

**PROYEK AKHIR “PENGEMBANGAN DATA WAREHOUSE DAN  
DASHBOARD” SEMESTER GANJIL  
2024/2025**

**Dosen Pengampu**

Mohamad Irwan Afandi, ST., MSC.



**Disusun Oleh Kelompok**

Adelia Putri Pratiwi (22082010034)

Nanda Kharisma Safitri (22082010036)

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”  
JAWA TIMUR, SURABAYA  
2024**

## DAFTAR ISI

DAFTAR ISI.....	2
BAB I PENDAHULUAN.....	3
1.1. Latar Belakang.....	3
1.2. Rumusan Masalah.....	4
1.3. Batasan.....	4
BAB II METODE.....	5
2.1. Metode Pengembangan.....	5
2.1.1. Perencanaan.....	5
2.1.2. Perancangan Skema.....	5
2.1.3. Ekstraksi Data.....	6
2.1.4. Pembuatan CUBE Mondrian di Tomcat.....	6
2.1.5. Perancangan Dashboard.....	7
BAB III HASIL DAN PENGERJAAN.....	8
3.1. Implementasi Model Multidimensi.....	8
3.1.1. Skema dw_sales.....	8
3.1.2. Skema dw_purchasing.....	14
3.2. Implementasi Cube dan OLAP.....	20
3.2.1. Cube dan OLAP (dw_sales).....	20
3.2.2. Cube dan OLAP (dw_purchasing).....	24
3.3. Implementasi Dashboard.....	29
3.3.1. Form Registrasi.....	29
3.3.2. Login.....	30
3.3.3. Dashboard.....	30
3.3.4. Dashboard Chart Sales by Territory.....	31
3.3.5. Dashboard Chart Sales by Order.....	32
3.3.6. Dashboard Chart Sales by Category Product.....	32
3.3.7. Dashboard Chart Purchasing.....	33
3.3.8. Dashboard Chart Product Stock.....	34
3.3.9. Dashboard OLAP.....	35
LAMPIRAN.....	37
Link Github.....	37

## BAB I PENDAHULUAN

### 1.1. Latar Belakang

Data Warehouse adalah sistem penyimpanan data yang dirancang untuk mengumpulkan dan mengintegrasikan informasi dari berbagai sumber dalam suatu organisasi. Tujuan utamanya adalah untuk mendukung proses pengambilan keputusan melalui analisis dan pelaporan data yang lebih terstruktur dan terintegrasi. Dalam konteks proyek "Pengembangan Data Warehouse dan Dashboard", sistem ini bertujuan untuk menyediakan informasi yang relevan dan akurat terkait data bisnis, yang dapat membantu manajemen dalam pengambilan keputusan strategis.

AdventureWorks merupakan basis data yang digunakan untuk keperluan pembelajaran dan demonstrasi dalam industri perangkat lunak yang dikeluarkan oleh Microsoft. Database ini memiliki beberapa skema, seperti skema sales, purchasing, human resource, person, productions. Adventure Works bisa dijalankan di Microsoft SQL Server 2005 dan Microsoft SQL Server 2008 dimana Adventure Works telah menggantikan Northwind dan Pubs dari contoh database SQL Server 2005.

Adventure Works Cycle adalah perusahaan fiktif yang tergambar dalam database Adventureworks. Perusahaan ini merupakan perusahaan besar dengan operasi yang meluas secara internasional. Mereka bergerak dalam penjualan dan produksi sepeda berbahan logam dan komposit, yang dipasarkan di Amerika Utara, Eropa, dan Asia. Kantor pusat perusahaan terletak di Bothell, Washington, dan mereka memiliki 290 karyawan serta beberapa tim pemasaran yang beroperasi di pasar perusahaan di seluruh dunia.

Selama tahun yang berhasil dalam hal anggaran, Adventure Works Cycle melakukan ekspansi pasar dengan memfokuskan penjualan pada pelanggan terbaik mereka. Mereka juga meningkatkan ketersediaan produk melalui situs web eksternal dan mengurangi biaya penjualan melalui efisiensi produksi yang lebih tinggi. Semua langkah ini bertujuan untuk memperluas pangsa pasar dan memaksimalkan keuntungan perusahaan. Perancangan data warehouse yang kelompok kami buat adalah skema penjualan (sales) dan skema pembelian (purchasing).

## **1.2. Rumusan Masalah**

Dalam penelitian ini, kami fokus pada analisis empat ukuran (measure) yang penting untuk memahami kinerja penjualan dan pembelian di Adventure Works Cycle. Dua measure diambil dari skema penjualan (Sales), yaitu Sales Amount dan Sales Quantity, yang mencerminkan total pendapatan dan jumlah unit yang terjual. Sementara itu, dua measure dari skema pembelian (Purchasing), yaitu Order Quantity dan Total Cost, memberikan gambaran mengenai jumlah pesanan dan total biaya yang dikeluarkan. Measure ini saling terhubung melalui tabel fakta, yaitu tabel fakta penjualan (fact\_sales) dan tabel fakta pembelian (fact\_purchasing).

## **1.3. Batasan**

Pada proyek akhir ini, kami hanya membahas database warehouse AdventureWorks yang berfokus pada skema penjualan (Sales) dan skema pembelian (Purchasing). Oleh karena itu, data yang dihasilkan pada dashboard merupakan hasil dari proses ETL (Extract, Transform, Load) pada star schema Fact, dengan mengacu pada tabel fact\_sales dalam skema database penjualan, yaitu dw\_sales, dan tabel fact\_purchasing dalam skema database pembelian, yaitu dw\_purchasing.

## **BAB II METODE**

### **2.1. Metode Pengembangan**

#### **2.1.1. Perencanaan**

Pada tahap awal perencanaan, dilakukan identifikasi dengan berdiskusi untuk menentukan fokus analisis data. Dalam proyek ini, analisis difokuskan pada aspek penjualan (Sales) dan pembelian (Purchasing). Selanjutnya, dilakukan penentuan sumber data dengan mengidentifikasi tabel-tabel yang relevan dari database operasional AdventureWorks, khususnya dari skema Sales dan Purchasing. Data yang dipilih meliputi informasi penting seperti data transaksi penjualan, data pembelian, data pelanggan, data pemasok, serta data produk. Pemilihan data yang relevan ini bertujuan untuk memastikan bahwa data yang digunakan dalam proses pengembangan data warehouse dapat memberikan gambaran yang akurat dan sesuai dengan kebutuhan data yang analisis.

#### **2.1.2. Perancangan Skema**

Pada tahap perancangan skema, kami menentukan model Star Schema untuk mendesain data warehouse. Dalam model ini, database dw\_sales memiliki tabel fakta bernama fact\_sales, sedangkan database dw\_purchasing memiliki tabel fakta bernama factpurchasing. Masing-masing tabel fakta ini terhubung dengan tabel dimensi yang dirancang untuk menyederhanakan proses analisis dengan mengelompokkan data berdasarkan dimensi tertentu.

Setelah menentukan model dan ukuran (measure) yang akan digunakan, kami merancang tabel dimensi yang relevan. Tabel fakta penjualan (fact\_sales) mencakup ukuran seperti Sales Amount dan Sales Quantity, sedangkan tabel fakta pembelian (factpurchasing) mencakup ukuran seperti Order Quantity dan Total Cost. Langkah terakhir adalah pembuatan diagram ERD (Entity Relationship Diagram) untuk menggambarkan hubungan antara tabel fakta dan dimensi, yang membantu memvisualisasikan struktur data warehouse dan memastikan integritas hubungan antar tabel.

### **2.1.3. Ekstraksi Data**

Tahap ini berfokus pada pengambilan data dari database operasional AdventureWorks untuk mengisi tabel fakta. Proses dimulai dengan mengakses database menggunakan alat ETL seperti Pentaho untuk mengolah data dari skema Sales dan Purchasing.

Selanjutnya, dilakukan identifikasi tabel sumber untuk memastikan bahwa hanya data yang relevan dan diperlukan yang dipilih. Setelah tabel-tabel ini diidentifikasi, data diekspor ke dalam format yang dapat diolah lebih lanjut, seperti tabel staging atau file input yang digunakan untuk proses transformasi. Data yang diekstrak kemudian disiapkan untuk dimasukkan ke dalam data warehouse, khususnya ke dalam tabel fact\_sales pada database dw\_sales dan tabel factpurchasing pada database dw\_purchasing.

### **2.1.4. Pembuatan CUBE Mondrian di Tomcat**

Pada tahap ini, dilakukan pembuatan OLAP Cube (CUBE) menggunakan Mondrian dan di-deploy menggunakan Tomcat sebagai server aplikasi. Mondrian adalah mesin OLAP yang memungkinkan pengguna untuk membuat dan meng-query data multidimensi. Proses pembuatan CUBE menggunakan Mondrian meliputi langkah-langkah berikut:

1. Konfigurasi Mondrian: Instalasi Mondrian pada server yang akan digunakan, seperti Apache Tomcat. Mondrian diinstal untuk menjalankan dan melayani cube analisis data secara real-time.
2. Membuat definisi CUBE: Definisikan struktur CUBE yang menghubungkan tabel fakta (fact\_sales dan fact\_purchasing) dengan dimensi yang ada. Di sini, dimensi akan berfungsi untuk mengelompokkan data, sementara tabel fakta akan berisi measure yang dapat dianalisis.
3. Penyusunan file XML: Buat file XML dengan konfigurasi Mondrian, yang mendefinisikan struktur data cube, termasuk dimensi, hierarki, dan measure. File ini penting untuk mendefinisikan bagaimana data akan diproses dan disajikan dalam bentuk multidimensi.
4. Integrasi dengan Tomcat: Deploy file konfigurasi Mondrian ke server Apache Tomcat untuk menjalankan dan mengakses cube. Tomcat akan menyediakan antarmuka untuk meng-query dan menampilkan data cube.

### **2.1.5. Perancangan Dashboard**

Setelah data tersedia dan dapat diakses melalui cube, tahap berikutnya adalah pembangunan dashboard. Dashboard ini akan menggunakan PHP dan Charting Libraries seperti Highcharts untuk membuat visualisasi yang interaktif dan informatif dari data yang telah dianalisis. Proses pembuatan dashboard meliputi:

1. Desain Antarmuka Pengguna (UI): Mendesain dashboard yang ramah pengguna, menampilkan berbagai visualisasi data seperti grafik batang, garis, dan pie chart berdasarkan ukuran (measure) yang ingin ditampilkan.
2. Penghubungan dengan Cube: Menghubungkan dashboard dengan Mondrian Cube untuk menampilkan data yang telah diproses dan disusun dalam format OLAP.

## BAB III HASIL DAN PENGERJAAN

### 3.1. Implementasi Model Multidimensi

#### 3.1.1. Skema dw\_sales

Pada tahap perancangan skema, kami menentukan model Star Schema untuk mendesain data warehouse. Dalam model ini, database dw\_sales memiliki beberapa tabel dimensi yang mendukung analisis data. Berikut adalah tabel dimensi dari dw\_sales:

##### 1. Dimensi Customer (dimcustomer)

Pada tabel dimensi Customer kolom yang digunakan adalah CustomerID, CustomerName, dan CustomerType (I = Individual, S = Store).

