampaikan melalui dashboard, dimana dashboard dapat menyajikan informasi penting secara visual yang telah dan diatur dalam satu tampilan layar, memungkinkan informasi tersebut untuk dengan mudah dicerna dan dijadikan untuk pengambilan keputusan

F. Soal Latihan

1. Jelaskan tahapan pembuatan Donut Chart menggunakan *Power Business Intelligence*.
2. Jelaskan tahapan pembuatan stacked bar chart dashboard menggunakan *Power Business Intelligence*.
3. Kerjakan analisa dengan menggunakan data Aglonema di Jawa Barat terlampir <https://bit.ly/3YDj4aD> dengan menggunakan *Power Business Intelligence*.

BAB IX

Evaluasi Sistem Intelijensia Bisnis

A. Tujuan dan Capaian Pembelajaran

Tujuan:

Mahasiswa mampu menjelaskan dan mengetahui aspek evaluasi sistem intelijensia bisnis. Setelah membaca buku ini diharapkan mahasiswa mengetahui evaluasi sistem intelijensia bisnis.

Materi:

1. Evaluasi Sistem Intelijensia Bisnis.
2. Validasi.
3. Verifikasi.

B. Evaluasi Sistem Intelijensia Bisnis

Perencanaan yang cermat diperlukan untuk mendapatkan hasil maksimal dari pengujian dan proses pemeriksaan. Perencanaan harus dimulai sejak awal proses pembangunan. Rencana tersebut harus mengidentifikasi keseimbangan antara verifikasi statis dan pengujian. Perencanaan pengujian adalah tentang menentukan standar untuk proses pengujian daripada menjelaskan pengujian produk.

Evaluasi sistem intelijensia bisnis melibatkan penilaian berbagai aspek untuk memastikan sistem tersebut efektif dalam mendukung pengambilan keputusan dan mencapai tujuan bisnis. Ini meliputi penilaian dari segi data, teknologi, proses, dan dampak terhadap bisnis.

Berikut adalah beberapa aspek yang perlu dipertimbangkan dalam evaluasi sistem intelijensia bisnis:

1. Keakuratan dan Kualitas Data:

a. Validasi data:

Memastikan data yang digunakan dalam sistem BI akurat, konsisten, dan relevan untuk analisis.

b. Integrasi data:

Memastikan data dari berbagai sumber dapat diintegrasikan dengan baik dan digunakan secara konsisten dalam sistem BI.

c. Penanganan data yang hilang atau tidak lengkap:

Memastikan sistem BI dapat mengelola data yang tidak lengkap atau memiliki kesalahan.

2. Keandalan dan Keamanan Sistem:

a. Keandalan:

Memastikan sistem BI dapat diandalkan dan berfungsi dengan baik secara terus-menerus.

b. **Keamanan data:**

Memastikan data yang disimpan dan diakses dalam sistem BI aman dari akses yang tidak sah.

c. **Pemulihan bencana:**

Memastikan sistem BI memiliki mekanisme pemulihan yang efektif jika terjadi kegagalan.

3. Kinerja Sistem:

a. *Waktu respon:* Memastikan sistem BI memberikan informasi secara cepat dan efisien.

b. *Ketersediaan:* Memastikan sistem BI dapat diakses oleh pengguna ketika dibutuhkan.

c. *Skalabilitas:* Memastikan sistem BI dapat menyesuaikan diri dengan pertumbuhan data dan kebutuhan pengguna.

4. Pengalaman Pengguna:

a. *Sederhana dan intuitif:* Memastikan sistem BI mudah digunakan dan dipahami oleh pengguna.

b. *Visualisasi data:* Memastikan sistem BI menyediakan visualisasi data yang efektif dan informatif.

c. *Navigasi yang mudah:* Memastikan pengguna dapat dengan mudah mengakses informasi yang mereka perlukan dalam sistem BI.

5. Dampak Terhadap Bisnis:

a. *Peningkatan efisiensi:* Memastikan sistem BI membantu meningkatkan efisiensi proses bisnis.

b. *Peningkatan pengambilan keputusan:* Memastikan sistem BI memberikan informasi yang akurat dan relevan untuk mendukung pengambilan keputusan.

c. *Peningkatan keuntungan:* Memastikan sistem BI membantu meningkatkan keuntungan bisnis.

6. Pemeliharaan dan Pengembangan:

- a. *Pemeliharaan sistem*: Memastikan sistem BI dipertahankan dan diperbarui secara teratur.
- b. *Pengembangan sistem*: Memastikan sistem BI dapat dikembangkan dan diintegrasikan dengan teknologi baru.

7. Metode Evaluasi:

Evaluasi sistem BI dapat dilakukan dengan berbagai metode, seperti:

a. Survei:

Mengumpulkan umpan balik dari pengguna tentang pengalaman mereka menggunakan sistem BI.

b. Wawancara:

Melakukan wawancara dengan pengguna dan pemangku kepentingan untuk memahami kebutuhan dan harapan mereka.

c. Pengujian kinerja:

Menilai kinerja sistem BI dengan menguji waktu respon, ketersediaan, dan skalabilitas.

d. Pengamatan:

Memantau penggunaan sistem BI oleh pengguna untuk mengidentifikasi masalah dan area yang perlu ditingkatkan.

Tools dan Teknologi Evaluasi:

1. **Tools analitik data**: Alat yang digunakan untuk mengumpulkan, menganalisis, dan memvisualisasikan data untuk mengevaluasi sistem BI.
2. **Software manajemen proyek**: Alat yang digunakan untuk mengelola proses evaluasi sistem BI.
3. **Sistem manajemen basis data**: Alat yang digunakan untuk mengelola dan menyimpan data yang digunakan dalam

evaluasi sistem BI.