- Query pengisian data:

```
INSERT INTO dimcustomer (CustomerID, CustomerName, CustomerType)
SELECT
    sc.CustomerID AS CustomerID,
    CONCAT_WS(' ', pp.FirstName, pp.MiddleName, pp.LastName) AS CustomerName,
    CASE
        WHEN pp.PersonType = 'IN' THEN 'I'
        WHEN pp.PersonType = 'SC' THEN 'S'
    END AS CustomerType
FROM
    adventureworks.sales_customer sc
JOIN
    adventureworks.person_person pp ON sc.CustomerID = pp.BusinessEntityID
WHERE
    pp.PersonType IN ('IN', 'SC');
```

- Hasil:

	CustomerID	CustomerName	CustomerType
1	11,000	Jon V Yang	
2	11,001	Eugene L Huang	
3	11,002	Ruben Torres	
4	11,003	Christy Zhu	
5	11,004	Elizabeth Johnson	
6	11,005	Julio Ruiz	
7	11,006	Janet G Alvarez	
8	11,007	Marco Mehta	
9	11,008	Rob Verhoff	
10	11,009	Shannon C Carlson	
11	11,010	Jacquelyn C Suarez	
12	11,011	Curtis Lu	
13	11,012	Lauren M Walker	
14	11,013	Ian M Jenkins	
15	11,014	Sydney Bennett	

## 2. Dimensi Employee (dimemployee)

Pada tabel dimensi Employee kolom yang digunakan adalah EmployeeID EmployeeID, EmployeeName, MaritalStatus , Gender, dan Age

- Query pengisian data:

```
INSERT INTO dimemployee (EmployeeID, EmployeeName, MaritalStatus, Gender, Age)
SELECT
    sp.BusinessEntityID AS EmployeeID,
    CONCAT_WS(' ', pp.FirstName, pp.MiddleName, pp.LastName) AS EmployeeName,
    he.MaritalStatus,
    he.Gender,
    DATEDIFF(YEAR, he.BirthDate, '2001-01-01') AS Age
FROM
    adventureworks.sales_salesperson sp
JOIN
    adventureworks.humanresources_employee he ON sp.BusinessEntityID = he.BusinessEntityID
JOIN
    adventureworks.person_person pp ON he.BusinessEntityID = pp.BusinessEntityID;
```

- Hasil:

	EmployeeID	EmployeeName	MaritalStatus	Gender	Age
1	274	Stephen Y Jiang	M	M	49
2	275	Michael G Blythe	S	M	32
3	276	Linda C Mitchell	M	F	20
4	277	Jillian Carson	S	F	38
5	278	Garrett R Vargas	M	M	25
6	279	Tsvi Michael Reiter	M	M	26
7	280	Pamela O Anzman-Wolfe	S	F	26
8	281	Shu K Ito	M	M	32
9	282	José Edvaldo Saraiva	M	M	37
10	283	David R Campbell	S	M	26
11	284	Tete A Mensa-Annan	M	M	22
12	285	Syed E Abbas	M	M	25
13	286	Lynn N Tsolfias	S	F	23
14	287	Amy E Alberts	M	F	43
15	288	Rachel B Valdez	S	F	25
16	289	Jae B Pak	M	F	32
17	290	Ranjit R Varkey Chudukatil	S	M	25

### 3. Dimensi Product (dimproduct)

Pada tabel dimensi Product kolom yang digunakan adalah ProductID, ProductName, ProductCategory, ProductSubCategory, dan ProductColor.

- Query pengisian data:

```
1 INSERT INTO dimproduct (ProductID, ProductName, ProductCategory, ProductSubCategory, ProductColor)
2 SELECT
3     p.ProductID,
4     p.Name AS ProductName,
5     pc.Name AS ProductCategory,
6     psc.Name AS ProductSubCategory,
7     p.Color AS ProductColor
8 FROM
9     adventureworks.production_product p
10 JOIN
11     adventureworks.production_productsubcategory psc ON p.ProductSubcategoryID = psc.ProductSubcategoryID
12 JOIN
13     adventureworks.production_productcategory pc ON psc.ProductCategoryID = pc.ProductCategoryID;
```

- Hasil:

	ProductID	ProductName	ProductCategory	ProductSubCategory	ProductColor
1	680	HL Road Frame - Black, 58	Components	Road Frames	Black
2	706	HL Road Frame - Red, 58	Components	Road Frames	Red
3	707	Sport-100 Helmet, Red	Accessories	Helmets	Red
4	708	Sport-100 Helmet, Black	Accessories	Helmets	Black
5	709	Mountain Bike Socks, M	Clothing	Socks	White
6	710	Mountain Bike Socks, L	Clothing	Socks	White
7	711	Sport-100 Helmet, Blue	Accessories	Helmets	Blue
8	712	AWC Logo Cap	Clothing	Caps	Multi
9	713	Long-Sleeve Logo Jersey, S	Clothing	Jerseys	Multi
10	714	Long-Sleeve Logo Jersey, M	Clothing	Jerseys	Multi
11	715	Long-Sleeve Logo Jersey, L	Clothing	Jerseys	Multi
12	716	Long-Sleeve Logo Jersey, XL	Clothing	Jerseys	Multi
13	717	HL Road Frame - Red, 62	Components	Road Frames	Red
14	718	HL Road Frame - Red, 44	Components	Road Frames	Red
15	719	HL Road Frame - Red, 48	Components	Road Frames	Red

### 4. Dimensi Sales Territory (dimsaleterritory)

Pada tabel dimensi SalesTerritory kolom yang digunakan adalah TerritoryID, TerritoryName, CountryRegionCode, dan Group

- Query pengisian data:

```
INSERT INTO dimsaleterritory (TerritoryID, TerritoryName, CountryRegionCode, `Group`)
SELECT
    TerritoryID,
    Name AS TerritoryName,
    CountryRegionCode,
    `group`
FROM
    adventureworks.sales_territory;
```

- Hasil:

	TerritoryID	TerritoryName	CountryRegionCode	Group
1	1	Northwest	US	North America
2	2	Northeast	US	North America
3	3	Central	US	North America
4	4	Southwest	US	North America
5	5	Southeast	US	North America
6	6	Canada	CA	North America
7	7	France	FR	Europe
8	8	Germany	DE	Europe
9	9	Australia	AU	Pacific
10	10	United Kingdom	GB	Europe

## 5. Dimensi Time (dimtime)

Pada tabel dimensi Time kolom yang digunakan adalah TimeID, Tahun, Bulan, Tanggal, TanggalLkp, HariEnglish, HariIndo, dan HariJawa

- Query procedure:

```

BEGIN
    mulailoop: LOOP
        INSERT INTO time(Tahun,Bulan,Tanggal,TanggalLkp,HariEnglish,HariIndo,HariJawa)
        VALUES (
            YEAR(waktuMulai),
            MONTH(waktuMulai),
            DAY(waktuMulai),
            waktuMulai,
            DAYNAME(waktuMulai),
            HariIndo,
            HariJawa
        );
        SET waktuMulai = DATE_ADD(waktuMulai, INTERVAL 1 DAY);

        SET HariJawa = CASE
            WHEN HariJawa = 'Wage' THEN 'Kliwon'
            WHEN HariJawa = 'Kliwon' THEN 'Legi'
            WHEN HariJawa = 'Legi' THEN 'Pahing'
            WHEN HariJawa = 'Pahing' THEN 'Pon'
            WHEN HariJawa = 'Pon' THEN 'Wage'
            ELSE HariJawa -- menambahkan ELSE untuk keamanan
        END;

        SET HariIndo = CASE
            WHEN HariIndo = 'Sabtu' THEN 'Minggu'
            WHEN HariIndo = 'Minggu' THEN 'Senin'
            WHEN HariIndo = 'Senin' THEN 'Selasa'
            WHEN HariIndo = 'Selasa' THEN 'Rabu'
            WHEN HariIndo = 'Rabu' THEN 'Kamis'
            WHEN HariIndo = 'Kamis' THEN 'Jumat'
            WHEN HariIndo = 'Jumat' THEN 'Sabtu'
            ELSE HariIndo -- menambahkan ELSE untuk keamanan
        END;

        IF DATEDIFF(waktuMulai, waktuSelesai) > 0 THEN
            LEAVE mulailoop;
        END IF;

    END LOOP mulailoop;
END

```

- Query pengisian data:

```
CALL Isitime('2011-04-01', '2014-06-24', 'Wage', 'Sabtu');
```

- Hasil:

	TimeID	Tahun	Bulan	Tanggal	TanggalKp	HariEnglish	HariIndo	HariJawa
1	1,128	2,011	1	1	2011-01-01	Saturday	Sabtu	Wage
2	1,129	2,011	1	2	2011-01-02	Sunday	Minggu	Kliwon
3	1,130	2,011	1	3	2011-01-03	Monday	Senin	Legi
4	1,131	2,011	1	4	2011-01-04	Tuesday	Selasa	Pahing
5	1,132	2,011	1	5	2011-01-05	Wednesday	Rabu	Pon
6	1,133	2,011	1	6	2011-01-06	Thursday	Kamis	Wage
7	1,134	2,011	1	7	2011-01-07	Friday	Jumat	Kliwon
8	1,135	2,011	1	8	2011-01-08	Saturday	Sabtu	Legi
9	1,136	2,011	1	9	2011-01-09	Sunday	Minggu	Pahing
10	1,137	2,011	1	10	2011-01-10	Monday	Senin	Pon
11	1,138	2,011	1	11	2011-01-11	Tuesday	Selasa	Wage
12	1,139	2,011	1	12	2011-01-12	Wednesday	Rabu	Kliwon
13	1,140	2,011	1	13	2011-01-13	Thursday	Kamis	Legi
14	1,141	2,011	1	14	2011-01-14	Friday	Jumat	Pahing
15	1,142	2,011	1	15	2011-01-15	Saturday	Sabtu	Pon

## 6. Tabel Fakta (fact\_sales)

Pada tabel TabelFact kolom yang digunakan adalah SalesID, ProductID, TimeID, EmployeeID, CustomerID, TerritoryID, SalesAmount, Quantity.