Dengan mengevaluasi sistem BI secara cermat, perusahaan dapat memastikan bahwa sistem tersebut memberikan nilai yang optimal dan membantu mencapai tujuan bisnis.

Verifikasi dan validasi bukanlah hal yang sama. Verifikasi menunjukkan kesesuaian dengan spesifikasi; validasi menunjukkan bahwa program tersebut memenuhi kebutuhan pelanggan. Rencana pengujian harus dibuat untuk memandu proses pengujian. Teknik verifikasi statis melibatkan pemeriksaan dan analisis program untuk mendeteksi kesalahan.

C. Validasi

Validasi tes adalah sebuah metodologi untuk mengetes software (Perry, 2006). Model Sistem Intelijensia Bisnis divalidasi dengan menggunakan teknik face validity (Sargent, 1999; Sekaran, 2000). Pertanyaan yang diajukan adalah model-model telah bermakna dan mempresentasikan sistem nyata serta memiliki kemanfaatan dalam aplikasinya untuk kemudian diajukan kepada para pakar. Para pakar tersebut merupakan manajer. Misalnya: Manajer koperasi susu hanya satu orang, sehingga dua orang pakar dianggap cukup mewakili pendapat koperasi susu keseluruhan. Kedua pakar tersebut dipresentasikan mengenai model dan Sistem Intelijensia Bisnis dan diberikan kuesioner untuk memberikan penilaian. Pilihan jawaban mulai dari sangat setuju sampai sangat tidak setuju.

Struktur rencana pengujian perangkat lunak

Proses pengujian.

1. Ketertelusuran persyaratan.
2. Software yang sudah diuji.
3. Jadwal pengujian.
4. Uji prosedur pencatatan.
5. Persyaratan perangkat keras dan perangkat lunak.

6. Kendala.

Face validity wajah penting karena ini merupakan langkah pertama yang sederhana untuk mengukur validitas keseluruhan suatu tes atau teknik. Ini adalah cara yang relatif intuitif, cepat, dan mudah untuk mulai memeriksa apakah suatu tindakan baru tampak berguna pada awalnya lirikan.

Face validity yang baik berarti siapa pun yang meninjau pengukuran Anda mengatakan bahwa pengukuran tersebut tampaknya mengukur apa itu seharusnya.

Dengan face validity yang buruk, seseorang yang meninjau pengukuran Anda mungkin akan bingung tentang siapa Anda mengukur dan mengapa Anda menggunakan metode ini.

Untuk mendapatkan validitas wajah, ukuran Anda harus jelas relevan dengan apa yang diukur, sesuai untuk peserta dan memadai untuk tujuannya.

Validasi clustering penting untuk keberhasilan aplikasi pengelompokan. Ukuran numerik yang disebut juga sebagai kriteria atau indeks, diterapkan untuk menilai berbagai aspek validitas cluster. Validasi clustering dapat dikategorikan menjadi dua jenis utama (Kantardzic, 2020):

- a. Internal measures: Digunakan untuk mengukur struktur pengelompokan yang baik tanpa menggunakan informasi eksternal atau ketika informasi eksternal tidak tersedia. Langkah-langkah validasi internal hanya mengandalkan informasi dalam data.
- b. External measures: Digunakan untuk mengukur sejauh mana label cluster cocok dengan label kelas yang disediakan secara eksternal. Misalnya, sampel yang diberi label kelas sebagai “ground truth”.

Setidaknya ada 12 internal measures untuk validasi clustering yaitu:

- a. Root-mean-square standard deviation (RMSSTD), akar kuadrat dari varians sampel gabungan dari semua atribut

untuk mengukur homogenitas cluster yang terbentuk.

- b. R-squared (RS), rasio jumlah kuadrat antara cluster dengan jumlah total kuadrat dari seluruh kumpulan data untuk mengukur tingkat perbedaan antara cluster.
- c. Modified Hubert Γ statistic (Γ), mengevaluasi perbedaan antara
- d. S Dbw index (S Dbw), memperhitungkan kepadatan untuk mengukur pemisahan intercluster. Ide dasarnya adalah bahwa untuk setiap pasangan pusat cluster, setidaknya satu dari kepadatan mereka harus lebih besar dari kepadatan titik tengah mereka. Kekompakan intracluster sama dengan SD. Demikian pula, indeks adalah penjumlahan dari kedua istilah ini dan nilai minimum S Dbw menunjukkan nomor cluster optimal.
- e. Clustering Validation index based on Nearest Neighbors (CV NN), mengevaluasi pemisahan intercluster berdasarkan objek yang membawa informasi geometris dari setiap cluster. CV NN menggunakan beberapa objek dinamis sebagai perwakilan untuk cluster yang berbeda dalam situasi yang berbeda ketika mengukur pemisahan intercluster. CV NN juga menggunakan jarak berpasangan rata-rata antara objek dalam cluster yang sama dengan pengukuran kekompakan intracluster. CV NN mengambil bentuk penjumlahan pemisahan intercluster dan kekompakan intracluster setelah normalisasi untuk keduanya.

Davies Bouldin Index merupakan salah satu metode yang digunakan untuk mengukur validitas cluster dalam metode pengelompokan, kohesi didefinisikan sebagai penjumlahan kedekatan data dengan titik pusat cluster dari cluster yang diikuti. Davies Bouldin Index termasuk dalam kategori internal measures. Hasil clustering terbaik ditentukan berdasarkan nilai Davies Bouldin Index yang mendekati 0 (Mughnyanti et al., 2020). Kelemahan umum untuk semua algoritma cluster, yaitu kinerjanya sangat bergantung pada pengguna yang mengatur berbagai parameter.

Bahkan, pengaturan yang tepat biasanya hanya dapat ditentukan dengan metode trial and error. Secara substansial Davies Bouldin Index mengatasi kelemahan ini dengan hanya meminta pengguna untuk menentukan jarak dan ukuran dispersi yang akan digunakan.

Terdapat dua aktivitas utama pada evaluasi sistem business intelligence yaitu verifikasi dan validasi. Uji verifikasi sebagai pembuktian bahwa tidak ada kesalahan pada sistem dan berjalan sesuai spesifikasi yang telah ditentukan dengan cara membandingkan hasil uji dengan indikator keberhasilan. Sistem dapat dikatakan lulus uji verifikasi jika bisa berjalan semestinya tanpa ada kecacatan (bug). Uji validasi rancangan sistem dilakukan untuk membuktikan bahwa sistem mampu menyajikan informasi sesuai dengan harapan dan kebutuhan pengguna menggunakan metode System Usability Scale (SUS). (Atsani et.al,2019)

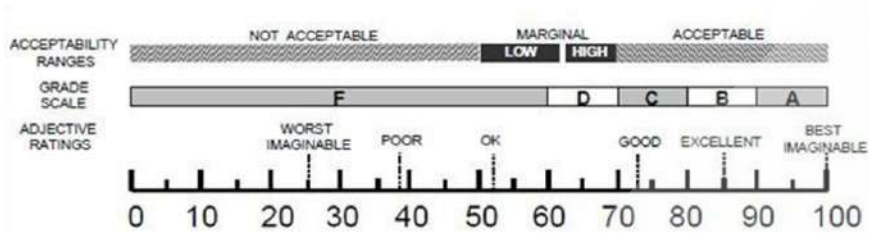
Uji validasi rancangan sistem dilakukan untuk membuktikan bahwa sistem mampu menyajikan informasi sesuai dengan harapan dan kebutuhan pengguna menggunakan metode *System Usability Scale (SUS)*. Metode *SUS* terdiri dari 10 pertanyaan dan 5 pilihan jawaban dengan bobot nilai antara 0 sampai 4 menggunakan skala Likert. Pertanyaan dengan nomor ganjil memiliki makna positif, sedangkan pertanyaan dengan nomor genap memiliki makna negatif (Fitriana, et.al.,2024). Daftar pertanyaan kuesioner *SUS* dapat dilihat pada Tabel 21.

Tabel 34. Daftar Pertanyaan Kuesioner *SUS*

No	Pertanyaan	Jawaban				
		STS	TS	RG	ST	SS
		1	2	3	4	5
1	Saya berpikir akan menggunakan sistem ini lagi.					
2	Saya merasa sistem ini rumit untuk digunakan.					
3	Saya merasa sistem ini mudah digunakan.					

4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.					
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini).					
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.					
8	Saya merasa sistem ini membingungkan.					
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini.					
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					
STS : Sangat Tidak Setuju, TS : Tidak Setuju RG : Ragu-ragu, ST : Setuju, SS : Sangat Setuju						

Kuesioner *SUS* diisi oleh *stakeholder* sistem *business intelligence* yaitu ketua program studi dan ketua LPPM. Nilai pada pertanyaan bernomor ganjil dihitung dengan cara nilai jawaban dikurangi dengan nilai 1, sedangkan nilai pada pertanyaan bernomor genap dihitung dengan cara 5 dikurangi nilai jawaban. Total nilai akhir dihitung dengan cara menjumlahkan total nilai setiap pertanyaan. Total nilai akhir dikalikan dengan 2,5 dan dibagi dengan jumlah responden untuk mendapatkan nilai *SUS* antara 0 sampai 100. Sistem dikategorikan baik jika nilai akhir *SUS* ≥ 70 [33]. Grafik peringkat persentil nilai *SUS* dapat dilihat pada Gambar 4. 35.



Gambar 224. Grafik Peringkat Persentil Nilai *SUS*

Jawaban kuesioner dari setiap responden bisa dilihat pada Tabel 4. 17 dan hasil perhitungan nilai *SUS* dapat dilihat pada Tabel .

Tabel 35. Jawaban Kuesioner *SUS*

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Kaprodi 1	5	2	4	1	4	1	5	2	5	1
Kaprodi 2	5	2	4	2	5	1	4	1	4	1
Kaprodi 3	5	2	4	1	5	2	4	2	5	1
Kaprodi 4	4	1	4	1	4	2	4	2	4	2
Ketua LPPM	5	1	4	1	4	1	5	2	4	2

Tabel 36. Perhitungan Nilai *SUS*

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Total
Kaprodi 1	4	3	3	4	3	4	4	3	4	4	36
Kaprodi 2	4	3	3	3	4	4	3	4	3	4	35
Kaprodi 3	4	3	3	4	4	3	3	3	4	4	35
Kaprodi 4	3	4	3	4	3	3	3	3	3	3	32
Ketua LPPM	4	4	3	4	3	4	4	3	3	3	35
Total Nilai Akhir											173
Nilai <i>SUS</i>	$(173 \times 2,5) / 5$										86,5

Hasil perhitungan nilai *SUS* dibandingkan dengan grafik peringkat persentil nilai *SUS* dapat disimpulkan bahwa sistem

business intelligence dengan nilai 86,5 termasuk dalam kategori baik serta layak digunakan.