- Narasi Kalimat:

Customer dengan id 101 (SalesID/SalesOrderID) bernama Rahmad (CustomerID) berada di wilayah Asia (SalesTerritory) membeli Jersey (ProductID) sebanyak 10 pasang (measure Quantity) pada tanggal 25 Oktober 2024 (TimeID), melalui sales bernama Budi (EmployeeID) Dengan total transaksi sebesar Rp1.500.000 (measure TotalAmount)

- Query pengisian data:

```

SELECT
    SOH.SalesOrderID AS SalesID,
    SOD.ProductID AS ProductID,
    SOH.CustomerID AS CustomerID,
    t.TimeID,
    SOH.SalesPersonID AS EmployeeID,
    SOH.TerritoryID AS TerritoryID,
    SOD.OrderQty AS Quantity,
    SOD.UnitPrice AS UnitPrice,
    SOD.UnitPrice * SOD.OrderQty AS SalesAmount
FROM
    adventureworks.sales_salesorderheader SOH
JOIN
    adventureworks.sales_salesorderdetail SOD ON SOH.SalesOrderID = SOD.SalesOrderID
JOIN
    dw_sales.dimtime t ON DATE(SOH.ShipDate) = t.TanggalKp;

```

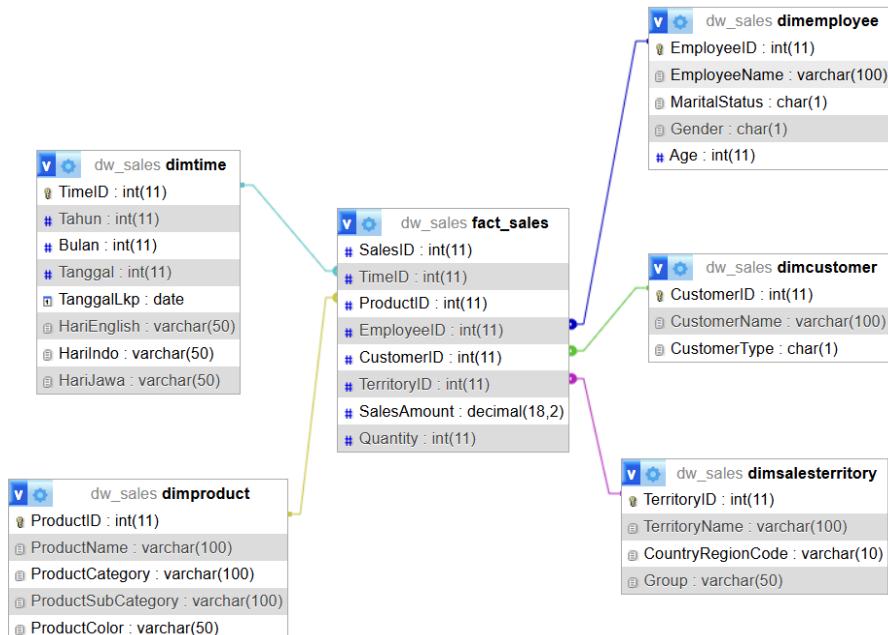
- ETL:



- Hasil:

	SalesID	TimeID	ProductID	EmployeeID	CustomerID	TerritoryID	SalesAmount	Quantity
1	43.659	1,285	776	279	29,825	5	2,024.99	1
2	43.659	1,285	777	279	29,825	5	6,074.98	3
3	43.659	1,285	778	279	29,825	5	2,024.99	1
4	43.659	1,285	771	279	29,825	5	2,039.99	1
5	43.659	1,285	772	279	29,825	5	2,039.99	1
6	43.659	1,285	773	279	29,825	5	4,079.99	2
7	43.659	1,285	774	279	29,825	5	2,039.99	1
8	43.659	1,285	714	279	29,825	5	86.52	3
9	43.659	1,285	716	279	29,825	5	28.84	1
10	43.659	1,285	709	279	29,825	5	34.2	6
11	43.659	1,285	712	279	29,825	5	10.37	2
12	43.659	1,285	711	279	29,825	5	80.75	4
13	43.660	1,285	762	279	29,672	5	419.46	1
14	43.660	1,285	758	279	29,672	5	874.79	1
15	43.661	1,285	745	282	29,734	6	809.76	1

Setelah membuat tabel dimensi dan melakukan ekstraksi data untuk mengisi tabel fact\_sales, berikut adalah diagram ERD yang menggambarkan hubungan antara tabel fakta dan dimensi:



### 3.1.2. Skema dw\_purchasing

Pada tahap perancangan skema, kami menentukan model Star Schema untuk mendesain data warehouse. Dalam model ini, database dw\_purchasing memiliki beberapa tabel dimensi yang mendukung analisis data. Berikut adalah tabel dimensi dari dw\_purchasing:

#### 1. Dimensi Warehouse (dimwarehouse)

Pada tabel dimensi Warehouse kolom yang digunakan adalah WarehouseID dan Location.

- Query pengisian data:

```
1 insert ignore into dimwarehouse (WarehouseID, Location)
2 SELECT
3     ppi.LocationID AS WarehouseID,
4     ppl.Name AS Location
5 FROM
6     adventureworks.production_productinventory ppi
7 JOIN
8     adventureworks.production_location ppl
9     ON ppi.LocationID = ppl.LocationID;
```

- Hasil:

	WarehouseID	Location
1	1	Tool Crib
2		Sheet Metal Racks
3		Paint Shop
4		Paint Storage
5		Metal Storage
6		Miscellaneous Storage
7		Finished Goods Storage
8		Frame Forming
9		Frame Welding
10		Debur and Polish
11		Paint
12		Specialized Paint
13		Subassembly
14		Final Assembly

#### 2. Dimensi Employee (dimemployee)

Pada tabel dimensi Employee kolom yang digunakan adalah EmployeeID, EmployeeName, MaritalStatus, Gender, dan Age

- Query pengisian data:

```

1 INSERT INTO dimemployee (EmployeeID, EmployeeName, JobTitle, MaritalStatus, Gender, Age)
2 SELECT
3     p.BusinessEntityID AS EmployeeID,
4     CONCAT_WS(' ', p.FirstName, p.MiddleName, p.LastName) AS EmployeeName,
5     e.JobTitle,
6     e.MaritalStatus,
7     e.Gender,
8     DATEDIFF(YEAR, e.BirthDate, '2001-01-01') AS Age
9 FROM
10    adventureworks.person_person p
11   JOIN
12      adventureworks.humanresources_employee e ON p.BusinessEntityID = e.BusinessEntityID
13   JOIN
14      adventureworks.humanresources_employeedepartmenthistory edh ON e.BusinessEntityID = edh.BusinessEntityID
15   JOIN
16      adventureworks.humanresources_department d ON edh.DepartmentID = d.DepartmentID
17 WHERE
18     d.Name = 'Purchasing'
19     AND p.PersonType IN ('EM')
20     AND e.JobTitle IN ('Buyer', 'Purchasing Manager', 'Purchasing Assistant');

```

- Hasil:

	EmployeeID	EmployeeName	JobTitle	MaritalStatus	Gender	Age
1	250	Sheela H Word	Purchasing Manager	S	F	22
2	251	Mikael Q Sandberg	Buyer	S	M	16
3	252	Arvind B Rao	Buyer	M	M	26
4	253	Linda P Meisner	Buyer	M	F	30
5	254	Fukiko J Ogisu	Buyer	M	M	30
6	255	Gordon L Hee	Buyer	M	M	34
7	256	Frank S Pellow	Buyer	M	M	48
8	257	Eric S Kurjan	Buyer	S	M	28
9	258	Erin M Hagens	Buyer	S	F	29
10	259	Ben T Miller	Buyer	M	M	27
11	260	Annette L Hill	Purchasing Assistant	M	F	22
12	261	Reinout N Hillmann	Purchasing Assistant	M	M	22

### 3. Dimensi Product (dimproduct)

Pada tabel dimensi Product kolom yang digunakan adalah ProductID, ProductName, ProductCategory, ProductSubCategory, ProductColor, dan UnitPrice.

- Query pengisian data:

```

1 INSERT IGNORE INTO dimproduct (ProductID, ProductName, ProductCategory, ProductSubCategory, ProductColor, UnitPrice)
2 SELECT
3     pod.ProductID,
4     p.Name AS ProductName,
5     pc.Name AS ProductCategory,
6     psc.Name AS ProductSubCategory,
7     p.Color AS ProductColor,
8     pod.UnitPrice
9 FROM
10    adventureworks.purchasing_purchaseorderdetail pod
11   JOIN
12      adventureworks.production_product p ON pod.ProductID = p.ProductID
13   JOIN
14      adventureworks.production_productsubcategory psc ON p.ProductSubcategoryID = psc.ProductSubcategoryID
15   JOIN
16      adventureworks.production_productcategory pc ON psc.ProductCategoryID = pc.ProductCategoryID;

```

- Hasil:

	ProductID	ProductName	ProductCategory	ProductSubCategory	ProductColor	UnitPrice
1	707	Sport-100 Helmet, Red	Accessories	Helmets	Red	13.09
2	708	Sport-100 Helmet, Black	Accessories	Helmets	Black	13.09
3	709	Mountain Bike Socks, M	Clothing	Socks	White	3.4
4	710	Mountain Bike Socks, L	Clothing	Socks	White	3.4
5	711	Sport-100 Helmet, Blue	Accessories	Helmets	Blue	13.08
6	712	AWC Logo Cap	Clothing	Caps	Multi	6.92
7	713	Long-Sleeve Logo Jersey, S	Clothing	Jerseys	Multi	38.49
8	714	Long-Sleeve Logo Jersey, M	Clothing	Jerseys	Multi	38.49
9	715	Long-Sleeve Logo Jersey, L	Clothing	Jerseys	Multi	38.49
10	716	Long-Sleeve Logo Jersey, XL	Clothing	Jerseys	Multi	38.75
11	841	Men's Sports Shorts, S	Clothing	Shorts	Black	24.75
12	842	Touring-Panniers, Large	Accessories	Panniers	Grey	51.56
13	843	Cable Lock	Accessories	Locks	[NULL]	10.31

#### 4. Dimensi Shipping Method (dimshippingmethod)

Pada tabel dimensi Shipping Method kolom yang digunakan adalah ShippingMethodID dan ShippingMethodName.