Contoh kasus 2

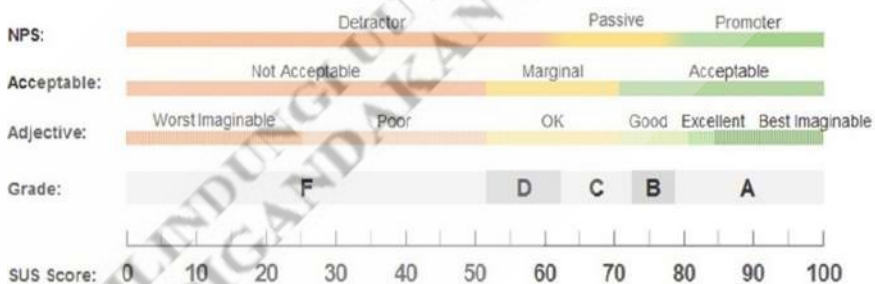
Pengujian validasi dengan menggunakan *System Usability Scale* (*SUS*). Sistem ini digunakan untuk melihat apakah model yang digunakan dapat memberikan informasi sesuai kebutuhan serta harapan. Metode ini memiliki 10 pertanyaan dan 5 pilihan jawaban menggunakan skala likert dari 0-4. Pertanyaan pada nomor 1, 3, 5, 7, 9 atau dengan nomor ganjil memiliki pertanyaan yang sifatnya positif, sedangkan pada pertanyaan nomor 2, 4, 6, 8, 10 atau dengan nomor genap memiliki pertanyaan yang sifatnya negatif Berikut merupakan pertanyaan dari kuesioner *SUS*.

Tabel 37. Kuesioner *SUS*

No	Pertanyaan	Jawaban				
		STS	TS	RG	ST	SS
		1	2	3	4	5
1	Saya berpikir akan menggunakan sistem ini lagi.					
2	Saya merasa sistem ini rumit untuk digunakan.					
3	Saya merasa sistem ini mudah digunakan.					
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.					
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini).					
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.					

8	Saya merasa sistem ini membingungkan.					
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini.					
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					
STS : Sangat Tidak Setuju, TS : Tidak Setuju RG : Ragu-ragu, ST : Setuju, SS : Sangat Setuju						

Dengan total dihitung dengan menjumlahkan total nilai setiap pertanyaan. Nilai akhir diperoleh dengan mengalikan total skor dengan 2,5 kemudian dibagi dengan jumlah responden. Jika nilai akhir $SUS \geq 70$ [45] maka dapat disimpulkan bahwa sistem dikategorikan baik. Grafik peringkat persentil nilai SUS dapat dilihat sebagai berikut.



Gambar 225. Grafik Peringkat Persentil Nilai SUS

Tabel 38. Skala Interpretasi SUS

Grade	SUS	Percentile range	Adjective	Acceptable	NPS
A+	84.1 – 100	96 – 100	Best Imaginable	Acceptable	Promoter
A	80.8 – 84.0	90 – 95	Excellent	Acceptable	Promoter
A-	78.9 – 80.7	85 – 89	Good	Acceptable	Promoter
B+	77.2 – 78.8	80 – 84		Acceptable	Passive
B	74.1 – 77.1	70 – 79		Acceptable	Passive
B-	72.6 – 74.0	65 – 69		Acceptable	Passive
C+	71.1 – 72.5	60 – 64		Acceptable	Passive
C	65.0 – 71.0	41 – 59	OK	Marginal	Passive
C-	62.7 – 64.9	35 – 40		Marginal	Passive
D	51.7 – 62.6	15 – 34		Marginal	Detractor

Dalam pengaplikasiannya, kuesioner diisi oleh lima responden dengan hasil sebagai berikut:

Tabel 39. Jawaban Kuesioner

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Responden 2.1	4	2	4	3	5	2	4	1	4	2
Responden 2.2	4	2	5	2	5	3	5	1	4	2
Responden 2.3	5	1	4	2	5	2	5	2	4	2
Responden 2.4	5	1	4	2	5	2	4	1	4	2
Responden Efficiency	5	1	5	3	5	1	5	2	5	2

Tabel 40. Perhitungan Nilai

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Total
Responden 2.1	4	2	4	3	5	2	4	1	4	2	31
Responden 2.2	4	2	5	2	5	3	5	1	4	2	33
Responden 2.3	5	1	4	2	5	2	5	2	4	2	32
Responden 2.4	5	1	4	2	5	2	4	1	4	2	30
Responden Efficiency	5	1	5	3	5	1	5	2	5	2	34
Total Nilai Akhir										160	
Nilai SUS	$(153 \times 2,5) / 5$									80	

Berdasarkan nilai yang didapat yaitu 80, dapat diartikan bahwa sistem dapat berjalan dengan baik, dengan nilai kategori yang didapat yaitu grade A- dengan status *acceptable* atau dapat diterima dengan rating yang didapat yaitu diantara *good* dengan *excellent*.

D. Verifikasi

Verification testing adalah metode pengetesan yang efektif untuk menghilangkan cacat di dalam software. Sedangkan tes validasi adalah sebuah metodologi untuk mengetes software (Perry, 2006).

Pengenalan data yang tersedia model matematika yang diadopsi dapat didefinisikan, dengan memastikan ketersediaan data yang dibutuhkan untuk setiap model dan verifikasi efisiensi algoritma yang akan digunakan akan cukup memadai untuk menyelesaikan besarnya persoalan.

Sistem Intelijensia Bisnis diverifikasi dengan jalan menguji apakah program untuk submodel tersebut telah berjalan dengan baik dan benar. Hal ini dilakukan dengan memberikan data input kepada setiap model program Sistem Intelijensia Bisnis dan hasil outputnya diperiksa apakah telah sesuai dengan hasil perhitungan memakai rumus. Bila masih ada kesalahan, maka program diperiksa dan diperbaiki. Setelah program berjalan dengan baik, maka akan ditentukan parameter-parameter submodel yang memberikan hasil paling optimal.