- Query pengisian data:

```

1 INSERT INTO dimshippingmethod (ShippingMethodName)
2 SELECT
3     sm.Name AS ShippingMethodName
4 FROM
5     adventureworks.purchasing_shipmethod sm;|
```

- Hasil:

	ShippingMethodID	ShippingMethodName
1	1	CARGO TRANSPORT 5
2		OVERNIGHT J-FAST
3		OVERSEAS - DELUXE
4		XRQ - TRUCK GROUND
5		ZY - EXPRESS

#### 5. Dimensi Supplier (dimsupplier)

Pada tabel dimensi Supplier kolom yang digunakan adalah VendorID, VendorName, ContactPhone, dan Address

- Query pengisian data:

```

1 INSERT INTO dimsupplier (VendorID, VendorName, ContactPhone, Address)
2 SELECT
3     v.BusinessEntityID AS VendorID,
4     v.Name AS VendorName,
5     MAX(pp.PhoneNumber) AS ContactPhone,
6     CONCAT(a.AddressLine1, ' ', a.City, ' ', a.PostalCode) AS Address
7 FROM
8     adventureworks.purchasing_vendor v
9 INNER JOIN
10    adventureworks.person_businessentityaddress bea ON bea.BusinessEntityID = v.BusinessEntityID
11 INNER JOIN
12    adventureworks.person_address a ON a.AddressID = bea.AddressID
13 INNER JOIN
14    adventureworks.person_stateprovince sp ON sp.StateProvinceID = a.StateProvinceID
15 INNER JOIN
16    adventureworks.person_countryregion cr ON cr.CountryRegionCode = sp.CountryRegionCode
17 INNER JOIN
18    adventureworks.person_addresstype at ON at.AddressTypeID = bea.AddressTypeID
19 LEFT JOIN
20    adventureworks.person_businessentitycontact bec ON bec.BusinessEntityID = v.BusinessEntityID
21 LEFT JOIN
22    adventureworks.person_person p ON p.BusinessEntityID = bec.PersonID
23 LEFT JOIN
24    adventureworks.person_personphone pp ON pp.BusinessEntityID = p.BusinessEntityID
25 LEFT JOIN
26    adventureworks.person_phonenumbertype pnt ON pnt.PhoneNumberTypeID = pp.PhoneNumberTypeID
27 GROUP BY
28     v.BusinessEntityID, v.Name, a.AddressLine1, a.City, a.PostalCode;

```

- Hasil:

	123 VendorID	az VendorName	az ContactPhone	az Address
1	1,492	Australia Bike Retailer	816-555-0100	28 San Marino Ct, Bellingham, 98225
2	1,494	Allenson Cycles	530-555-0100	4659 Montoya, Altadena, 91001
3	1,496	Advanced Bicycles	859-555-0100	7995 Edwards Ave., Lynnwood, 98036
4	1,498	Trikes, Inc.	315-555-0100	90 Sunny Ave, Berkeley, 94704
5	1,500	Morgan Bike Accessories	142-555-0100	9098 Story Lane, Albany, 12210
6	1,502	Cycling Master	920-555-0100	4823 Stonewood Ct, Walla Walla, 99362
7	1,504	Chicago Rent-All	879-555-0100	15 Pear Dr, Newport Beach, 92625
8	1,506	Greenwood Athletic Company	959-555-0100	6441 Co Road, Lemon Grove, 85252
9	1,508	Compete Enterprises, Inc	516-555-0100	50 Via Del Sol, Lynnwood, 98036
10	1,510	International	432-555-0100	683 Larch Ct, Salt Lake City, 84101
11	1,512	Light Speed	114-555-0100	298 Sunnybrook Drive, Spring Valley, 91977
12	1,514	Training Systems	945-555-0100	6 Dancing Road, Burien, 98168
13	1,516	Gardner Touring Cycles	882-555-0100	8513 Hurlstone Ct., Altadena, 91001
14	1,518	International Trek Center	439-555-0100	8844 Garcia, West Covina, 91791
15	1,520	G & K Bicycle Corp.	235-555-0100	8981 Carmel Drive, W. Linn, 89701

## 6. Dimensi Time (dimdate)

Pada tabel dimensi Time kolom yang digunakan adalah DateID, Tahun, Bulan, Tanggal, TanggalLkp, HariEnglish, HariIndo, dan HariJawa

- Query procedure:

```

BEGIN
    mulailoop: LOOP
        INSERT INTO dimtime(Tahun,Bulan,Tanggal,TanggalLkp,HariEnglish,HariIndo,HariJawa)
        VALUES (
            YEAR(waktuMulai),
            MONTH(waktuMulai),
            DAY(waktuMulai),
            waktuMulai,
            DAYNAME(waktuMulai),
            HariIndo,
            HariJawa
        );
        SET waktuMulai = DATE_ADD(waktuMulai, INTERVAL 1 DAY);

        SET HariJawa = CASE
            WHEN HariJawa = 'Wage' THEN 'Kliwon'
            WHEN HariJawa = 'Kliwon' THEN 'Legi'
            WHEN HariJawa = 'Legi' THEN 'Pahing'
            WHEN HariJawa = 'Pahing' THEN 'Pon'
            WHEN HariJawa = 'Pon' THEN 'Wage'
            ELSE HariJawa -- menambahkan ELSE untuk keamanan
        END;

        SET HariIndo = CASE
            WHEN HariIndo = 'Sabtu' THEN 'Minggu'
            WHEN HariIndo = 'Minggu' THEN 'Senin'
            WHEN HariIndo = 'Senin' THEN 'Selasa'
            WHEN HariIndo = 'Selasa' THEN 'Rabu'
            WHEN HariIndo = 'Rabu' THEN 'Kamis'
            WHEN HariIndo = 'Kamis' THEN 'Jumat'
            WHEN HariIndo = 'Jumat' THEN 'Sabtu'
            ELSE HariIndo -- menambahkan ELSE untuk keamanan
        END;

        IF DATEDIFF(waktuMulai, waktuSelesai) > 0 THEN
            LEAVE mulailoop;
        END IF;

    END LOOP mulailoop;
END

```

- Query pengisian data:

```
CALLisitime('2011-04-01', '2014-06-24', 'Wage', 'Sabtu');
```

- Hasil:

	DatetID	Tanggal	Bulan	Tahun	TanggalLkp	HariIndo	HariJawa	HariEnglish
1	1	1	4	2,011	2011-04-01	Sabtu	Wage	Friday
2	2	2	4	2,011	2011-04-02	Minggu	Kliwon	Saturday
3	3	3	4	2,011	2011-04-03	Senin	Legi	Sunday
4	4	4	4	2,011	2011-04-04	Selasa	Pahing	Monday
5	5	5	4	2,011	2011-04-05	Rabu	Pon	Tuesday
6	6	6	4	2,011	2011-04-06	Kamis	Wage	Wednesday
7	7	7	4	2,011	2011-04-07	Jumat	Kliwon	Thursday
8	8	8	4	2,011	2011-04-08	Sabtu	Legi	Friday
9	9	9	4	2,011	2011-04-09	Minggu	Pahing	Saturday
10	10	10	4	2,011	2011-04-10	Senin	Pon	Sunday
11	11	11	4	2,011	2011-04-11	Selasa	Wage	Monday
12	12	12	4	2,011	2011-04-12	Rabu	Kliwon	Tuesday
13	13	13	4	2,011	2011-04-13	Kamis	Legi	Wednesday
14	14	14	4	2,011	2011-04-14	Jumat	Pahing	Thursday
15	15	15	4	2,011	2011-04-15	Sabtu	Pon	Friday

## 7. Tabel Fakta (factpurchasing)

Pada tabel TabelFact kolom yang digunakan adalah PurchaseOrderID, ProductID, DateID, EmployeeID, WarehouseID, ShippingMethodID, VendorID, TotalCost, Qty.

- Narasi Kalimat:

Customer dengan id 101 (SalesID/SalesOrderID) bernama Rahmad (CustomerID) berada di wilayah Asia (SalesTerritory) membeli Jersey (ProductID) sebanyak 10 pasang (measure Quantity) pada tanggal 25 Oktober 2024 (TimeID), melalui sales bernama Budi (EmployeeID) Dengan total transaksi sebesar Rp1.500.000 (measure TotalAmount)

- Query pengisian data:

```
1 SELECT
2     SOH.SalesOrderID AS SalesID,
3     SOD.ProductID AS ProductID,
4     SOH.CustomerID AS CustomerID,
5     t.TimeID,
6     SOH.SalesPersonID AS EmployeeID,
7     SOH.TerritoryID AS TerritoryID,
8     SOD.OrderQty AS Quantity,
9     SOD.UnitPrice AS UnitPrice,
10    SOD.UnitPrice * SOD.OrderQty AS SalesAmount
11   FROM
12      adventureworks2.sales_salesorderheader SOH
13  JOIN
14      adventureworks2.sales_salesorderdetail SOD ON SOH.SalesOrderID = SOD.SalesOrderID
15  JOIN
16      dwadw24.dimtime t ON DATE(SOH.ShipDate) = t.TanggalLkp;
```

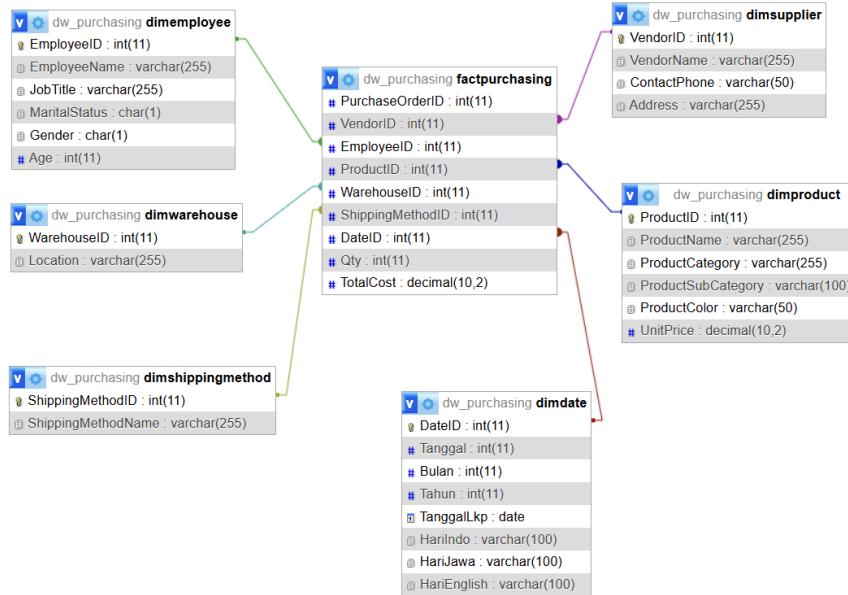
- ETL:



- Hasil:

	PurchaseOrderID	VendorID	EmployeeID	ProductID	WarehouseID	ShippingMethodID	DateID	Qty	TotalCost
1	67	1,632	255	928	6	5	308	550	18,023,78
2	67	1,632	255	928	50	5	308	550	18,023,78
3	67	1,632	255	928	6	5	308	550	18,023,78
4	67	1,632	255	928	50	5	308	550	18,023,78
5	67	1,632	255	929	6	5	308	550	20,397,3
6	67	1,632	255	929	50	5	308	550	20,397,3
7	67	1,632	255	929	6	5	308	550	20,397,3
8	67	1,632	255	929	50	5	308	550	20,397,3
9	67	1,632	255	930	6	5	308	550	23,671,73
10	67	1,632	255	930	50	5	308	550	23,671,73
11	67	1,632	255	930	6	5	308	550	23,671,73
12	67	1,632	255	930	50	5	308	550	23,671,73
13	62	1,684	254	931	6	5	308	550	19,178,78
14	65	1,588	251	931	6	5	308	550	18,890,03
15	62	1,684	254	931	50	5	308	550	19,178,78
16	65	1,588	251	931	50	5	308	550	18,890,03
17	62	1,684	254	931	6	5	308	550	19,178,78
18	65	1,588	251	931	6	5	308	550	18,890,03
19	62	1,684	254	931	50	5	308	550	19,178,78
20	65	1,588	251	931	50	5	308	550	18,890,03

Setelah membuat tabel dimensi dan melakukan ekstraksi data untuk mengisi tabel factpurchasing, berikut adalah diagram ERD yang menggambarkan hubungan antara tabel fakta dan dimensi:



## 3.2. Implementasi Cube dan OLAP

### 3.2.1. Cube dan OLAP (dw\_sales)

Berikut adalah Query Cube pada Sales.xml:

```
<Cube name="Sales" defaultMeasure="Amount">
    <Table name="fact_sales"/>
```

Berikut adalah Query Cube pada Sales.jsp:

```
<%@ page session="true" contentType="text/html; charset=ISO-8859-1" %>
<%@ taglib uri="http://www.tonbeller.com/jpivot" prefix="jp" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

<jp:mondrianQuery id="query01" jdbcDriver="com.mysql.jdbc.Driver"
    jdbcUrl="jdbc:mysql://localhost/dw_sales?user=root&password"
    catalogUri="/WEB-INF/queries/sales.xml">

    select {[Measures].[SalesAmount],[Measures].[Quantity]} ON COLUMNS,
    {[Time],[Customer],[Employee],[Product],[Territory]} ON ROWS

    from [Sales]

</jp:mondrianQuery>

<c:set var="title01" scope="session">Query SALES using Mondrian OLAP</c:set>
```

## 1. fact\_sales (Menganalisis Dimensi Time)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_sales) dengan dimensi Time.

- Query pengisian data:

```
<Dimension name="Time" primaryKey="TimeID">
    <Hierarchy hasAll="true" allMemberName="All Times" primaryKey="TimeID">
        <Table name="dimtime" />

        <!-- Menentukan level tahun sebagai "Years" -->
        <Level name="Year" column="tahun" type="Numeric" uniqueMembers="true" levelType="TimeYears"/>

        <!-- Menentukan level bulan sebagai "Months" -->
        <Level name="Month" column="bulan" uniqueMembers="false" type="Numeric" levelType="TimeMonths"/>

        <!-- Menentukan level tanggal sebagai "Days" -->
        <Level name="Date" column="tanggal" uniqueMembers="false" type="Numeric" levelType="TimeDays"/>

        <!-- Level tambahan untuk tanggal lookup dan hari -->
        <!-- <Level name="Tanggallkp" column="Tanggallkp" uniqueMembers="false" type="String" levelType="Regular"/>
        <Level name="HariEnglish" column="HariEnglish" uniqueMembers="false" type="String" levelType="Regular"/>
        <Level name="HariIndo" column="HariIndo" uniqueMembers="false" type="String" levelType="Regular"/>
        <Level name="HariJawa" column="HariJawa" uniqueMembers="false" type="String" levelType="Regular"/> -->
    </Hierarchy>
</Dimension>
```

- Hasil

Query SALES using Mondrian OLAP



Time	Customer	Employee	Product	Territory	Measures
					TotalAmount
					Quantity
All Times	All Customers	All Employee	All Product	All Territory	220,747,779
2011	All Customers	All Employee	All Product	All Territory	25,064,844
2012	All Customers	All Employee	All Product	All Territory	62,686,317
1	All Customers	All Employee	All Product	All Territory	4,975,540
2	All Customers	All Employee	All Product	All Territory	4,123,622
3	All Customers	All Employee	All Product	All Territory	3,039,930
4	All Customers	All Employee	All Product	All Territory	6,007,504
5	All Customers	All Employee	All Product	All Territory	3,555,027
6	All Customers	All Employee	All Product	All Territory	5,883,873
7	All Customers	All Employee	All Product	All Territory	8,284,755
8	All Customers	All Employee	All Product	All Territory	6,597,590
9	All Customers	All Employee	All Product	All Territory	4,399,328
10	All Customers	All Employee	All Product	All Territory	6,804,458
11	All Customers	All Employee	All Product	All Territory	5,449,939
12	All Customers	All Employee	All Product	All Territory	3,564,751
2013	All Customers	All Employee	All Product	All Territory	86,734,490
2014	All Customers	All Employee	All Product	All Territory	46,262,128
					138,284

## 2. fact\_sales (Menganalisis Dimensi Customer)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_sales) dengan dimensi Customer.

- Query pengisian data:

```
<Dimension name="Customer" primaryKey="CustomerID">
    <Hierarchy hasAll="true" allMemberName="All Customers" primaryKey="CustomerID">
        <Table name="dimcustomer" />

        <!-- Level 1: Detail pelanggan -->
        <Level name="Customer Name" column="CustomerName"/>
        <!-- Properti tambahan pelanggan -->
        <Level name="Customer Type" column="CustomerType"/>
    </Hierarchy>
</Dimension>
```

- Hasil

### Query SALES using Mondrian OLAP



					Measures	
Time	Customer	Employee	Product	Territory	TotalAmount	Quantity
+All Times	-All Customers	+All Employee	+All Product	+All Territory	220,747,779	549,828
	+I	+All Employee	+All Product	+All Territory	58,717,354	120,796
	+S	+All Employee	+All Product	+All Territory	162,030,424	429,032

					Measures	
Time	Customer	Employee	Product	Territory	TotalAmount	Quantity
+All Times	-All Customers	+All Employee	+All Product	+All Territory	220,747,779	549,828
	-I	+All Employee	+All Product	+All Territory	58,717,354	120,796
	Aaron A Allen	+All Employee	+All Product	+All Territory	6,800	2
	Aaron A Hayes	+All Employee	+All Product	+All Territory	6,226	6
	Aaron A Zhang	+All Employee	+All Product	+All Territory	1,201	8
	Aaron Alexander	+All Employee	+All Product	+All Territory	140	2

### 3. fact\_sales (Menganalisis Dimensi Employee)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_sales) dengan dimensi Employee.

- Query pengisian data:

```
<Dimension name="Employee" primaryKey="EmployeeID">
    <Hierarchy hasAll="true" allMemberName="All Employee" primaryKey="EmployeeID">
        <Table name="dimemployee"/>

        <!-- Level 1: Nama karyawan -->
        <Level name="Employee Name" column="EmployeeName"/>

        <!-- Level 2: Status pernikahan -->
        <Level name="Marital Status" column="MaritalStatus"/>

        <!-- Level 3: Jenis kelamin -->
        <Level name="Gender" column="Gender"/>

        <!-- Level 4: Usia -->
        <Level name="Age" column="Age"/>
    </Hierarchy>
</Dimension>
```

- Hasil

					Measures	
Time	Customer	Employee	Product	Territory	TotalAmount	Quantity
+All Times	+All Customers	-All Employee	+All Product	+All Territory	220,747,779	549,828
		-Amy E Alberts	+All Product	+All Territory	1,476,505	4,024
		-M	+All Product	+All Territory	1,476,505	4,024
		-F	+All Product	+All Territory	1,476,505	4,024
		43	+All Product	+All Territory	1,476,505	4,024
		+David R Campbell	+All Product	+All Territory	7,513,723	16,344
		+Garrett R Vargas	+All Product	+All Territory	7,263,500	23,088
		+Jae B Pak	+All Product	+All Territory	17,083,629	52,462
		+Jillian Carson	+All Product	+All Territory	20,190,082	54,102
		+José Edvaldo Saraiva	+All Product	+All Territory	11,929,043	30,440
		+Linda C Mitchell	+All Product	+All Territory	20,889,252	54,458
		+Lynn N Tsfilas	+All Product	+All Territory	2,889,125	8,246
		+Michael G Blythe	+All Product	+All Territory	18,648,274	46,116
		+Pamela O Ansman-Wolfe	+All Product	+All Territory	6,679,427	14,720
		+Rachel B Valdez	+All Product	+All Territory	3,708,661	14,066

#### 4. fact\_sales (Menganalisis dimensi Product)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_sales) dengan dimensi Product.

- Query pengisian data:

```
<Dimension name="Product" primaryKey="ProductID">
    <Hierarchy hasAll="true" allMemberName="All Product" primaryKey="ProductID">
        <Table name="dimproduct"/>

        <!-- Level 2: Kategori Produk -->
        <Level name="Product Category" column="ProductCategory"/>

        <!-- Level 3: Subkategori Produk -->
        <Level name="Product SubCategory" column="ProductSubCategory"/>

        <!-- Level 1: Detail Produk -->
        <Level name="Product Name" column="ProductName"/>

        <!-- Level 3: Subkategori Produk -->
        <Level name="Product Color" column="ProductColor"/>
    </Hierarchy>
</Dimension>
```

- Hasil

The screenshot shows the Mondrian OLAP interface at [localhost:8080/mondrian/testpage.jsp](http://localhost:8080/mondrian/testpage.jsp). The title bar says "Query SALES using Mondrian OLAP". The interface includes a toolbar with various icons for MDX, cube selection, and data visualization. Below the toolbar is a hierarchical tree view with nodes like "Time", "Customer", "Employee", "Product", and "Territory". To the right is a data grid with columns for Time, Customer, Employee, Product, Territory, TotalAmount, and Quantity. The data grid contains numerous rows of sales data, such as "Mountain-100 Black, 38" and "Mountain-200 Silver, 38", with their respective territory and sales figures.

Time	Customer	Employee	Product	Territory	Measures
▪ All Times	▪ All Customers	▪ All Employee	▪ All Product	▪ All Territory	▪ TotalAmount ▪ Quantity
			▪ Accessories	▪ All Territory	220,747,779 549,828
			- Bikes	▪ All Territory	2,557,522 123,864
			- Mountain Bikes	▪ All Territory	190,291,627 180,536
			- Mountain-100 Black, 38	▪ All Territory	73,244,593 56,642
			▪ Mountain-100 Black	▪ All Territory	2,724,157 1,364
			▪ Mountain-100 Black, 42	▪ All Territory	2,545,822 1,268
			▪ Mountain-100 Black, 44	▪ All Territory	2,767,762 1,356
			▪ Mountain-100 Black, 48	▪ All Territory	2,504,580 1,232
			▪ Mountain-100 Silver, 38	▪ All Territory	2,622,412 1,284
			▪ Mountain-100 Silver, 42	▪ All Territory	2,405,629 1,186
			▪ Mountain-100 Silver, 44	▪ All Territory	2,465,469 1,202
			▪ Mountain-100 Silver, 48	▪ All Territory	2,063,114 1,010
			▪ Mountain-200 Black, 38	▪ All Territory	8,812,303 5,954
			▪ Mountain-200 Black, 42	▪ All Territory	8,028,136 5,328
			▪ Mountain-200 Black, 46	▪ All Territory	6,622,197 4,222
			▪ Mountain-200 Silver, 38	▪ All Territory	7,392,973 4,788

#### 5. fact\_sales (Menganalisis dimensi Territory)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_sales) dengan dimensi Territory.