Dari sistem sebelumnya setiap komponen tidak didata menjadi satu kesatuan *database* yang mampu ditampilkan suatu dashboard, melainkan data yang ada berdiri sendiri, sehingga kemampuan manajerial untuk melihat performa secara keseluruhan tidak dapat terlihat, berikut ini merupakan penggambaran sistem sebelumnya.

Selanjutnya dari sistem yang ada kemudian digabungkan menjadi satu kesatuan database yang dapat melihat performa dari segi biaya ataupun performa efektifitas dan rejection proses produksi yang ada. Dengan bantuan dari Power BI manajerial mampu memilih keputusan yang lebih tepat, dan mampu memantu project yang berjalan dan memonitoring setiap bulannya, sehingga dihasilkan

Setelah data diolah dan dikerjakan sesuai framework, selanjutnya menjalankan dashboard power BI, untuk melihat apakah ada bug ataupun kendala saat memproses dan membaca trend dari

data yang ada. Berikut ini merupakan tabel hasil uji dari verifikasi sistem yang ada.

Tabel 41. Hasil uji verifikasi sistem

Daftar Uji	Langkah Pengujian	Indikator Keberhasilan	Hasil Pengujian	Kesimpulan
Menguji Dash-board Cost Performance Plant	Menjalankan Dashboard Cost Performance Plant	- Pada saat dashboard berjalan tidak ada bug ataupun kendala.	Tidak ada <i>bug</i> atau kendala saat dash-board di-jalankan.	Berhasil

Daftar Uji	Langkah Pengujian	Indikator Keberhasilan	Hasil Pengujian	Simpulan
Menguji Dash-board Cost Material Consumable	Menjalankan Dash-board Cost Material Consumable	- Pada saat dashboard berjalan tidak ada bug ataupun kendala.	Tidak ada <i>bug</i> atau kendala saat dashboard di-jalankan.	Berhasil

Menguji Dash-board Performance by type	Menjalan-kan Dash-board Performance by type	- Pada saat dashboard berjalan tidak ada bug ataupun kendala.	Tidak ada <i>bug</i> atau kendala saat dashboard dijalankan.	Berhasil
--	---	---	--	----------

E. Rangkuman

Evaluasi sistem dilakukan untuk menilai sistem sebelum digunakan, terdapat dua aktivitas utama pada evaluasi sistem intelijensia bisnis yaitu verifikasi dan validasi. Validasi adalah metode untuk melakukan pengujian software. Verifikasi adalah metode pengujian untuk menghilangkan cacat dalam software.

F. Soal Latihan

1. Jelaskan dan beri contoh apa yang dimaksud dengan validasi!
2. Jelaskan dan beri contoh apa yang dimaksud dengan verifikasi!

DAFTAR PUSTAKA

- Abbas S. A., Aslam A., Rehman A. U., Abbasi W. A., Arif S., and Kazmi S. Z. H. 2020. K-Means and K-Medoids: Cluster Analysis on Birth Data Collected in City Muzaffarabad, Kashmir, *IEEE Access*, vol. 8, pp. 151847–151855, 2020.
- Akbar R. , Rasyiddah D. , Anrisya M., Julyazti N. F., and Syaputri S. 2018. Penerapan Aplikasi Power Business Intelligence Dalam Menganalisis Prioritas Pekerjaan di Indonesia,” *J. Edukasi dan Penelit. Inform.*, vol. 4, no. 1, p. 54, 2018, doi: 10.26418/jp.v4i1.25497.
- Amin M. M., Sutrisman A., Dwitayanti Y., “Development of Star-Schema Model for Lecturer Performance in Research Activities,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 9, pp. 74–80, 2021, doi: 10.14569/IJACSA.2021.0120909.
- Adila N., Sembiring F., Jatmiko W..2022. Implementation of Web Scraping for Journal Data Collection on the SINTA Website, *Sinkron*, vol. 7, no. 4, pp. 2478–2485, 2022, doi: 10.33395/sinkron.v7i4.11576.
- Atsani M. R., Anjari G. T., Saraswati N. M. 2019. Pengembangan Business Intelligence Di Rumah Sakit (Studi Kasus: RSUD Prof. Dr. Margono Soekarjo Purwokerto),” *Telematika*, vol. 12, no. 2, pp. 124–138, 2019, doi: 10.35671/telematika.v12i2.839.
- Berahmana R. W. B. S. , Mohammed F. A., and Chairuang K. 2020. Customer Segmentation Based on RFM Model Using K Means, K Medoids and DBScan Methods. *Lontar Komputer*, Vol. 11, no. 1, doi : 10.24843/LKJITI.2020.v11.i01.p04.
- Chang, E., Hussain, F., & Dillon, T. 2020. Trust and Reputation for Service-Oriented Environments: Technologies For Building Business Intelligence And Consumer Confidence. John Wiley & Sons.
- DAMA International. 2017. Data Management Body of Knowledge (D. Henderson & S. Earley (eds.)). Technics Publications.

Edraw Content Team. 2020. ER Diagram Symbols and Notations. Edrawsoft.Com. <https://www.edrawsoft.com/er-diagram-symbols.html>

Fitriana R., Eriyatno, Djatna T.. 2011. Progress in Business Intelligence System research: a literature review. International Journal of Basic & Applied Sciences IJBAS IJENS.

Fitriana R., Habyba A.N., Febriani E. 2022. Data Mining dan Aplikasinya. Contoh Kasus di Industri Manufaktur dan Jasa. Banyumas:Wawasan Ilmu.

Fitriana R, Saragih J., Hasyati B.A. 2018. Perancangan Model Sistem Intelijensia Bisnis Untuk Menganalisis Pemasaran Produk Roti Di Pabrik Roti Menggunakan Metode Data Mining Dan Cube. Jurnal Teknologi Industri Pertanian 28 (1):113-126.