- Query pengisian data:

```
<Dimension name="Territory" primaryKey="TerritoryID">
    <Hierarchy hasAll="true" allMemberName="All Territory" primaryKey="TerritoryID">
        <Table name="dimsalesterritory"/>
        <Level name="Territory ID" column="TerritoryID"/>
        <Level name="Group" column="Group"/>
        <Level name="Country Region Code" column="CountryRegionCode"/>
        <Level name="Territory Name" column="TerritoryName" />
        |   |   <!-- <Property name="Group Description" column="group_description" type="String" /> -->
    </Hierarchy>
</Dimension>
```

- Hasil

The screenshot shows a Mondrian OLAP cube visualization. At the top, there's a toolbar with icons for MDX, cube navigation, and other functions. Below the toolbar is a title "Query SALES using Mondrian OLAP". Underneath is a tree view of dimensions: Time, Customer, Employee, Product, and Territory. The Territory dimension is expanded, showing All Territory, Europe, DE, Germany, FR, GB, North America, and Pacific. The data grid below contains two columns: Measures (TotalAmount and Quantity) and their corresponding values for each territory level.

		Measures				
Time	Customer	Employee	Product	Territory	TotalAmount	Quantity
All Times	All Customers	All Employee	All Product	All Territory	220,747,779	549,828
				-Europe	39,880,342	106,296
				-DE	9,891,720	26,286
				Germany	9,891,720	26,286
				+FR	14,582,944	39,812
				+GB	15,405,678	40,198
				+North America	159,499,696	406,946
				+Pacific	21,367,740	36,586

Slicer:

### 3.2.2. Cube dan OLAP (dw\_purchasing)

Berikut adalah Query Cube pada purchase.xml:

```
<Cube name="purchase" defaultMeasure="Amount">
|   <Table name="factpurchasing"/>
```

Berikut adalah Query Cube pada purchase.jsp:

```
<%@ page session="true" contentType="text/html; charset=ISO-8859-1" %>
<%@ taglib uri="http://www.tonbeller.com/jpivot" prefix="jp" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

<jp:mondrianQuery id="query01" jdbcDriver="com.mysql.jdbc.Driver"
jdbcUrl="jdbc:mysql://localhost/dw_purchasing?user=root&password="
catalogUri="/WEB-INF/queries/purchase.xml">

    select {[Measures].[Total Cost],[Measures].[Quantity]} ON COLUMNS,
           {[Time],[Vendor],[Employee],[Product],[Shipping Method],[Warehouse]}) ON ROWS

    from [purchase]

</jp:mondrianQuery>
<c:set var="title01" scope="session">Query PURCHASING using Mondrian OLAP</c:set>
```

## 1. fact\_purchasing (Menganalisis Dimensi Time)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_purchasing) dengan dimensi Time.

- Query pengisian data:

```
<Cube name="purchase" defaultMeasure="Amount">
  <Table name="factpurchasing"/>

  <Dimension name="Time" foreignKey="DateID">
    <Hierarchy hasAll="true" allMemberName="All Times" primaryKey="DateID">
      <Table name="dimdate" />

      <!-- Menentukan level tahun sebagai "Years" -->
      <Level name="Year" column="tahun" type="Numeric" uniqueMembers="true" levelType="TimeYears"/>

      <!-- Menentukan level bulan sebagai "Months" -->
      <Level name="Month" column="bulan" uniqueMembers="false" type="Numeric" levelType="TimeMonths"/>

      <!-- Menentukan level tanggal sebagai "Days" -->
      <Level name="Date" column="tanggal" uniqueMembers="false" type="Numeric" levelType="TimeDays"/>

      <!-- Level tambahan untuk tanggal lookup dan hari -->
      <Level name="Tanggallkp" column="Tanggallkp" uniqueMembers="false" type="String" levelType="Regular"/>
      <Level name="HariEnglish" column="HariEnglish" uniqueMembers="false" type="String" levelType="Regular"/>
      <Level name="HariIndo" column="HariIndo" uniqueMembers="false" type="String" levelType="Regular"/>
      <Level name="HariJawa" column="HariJawa" uniqueMembers="false" type="String" levelType="Regular"/> ...
    </Hierarchy>
  </Dimension>
```

- Hasil

Query PURCHASING using Mondrian OLAP



Time	Vendor	Employee	Product	Shipping Method	Warehouse	Total Cost	Quantity
All Times	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	148,263,062	3,985,315
+2011	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	946,361	26,400
-2012	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	11,807,358	314,040
+1	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	1,240,799	34,650
+2	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	1,497,779	37,030
+3	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	2,152,199	57,750
+4	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	1,184,520	28,230
+5	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	165,361	6,600
+6	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	103,933	1,650
+7	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	1,886,087	57,750
+8	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	949,387	24,200
+9	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	1,317,524	33,180
+10	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	1,309,770	33,000
+11	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse		
+12	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse		
+2013	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	53,532,577	1,451,180
+2014	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	81,976,766	2,193,695

## 2. fact\_purchasing (Menganalisis Dimensi Supplier)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_purchasing) dengan dimensi Supplier.

- Query pengisian data:

```
<Dimension name="Vendor" primaryKey="VendorID">
    <Hierarchy hasAll="true" allMemberName="All Vendor" primaryKey="VendorID">
        <Table name="dimsupplier"/>

        <!-- Level 1: Detail Vendor -->
        <Level name="Vendor Name" column="VendorName"/>
        <!-- Level 2: Detail Vendor -->
        <Level name="Contact" column="ContactPhone"/>
        <!-- Level 3: Detail Vendor -->
        <Level name="Address" column="Address"/>
    </Hierarchy>
</Dimension>
```

- Hasil

Query PURCHASING using Mondrian OLAP



						Measures	
Time	Vendor	Employee	Product	Shipping Method	Warehouse	Total Cost	Quantity
All Times	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	148,263,062	3,985,315
	A. Datum Corporation	All Employee	All Product	All Shipping Method	All Warehouse		
	Advanced Bicycles	All Employee	All Product	All Shipping Method	All Warehouse		
	Allenson Cycles	All Employee	All Product	All Shipping Method	All Warehouse		
	American Bicycles and Wheels	All Employee	All Product	All Shipping Method	All Warehouse		
	American Bikes	All Employee	All Product	All Shipping Method	All Warehouse		
	Anderson's Custom Bikes	All Employee	All Product	All Shipping Method	All Warehouse		
	Aurora Bike Center	All Employee	All Product	All Shipping Method	All Warehouse		
	Australia Bike Retailer	All Employee	All Product	All Shipping Method	All Warehouse		
	Beaumont Bikes	All Employee	All Product	All Shipping Method	All Warehouse		
	Bergeron Off-Roads	All Employee	All Product	All Shipping Method	All Warehouse		
	Bicycle Specialists	All Employee	All Product	All Shipping Method	All Warehouse		
	Bike Satellite Inc.	All Employee	All Product	All Shipping Method	All Warehouse	4,469,106	70,950
	Bloomington Multisport	All Employee	All Product	All Shipping Method	All Warehouse		
	Burnett Road Warriors	All Employee	All Product	All Shipping Method	All Warehouse		
	Business Equipment Center	All Employee	All Product	All Shipping Method	All Warehouse		
	Capital Road Cycles	All Employee	All Product	All Shipping Method	All Warehouse		

### 3. fact\_purchasing (Menganalisis Dimensi Employee)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_purchasing) dengan dimensi Employee.

- Query pengisian data:

```
<Dimension name="Employee" primaryKey="EmployeeID">
    <Hierarchy hasAll="true" allMemberName="All Employee" primaryKey="EmployeeID">
        <Table name="dimemployee"/>

        <!-- Level 1: Nama karyawan -->
        <Level name="Employee Name" column="EmployeeName"/>

        <!-- Level 1: Nama karyawan -->
        <Level name="Job Title" column="JobTitle"/>

        <!-- Level 2: Status pernikahan -->
        <Level name="Marital Status" column="MaritalStatus"/>

        <!-- Level 3: Jenis kelamin -->
        <Level name="Gender" column="Gender"/>

        <!-- Level 4: Usia -->
        <Level name="Age" column="Age"/>
    </Hierarchy>
</Dimension>
```

- Hasil

**Query PURCHASING using Mondrian OLAP**



Time	Vendor	Employee	Product	Shipping Method	Warehouse	Measures
						Total Cost
						Quantity
+All Times	+All Vendor	+All Employee	+All Product	+All Shipping Method	+All Warehouse	148,263,062 3,985,315
		-Annette L Hill	+All Product	+All Shipping Method	+All Warehouse	12,550,363 338,820
		-Purchasing Assistant	+All Product	+All Shipping Method	+All Warehouse	12,550,363 338,820
		-M	+All Product	+All Shipping Method	+All Warehouse	12,550,363 338,820
		-F	+All Product	+All Shipping Method	+All Warehouse	12,550,363 338,820
		22	+All Product	+All Shipping Method	+All Warehouse	12,550,363 338,820
		+Arvind B Rao	+All Product	+All Shipping Method	+All Warehouse	6,339,709 176,160
		+Ben T Miller	+All Product	+All Shipping Method	+All Warehouse	11,031,444 295,520
		+Eric S Kurjan	+All Product	+All Shipping Method	+All Warehouse	14,678,978 394,340
		+Erin M Hagens	+All Product	+All Shipping Method	+All Warehouse	11,366,594 311,070
		+Frank S Pellow	+All Product	+All Shipping Method	+All Warehouse	14,301,903 372,745
		+Fukiko J Ogiu	+All Product	+All Shipping Method	+All Warehouse	11,854,237 313,670
		+Gordon L Hee	+All Product	+All Shipping Method	+All Warehouse	13,456,864 363,720
		+Linda P Meisner	+All Product	+All Shipping Method	+All Warehouse	15,841,957 414,870
		+Mikael Q Sandberg	+All Product	+All Shipping Method	+All Warehouse	15,408,324 424,170
		+Reinout N Hillmann	+All Product	+All Shipping Method	+All Warehouse	16,012,929 439,620
		+Sheela H Word	+All Product	+All Shipping Method	+All Warehouse	5,419,759 140,610

#### 4. fact\_purchasing (Menganalisis Dimensi Product)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_purchasing) dengan dimensi Product.

- Query pengisian data:

```
<Dimension name="Product" primaryKey="ProductID">
    <Hierarchy hasAll="true" allMemberName="All Product" primaryKey="ProductID">
        <Table name="dimproduct"/>

        <!-- Level 2: Kategori Produk -->
        <Level name="Product Category" column="ProductCategory"/>

        <!-- Level 3: Subkategori Produk -->
        <Level name="Product SubCategory" column="ProductSubCategory"/>

        <!-- Level 1: Detail Produk -->
        <Level name="Product Name" column="ProductName"/>

        <!-- Level 3: Subkategori Produk -->
        <Level name="Product Color" column="ProductColor"/>
    </Hierarchy>
</Dimension>
```

- Hasil

#### Query PURCHASING using Mondrian OLAP



						Measures	
Time	Vendor	Employee	Product	Shipping Method	Warehouse	Total Cost	Quantity
+All Times	+All Vendor	+All Employee	-All Product	+All Shipping Method	+All Warehouse	148,263,062	3,985,315
			-Accessories	+All Shipping Method	+All Warehouse	40,943,238	1,180,875
			+Bike Racks	+All Shipping Method	+All Warehouse		
			+Bike Stands	+All Shipping Method	+All Warehouse		
			+Bottles and Cages	+All Shipping Method	+All Warehouse	8,035	3,000
			+Cleaners	+All Shipping Method	+All Warehouse		
			+Fenders	+All Shipping Method	+All Warehouse		
			-Helmets	+All Shipping Method	+All Warehouse	4,578	350
			+Sport-100 Helmet, Black	+All Shipping Method	+All Warehouse	327	25
			+Sport-100 Helmet, Blue	+All Shipping Method	+All Warehouse	3,924	300
			-Sport-100 Helmet, Red	+All Shipping Method	+All Warehouse	327	25
			Red	+All Shipping Method	+All Warehouse	327	25
			+Hydration Packs	+All Shipping Method	+All Warehouse		
			+Lights	+All Shipping Method	+All Warehouse	4,458	475
			+Locks	+All Shipping Method	+All Warehouse	773	75
			+Panniers	+All Shipping Method	+All Warehouse	11,601	225
			+Pumps	+All Shipping Method	+All Warehouse	5,050	550

#### 5. fact\_purchasing (Menganalisis Dimensi Shipping Method)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_purchasing) dengan dimensi Shipping Method.

- Query pengisian data:

```
<Dimension name="Shipping Method" primaryKey="ShippingMethodID">
    <Hierarchy hasAll="true" allMemberName="All Shipping Method" primaryKey="ShippingMethodID">
        <Table name="dimshippingmethod"/>
        <Level name="Shipping Method ID" column="ShippingMethodID"/>
        <Level name="Shipping Method Name" column="ShippingMethodName"/>
    </Hierarchy>
</Dimension>
```

- Hasil

#### Query PURCHASING using Mondrian OLAP



						Measures	
Time	Vendor	Employee	Product	Shipping Method	Warehouse	Total Cost	Quantity
+All Times	+All Vendor	+All Employee	+All Product	-All Shipping Method	+All Warehouse	148,263,062	3,985,315
				-1	+All Warehouse	3,776,752	79,750
				CARGO TRANSPORT 5	+All Warehouse	3,776,752	79,750
				+2	+All Warehouse	27,311,384	954,660
				+3	+All Warehouse	1,111,146	70,005
				+4	+All Warehouse	26,028,001	761,200
				+5	+All Warehouse	90,035,779	2,119,700

#### 6. fact\_purchasing (Menganalisis Dimensi Warehouse)

Mendefinisikan struktur CUBE yang menghubungkan tabel fakta (fact\_purchasing) dengan dimensi Warehouse.

- Query pengisian data:

```
<Dimension name="Warehouse" primaryKey="WarehouseID">
    <Hierarchy hasAll="true" allMemberName="All Warehouse" primaryKey="WarehouseID">
        <Table name="dimwarehouse"/>
        <Level name="Warehouse ID" column="WarehouseID"/>
        <Level name="Location" column="Location"/>
    </Hierarchy>
</Dimension>
```

- Hasil

#### Query PURCHASING using Mondrian OLAP



						Measures	
Time	Vendor	Employee	Product	Shipping Method	Warehouse	Total Cost	Quantity
+All Times	+All Vendor	+All Employee	+All Product	+All Shipping Method	+All Warehouse	148,263,062	3,985,315
					-1	19,930,483	438,670
					Tool Crib	19,930,483	438,670
					+2		
					+3		
					+4		
					+5	1,907,582	25,070
					+6	54,582,673	1,482,250
					+7	1,111,146	70,005
					+10		
					+20		
					+30		
					+40		
					+45		
					+50	56,490,255	1,507,320
					+60	14,240,923	462,000

### 3.3. Implementasi Dashboard

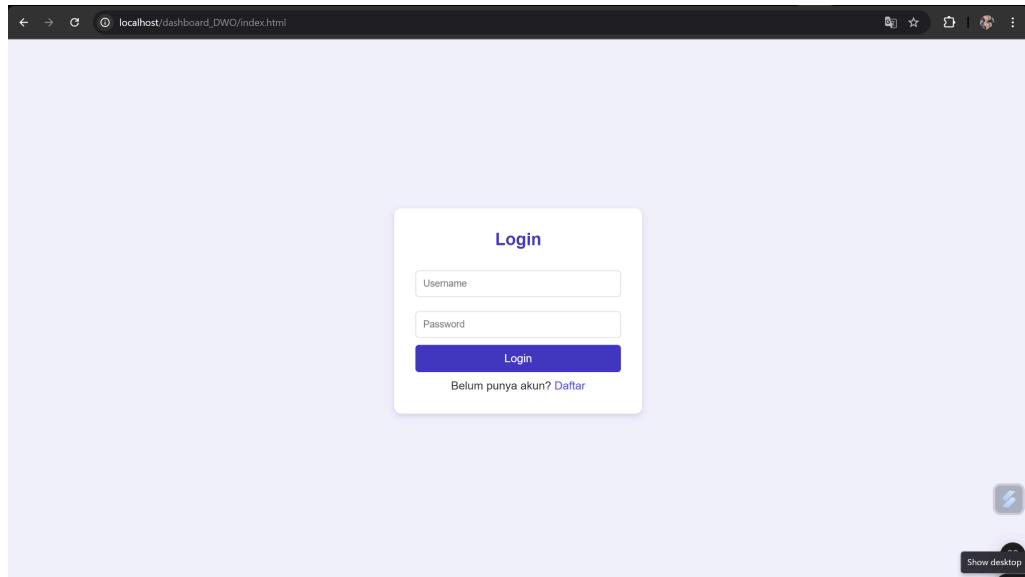
#### 3.3.1. Form Registrasi

The screenshot shows a web browser window with the URL 'localhost/dashboard\_DWO/index.html#'. In the center is a registration form with the title 'Register'. It contains two input fields: 'Username' and 'Password', both with placeholder text. Below these is a blue 'Register' button. At the bottom of the form, there is a link 'Sudah punya akun? Login'.

Tampilan form registrasi ini dirancang untuk memungkinkan pengguna membuat akun baru dengan cara yang sederhana. Pengguna diharuskan memasukkan nama pengguna (username) dan kata sandi (password) ke dalam form yang tersedia. Setelah data diisi dan tombol **Register** ditekan, sistem akan memproses registrasi dan menampilkan notifikasi berupa pesan pop-up, seperti "Registrasi berhasil! Silakan Login." Pesan ini mengonfirmasi bahwa proses registrasi telah sukses. Di bagian bawah form, terdapat tautan bertuliskan **Login**, yang memudahkan pengguna untuk langsung menuju halaman login jika mereka telah berhasil melakukan registrasi.

Desain form ini memberikan pengalaman yang intuitif dan efisien bagi pengguna dalam mendaftarkan akun mereka.

### 3.3.2. Login



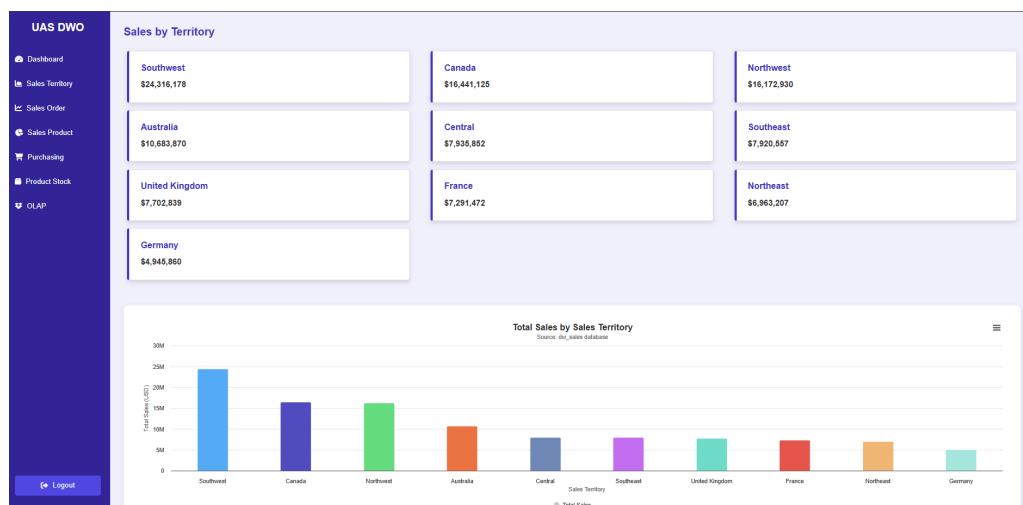
Form login dirancang untuk pengguna mengakses akun mereka dengan memasukkan nama pengguna (username) dan kata sandi (password) yang telah didaftarkan sebelumnya. Setelah pengguna mengisi form dan menekan tombol **Login**, sistem akan memverifikasi kredensial yang dimasukkan. Jika validasi berhasil, pengguna akan diarahkan ke halaman dashboard atau halaman utama aplikasi. Sebaliknya, jika nama pengguna atau kata sandi salah, sistem akan menampilkan notifikasi atau pesan error yang menginformasikan bahwa login gagal.

### 3.3.3. Dashboard

A screenshot of the UAS DWO dashboard. On the left side, there is a dark blue sidebar with the title 'UAS DWO' at the top. Below it is a vertical list of navigation items: 'Dashboard' (selected), 'Sales Territory', 'Sales Order', 'Sales Product', 'Purchasing', 'Product Stock', and 'OLAP'. At the bottom of the sidebar is a blue button labeled 'Logout'. The main content area has a light gray header with the word 'Dashboard'. Below the header are six data cards arranged in a grid. The first row contains three cards: 'Total Pembelian \$148,263,062.14', 'Total Penjualan \$110,373,888.74', and 'Jumlah Vendor 104'. The second row contains three cards: 'Jumlah Produk Terjual 295', 'Rata-Rata Pembelian per Tahun \$12,355,255.18', and 'Rata-Rata Penjualan per Tahun \$9,197,824.06'. At the very bottom of the dashboard, centered, is the text 'Copyright © UAS DWO 2024'.

Dashboard ini menampilkan ringkasan data utama terkait pembelian dan penjualan, dirancang untuk memberikan informasi secara cepat dan terstruktur. Elemen utama dashboard mencakup **Total Pembelian** yang menunjukkan keseluruhan biaya pembelian, **Total Penjualan** yang menampilkan pendapatan total, serta **Jumlah Vendor** yang mencatat total pemasok yang terlibat. Selain itu, terdapat metrik seperti **Jumlah Produk Terjual** untuk menunjukkan total produk yang telah dijual, **Rata-Rata Pembelian per Tahun**, dan **Rata-Rata Penjualan per Tahun** untuk memberikan wawasan mengenai tren pembelian dan penjualan tahunan. Di sisi kiri, terdapat panel navigasi dengan menu seperti **Dashboard**, **Sales Territory**, **Sales Order**, **Sales Product**, **Purchasing**, **Product Stock**, dan **OLAP**, yang memungkinkan pengguna mengakses berbagai fitur lainnya. Tombol **Logout** terletak di bagian bawah panel untuk keluar dari sistem.