Fitriana R. , Saragih J. , and Aseanita M. A. 2019. Model Sistem Intelijensia Bisnis Untuk Perbaikan Pelayanan E-Service Pada PT. X," *J. Tek. Ind.*, vol. 9, no. 2, pp. 112–120. doi: 10.25105/jti.v9i2.4925.

Fitriana,R., Sugiarto,D., Hidayat M.K., Turnip Y.E.H. 2024. Perancangan Model Sistem Intelijensia Bisnis. Banyumas: Wawasan Ilmu

Filiana A., Prabawati A. G., Rini M. N. A, VirginiaG., and SusantoB. 2020. Perancangan Data Warehouse Perguruan Tinggi untuk Kinerja Penelitian dan Pengabdian kepada Masyarakat. *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 2, pp. 174–183. doi: 10.28932/jutisi.v6i2.2557.

Fernandez F. V. 2020. Intelligent information extraction from scholarly document databases. *JISIB (Journal Intell. Stud. Business)*, vol. 10, no. 3, pp. 38–62.

Febiyanti W. , K. Ghozali, and A. S. Indrawanti. 2022. Rancang Bangun Aplikasi Dashboard Penjualan, Logistik dan Tenaga Kerja di Pt. XYZ," *J. Tek. ITS*, vol. 11, no. 2, 2022, doi: 10.12962/j23373539.v11i2.85815.

Han, Jiawei., Micheline Kamber., Jian Pei. 2012. DATA MINING

Concepts and Techniques. 225 Wyman Street, Waltham, USA : Morgan Kaufmann Publishers

Hastie T, R. Tibshirani dan J. Friedman. 2008. The Elements of Statistical Learning, Data Mining, Inference, and Prediction. Ed ke-2. New York: Springer

Hermawan A, MangkuI. W., Ardana N. K. K, and Sumarno H. 2022. Analisis Support Vector Regression Dengan Algoritma Grid Search Untuk Memprediksi Harga Saham. *J. Math. Its Appl.*, vol. 18, no. 1, pp. 41–60, 2022, doi: 10.29244/milang.18.1.41-60

Hidayat M. K., Fitriana R. 2022. Penerapan Sistem Intelijensia Bisnis Dan K-Means Clustering Untuk Memantau Produksi Tanaman Obat. *J. Teknol. Ind. Pertan.*, vol. 32, no. 2, pp. 204–219.

Hidayat M. K., Fitriana R.. 2022. Implementasi K-Means Dan K-Medoids Dalam Pengelompokan Wilayah Potensial Produksi Daging Ayam,” vol. 32, no. 3, pp. 239– 247.

Hidayat M.K , Sugiarto D. , Fitriana R., Liang Y. C. Business Intelligence System Model to Measure the Performance of Lecturers. *TELKOMNIKA Telecommunication Computing Electronics and Control* Vol. 23, No. 4, August 2025, pp. 954~964 ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v23i4.26221

Indraputra R. A. and Fitriana R.2020. K-Means Clustering Data COVID-19,” *J. Tek. Ind.*, vol. 10, no. 3, pp. 275–282, 2020, doi: 10.25105/jti.v10i3.8428.

Jiawei, H., Kamber, M., Pei, J. 2012. Data Mining Concepts and Techniques. 225 Wyman Street, Waltham, USA : Morgan Kaufmann Publishers

Joshi M., Dubbewar A..2021.Review on Business Intelligence, Its Tools and Techniques, and Advantages and Disadvantages. *Int. J. Eng. Res. Technol.*, vol. 10, no. 12, pp. 386– 391.

Kantardzic M. 2020. *Data Mining Concepts, Model, and Algorithms*, 3rd ed. Piscataway: IEEE Press.

Karmani M. Z.I, Mustafa G., Sarwar N. , S. H. Wajid, J. Nasir, and

- S. Siddque. 2020. A Review of Star Schema and Snowflakes Schema," *Commun. Comput. Inf. Sci.*, vol. 1198, pp. 129–140, 2020, doi: 10.1007/978-981-15-5232-8_12.
- Kraynak, J., & Baum, D. 2020. *Cloud Data Warehousing For Dummies*. John Wiley & Sons, Inc
- Krishnamoorthy M, Karthikeyan R. 2022. Competence of medicinal plant database using data mining algorithms for large biological databases, *Meas. Sensors*, vol. 24, no. August, p. 100420, doi: 10.1016/j.measen.2022.100420.
- Liu Luhao. 2010. Supply Chain Integration through Business Intelligence; Management and Service Science (MASS), International Conference on Digital Object Identifier. Publication Year: 2010 , Page(s): 1 – 4 IEEE Conferences
- Metre K. V, Mathur A., Dahake R.P., Bhapkar Y, Ghadge J., Jain P., Gore S. 2024. An Introduction to Power BI for Data Analysis. *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 1s, pp. 142–147.
- Mughnyanti M., Efendi S. , Zarlis, M. 2020. Analysis of determining centroid clustering x- means algorithm with davies-bouldin index evaluation," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 725, no. 1, 2020, doi: 10.1088/1757-899X/725/1/012128.
- Orisa M. 2022. Optimasi Cluster pada Algoritma K-Means," in *Seminar Nasional 2022 METAVERSE: Peluang Dan Tantangan Pendidikan Tinggi Di Era Industri 5.0*. pp. 430–437.
- Negash, S. 2004. Business Intelligence. *Communications of the Association for Information Systems*, 13. <https://doi.org/10.17705/1CAIS.01315>
- Nurachman, Sugiarto D, Fitriana R. 2025. Customer Segmentation With K-Means Algorithm And Business Strategy Business Intelligence In Vegetable Online Retailing, *Jurnal Teknologi Industri Pertanian* 35 (2): 118-135, August 2025
- Piedade, M.B et a.2010. Business intelligence in higher education: Enhancing the teaching-learning process with a SRM system. *Information Systems and Technologies (CISTI)*, 5th Iberian

Conference. Page(s): 1 – 5 IEEE Conferences.

Ranjan, J. 2009. Business Intelligence: Concepts, components, techniques and benefits. *Journal of Theoretical and Applied Information Technology*, 9(1), 60–70.

Rehman, M. U., Ullah, R., Allowatia, H., Hasan, T. N., Perween, S., Ain, Q. U., Ammad, M. 2022. Elaborating the Role of Business Intelligence (BI) in Healthcare Management. *JISIB (Journal of Intelligence Studies in Business)*, 12(2), 26–35.

Santoso M. H., 2021. Application of Association Rule Method Using Apriori Algorithm to Find Sales Patterns Case Study of Indomaret Tanjung Anom. *Brilliance Research of Artificial Intelligence*. vol. 1, no. 2. <https://doi.org/10.47709/brilliance.vli2.1228>.

Sharda R., Delen D. , and Turban E. 2018. *Business Intelligence, Analytics, And Data Science: A Managerial Perspective*. Pearson Education Limited.

ShardaR. , Delen D. , E. Turban, J. E. Aronson, T.-P. Liang, and David. King.2018. *Business intelligence, analytics, and data science : a managerial perspective*, 4th ed. Pearson..

Setiyani L. , Rostiani Y. , Ratnasari T. 2020. Analisis Kebutuhan Fungsional Sistem Informasi Persediaan Barang Perusahaan General Trading (Studi Kasus : PT. Amco Multitech),” *Owner*, vol. 4, no. 1, p. 288, doi: 10.33395/owner.v4i1.205.

Sinaga K. P., Yang M. S.. 2020. Unsupervised K-means clustering algorithm, *IEEE Access*, vol. 8, pp. 80716– 80727, 2020.

Sindhu D, Sangwan A. 2017. Optimization of Business Intelligence using Data Digitalization and Various Data Mining Techniques,” *Int. J. Comput. Intell. Res.*, vol. 13, no. 8, pp. 1991–1997, [Online]. Available: <http://www.ripublication.com>.

Sornalakhsmi M., Balamurali S., Venkatesulu M., Navaneetha K. M., Ramasamy L. K., Kadry S., Manogaran G., Hsu C. H., and Muthu B. A. 2020. Hybrid method for mining rules based on enhanced Apriori algorithm with sequential minimal optimization in

healthcare industry. *Neutral Computing Application*, 2. <https://doi.org/10.1007/s00521-020-04862-2>.

Susilowati T. , Sugiarto D. , and Mardianto I. 2020. Uji Validasi Algoritme Self-Organizing Map (SOM) dan K-Means untuk Pengelompokan Pegawai, *Rekayasa Sist. dan Teknol. Inf.*, vol. 4, no. 6, pp. 1171–1178.

Sugiarto D. , I. Mardianto, M. Najih, D. Adrian, and D. A. Pratama. 2021. Perancangan Dashboard Untuk Visualisasi Harga Dan Pasokan Beras Di Pasar Induk Beras Cipinang. *J. Teknol. Ind. Pertan.*, vol. 31, no. 1, pp. 12–19.

Syadzali C, Suryono S, and Endro Suseno J. 2020. Business Intelligence using the K-Nearest Neighbor Algorithm to Analyze Customer Behavior in Online Crowdfunding Systems. *E3S Web Conf.*, vol. 202, pp. 1–7, 2020, doi: 10.1051/e3sconf/202020216005.

Turnip Y.E.H., Fitriana R.. 2023. Clustering Kabupaten Berdasarkan Luas Hutan Menggunakan Metode K Means di Provinsi Jawa Tengah. *Jurnal Teknologi Industri Pertanian IPB*. Vol. 33. No.1.pp 1-9.

Vaisman A and Zimányi E. 2014. *Data warehouse systems: Design and implementation*. Springer Berlin Heidelberg.

Vaisman, A., & Zimányi, E. (2014). *Data warehouse systems: Design and implementation*. In *Data Warehouse Systems: Design and Implementation*. Springer Berlin Heidelberg.

Watson, H. J. (2009). Tutorial: Business Intelligence – Past, Present, and Future. *Communications of the Association for Information Systems*, 25(1), 487–510. <https://doi.org/10.17705/1cais.02539>

Widjaja S. and Mauritsius T. .2019. The Development of Performance Dashboard Visualization with Power BI as Platform,” *Int. J. Mech. Eng. Technol.*, vol. 10, no. 5, pp. 235–249, 2019

Xie S. , Lin H., Chen Y., Duan H., Liu H. , and Liu B. 2023. Prediction of shear strength of rock fractures using support vector regression and grid search optimization,” *Mater. Today Commun.*, vol. 36, no. May, p. 106780. doi: 10.1016/j.

mtcomm.2023.106780.

Zhang Hai. 2009. Research Automated Negotiation Framework for Business Intelligence Systems. International Conference Networking and Digital Society.. ICNDS '09. Digital Object Identifier. Page(s): 292 – 295 IEEE Conferences.

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

DILINDUNGI UU NO. 28 TAHUN 2014
DILARANG DIGANDAKAN DAN DISEBARLUASKAN

PROFIL PENULIS



Dr. Ir. Rina Fitriana, S.T., M.M., IPM., adalah Associate professor dan Dosen S1 Jurusan Teknik Industri, S2 Magister Teknik Industri dan S3 Doktor Teknik Industri FTI Universitas Trisakti. Rina memperoleh gelar Sarjana Teknik dari Jurusan Teknik Industri FTI Universitas Trisakti, gelar Magister Manajemen dari Sekolah Tinggi Manajemen PPM di Jakarta, menyelesaikan program Doktor dari Program Studi Teknologi Industri Pertanian, Institut Pertanian Bogor dan gelar Insinyur dari Program Profesi Insinyur dari Unika Atma Jaya. Pada tahun 2017 sampai sekarang dipercaya menjadi Ketua Jurusan Teknik Industri FTI Universitas Trisakti. Beberapa mata kuliah yang diampu yaitu Pengendalian dan Penjaminan Mutu, Data Mining, Business Intelligence, Simulasi Sistem, Statistika Multivariat. Saat ini Rina aktif meneliti di bidang Rekayasa Kualitas, Sistem dan Simulasi Industri, Data Mining, Business Intelligence. Rina memiliki Sertifikasi Dosen, Certified International Business Intelligence Associate and Professional, Certified in Big Data Associate dan Insinyur Profesional Madya dan Asean Engineer. Rina juga telah dipercaya menjadi Asesor Beban Kerja Dosen. Rina telah membuat buku mengenai ajar mengenai Pengendalian Penjaminan Mutu, Analisa Perancangan Sistem Informasi, Perancangan Data Mining dan aplikasinya. Contoh kasus di industri manufaktur dan jasa, Perancangan Model Sistem Intelijensia Bisnis. Rina dapat dihubungi pada email : rinaf@trisakti.ac.id.



Dr. Dedy Sugiarto, S.Si., M.M., M.Kom. lahir pada tanggal 14 Oktober 1969 di Jakarta. Pendidikan dasar ditempuh di SDN Tebet Timur 19 Jakarta, pendidikan menengah di SMPN 115 Jakarta dan SMAN 8 Jakarta. Pendidikan sarjana ditempuh di Program Studi Statistika, FMIPA IPB Bogor, lulus tahun 1993. Pendidikan S2 di Program Studi Magister Manajemen Universitas Trisakti, konsentrasi Manajemen Produksi/Operasi dengan

beasiswa dari Universitas Trisakti, lulus tahun 1998. Pendidikan S3 di Program Studi Teknologi Industri Pertanian IPB Bogor dengan beasiswa Universitas Trisakti dan bantuan hibah penelitian mahasiswa program doktor dari Kementerian Pendidikan Nasional, lulus tahun 2012. Pendidikan S2 kembali ditempuh di Program Studi S2 PJJ Teknik Informatika Universitas AMIKOM Yogyakarta, lulus tahun 2023. Karir pertama kali sebagai dosen tetap di Jurusan Teknik Industri, Fakultas Teknologi Industri (FTI), Universitas Trisakti tahun 1994 dengan penempatan pada Laboratorium Statistika Industri. Tahun 2015 pindah homebase sebagai dosen tetap di Program Studi Sistem Informasi, FTI Universitas Trisakti dan membantu mengajar di beberapa program studi seperti Magister Teknik Industri FTI Trisakti, Magister Manajemen FEB Universitas Trisakti serta Program Studi Manajemen Trisakti School of Management. Saat ini menjabat sebagai sekretaris Jurusan Teknik Informatika FTI Universitas Trisakti sejak 2017, mengelola dua program studi di bawahnya yaitu Program Studi Informatika dan Program Studi Sistem Informasi. Sejak 2021 mendapatkan tugas dan tanggung jawab sebagai ketua bidang data science, website dan media sosial pada Gugus Tugas Implementasi dan Integrasi Teknologi Informasi Universitas Trisakti. Dedy dapat dihubungi melalui email dedy@trisakti.ac.id

Miwan Kurniawan Hidayat, S.T., M.Kom., M.T. telah menyelesaikan pendidikan sarjana Teknik Informatika dari Universitas Pasundan Bandung pada tahun 2003, mendapatkan gelar Magister Ilmu Komputer dari Universitas Nusa Mandiri Jakarta pada tahun 2010, dan meraih gelar Magister Teknik Industri dari Universitas Trisakti Jakarta pada tahun 2023. Berprofesi sebagai praktisi Teknologi Informasi dan Dosen di Universitas Bina Sarana Informatika dengan fokus pada bidang Rekayasa Informasi. Telah mendapatkan sertifikat profesi berupa Sertifikasi Dosen dan sertifikat kompetensi sebagai Asesor Kompetensi, ICT Project Manager, Network Administrator, Programmer, Certificate in Business Intelligence Associate. Miwan juga telah menerbitkan banyak makalah dalam jurnal ilmiah. Miwan dapat dihubungi melalui email: miwan@bsi.ac.id.

Yusri Eli Hotman Turnip, S.T., M.T. adalah praktisi profesional di perusahaan global di bidang healthcare. Yusri menyelesaikan pendidikan Diploma 3 dari Jurusan Teknik Elektro Fakultas Teknik Universitas Gadjah Mada (UGM), menerima gelar Sarjana Teknik dari Jurusan Teknik Elektro Fakultas Teknik Universitas Pancasila, dan gelar Magister Teknik dari Jurusan Magister Teknik Industri FTI Trisakti. Memiliki pengalaman bekerja di beberapa bidang seperti pertambangan, konstruksi bangunan, dan healthcare selama lebih dari 10 tahun. Yusri memiliki Certified Business Intelligence Analyst (CBIA) pada tahun 2022 untuk bidang Business Intelligence. Saat mengambil studi Magister, Yusri membuat publikasi di bidang data mining yang berjudul District Clustering Based On Forest Area Using K-Means Method In Central Java Province yang terbit pada jurnal Teknologi Industri Pertanian IPB.