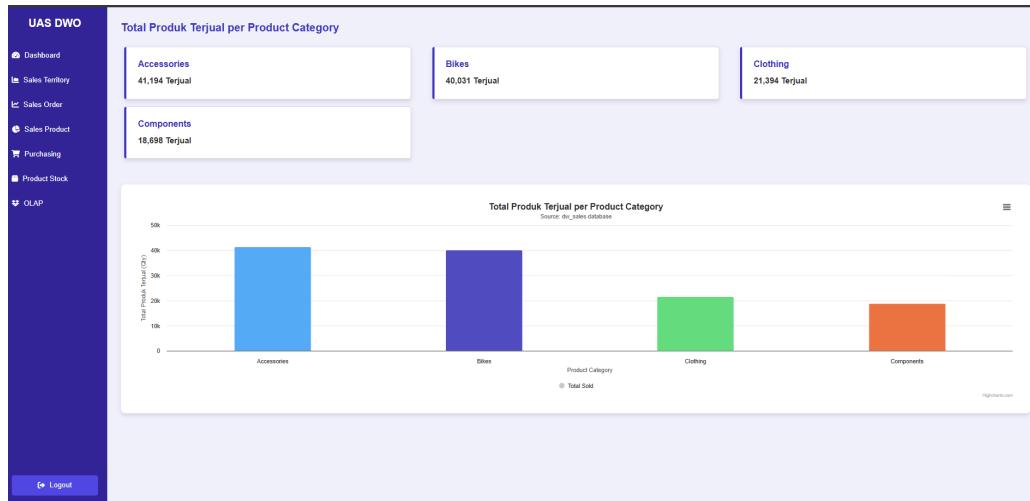
### 3.3.4. Dashboard Chart Sales by Territory



Dashboard **Sales by Territory** menampilkan total penjualan berdasarkan wilayah geografis. Di bagian atas, terdapat daftar wilayah penjualan seperti **Southwest**, **Canada**, **Northwest**, **Australia**, **Central**, **France**, dan lainnya, bersama dengan total penjualan masing-masing wilayah. Data ini memberikan informasi terperinci tentang kontribusi penjualan dari setiap wilayah. Di bagian bawah, terdapat grafik batang yang memvisualisasikan total penjualan per wilayah untuk memudahkan perbandingan antar wilayah. Grafik ini memungkinkan pengguna dengan cepat mengidentifikasi wilayah dengan penjualan tertinggi, seperti **Southwest**, yang memiliki total penjualan terbesar, dan wilayah dengan penjualan terendah,

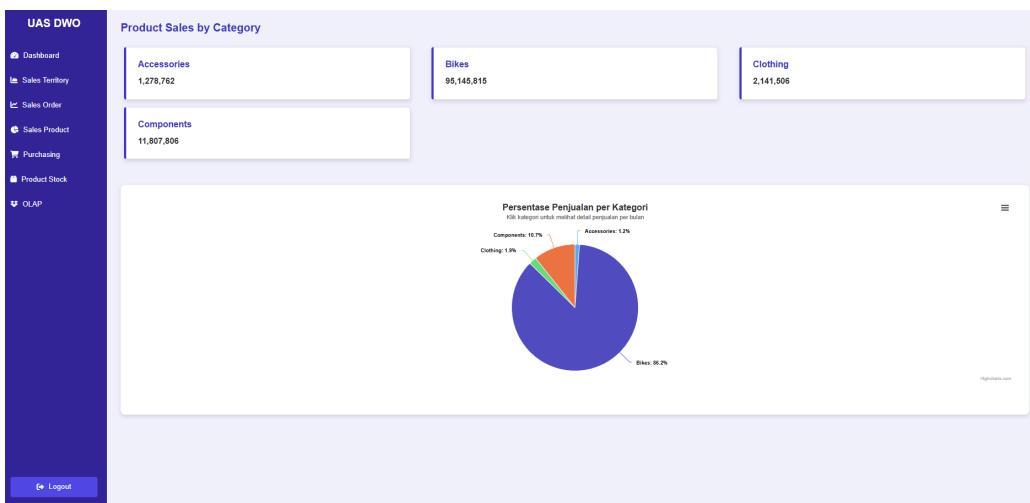
seperti **Germany**. Dashboard ini dirancang untuk memberikan wawasan tentang performa penjualan per wilayah.

### 3.3.5. Dashboard Chart Sales by Order



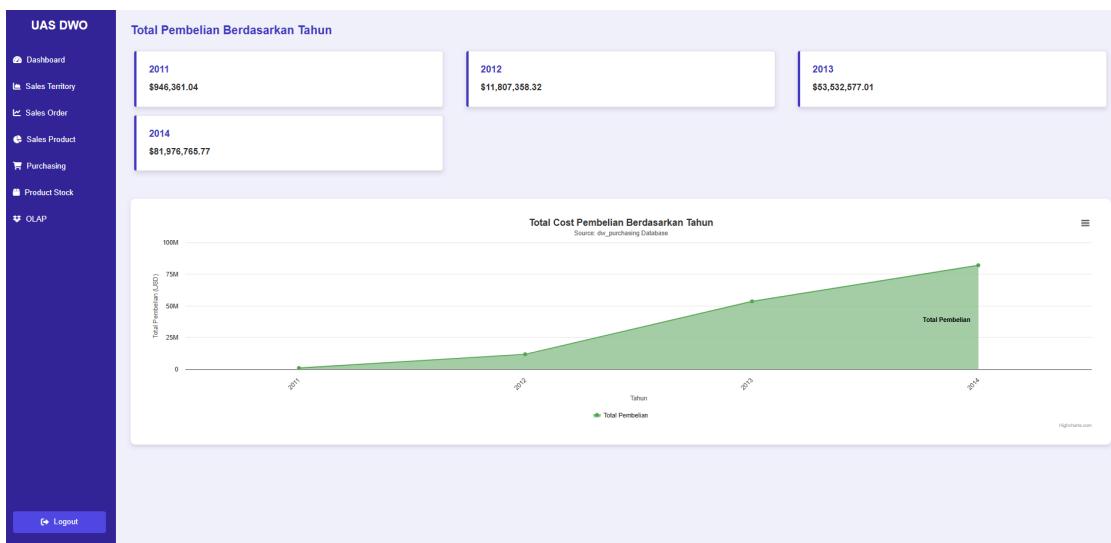
Dashboard **Total Produk Terjual per Product Category** menyajikan data penjualan berdasarkan kategori produk. Di bagian atas, terdapat daftar kategori produk, yaitu **Accessories**, **Bikes**, **Clothing**, dan **Components**, yang disertai dengan total produk terjual untuk setiap kategori. Informasi ini memberikan gambaran kontribusi masing-masing kategori terhadap total penjualan. Di bagian bawah, terdapat grafik batang yang memvisualisasikan total produk terjual untuk setiap kategori, memungkinkan pengguna untuk membandingkan performa kategori secara langsung. Dari grafik ini terlihat bahwa kategori **Accessories** dan **Bikes** memiliki jumlah produk terjual tertinggi dibandingkan kategori lainnya.

### 3.3.6. Dashboard Chart Sales by Category Product



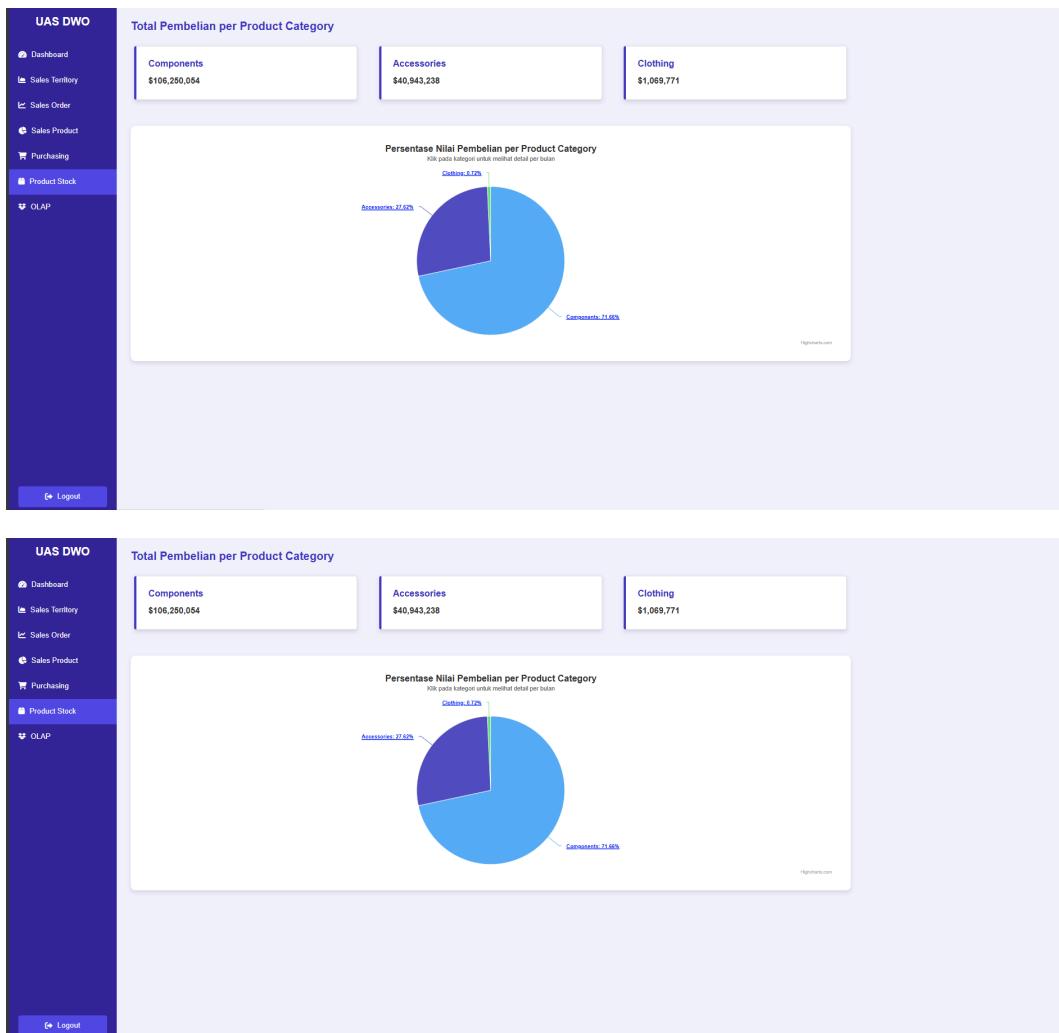
Dashboard ini menampilkan **Product Sales by Category**, yang memberikan informasi mengenai total penjualan untuk setiap kategori produk. Di bagian atas, ditampilkan jumlah total penjualan per kategori, seperti **Bikes** (kategori dengan kontribusi terbesar), **Components**, **Clothing**, dan **Accessories**. Data ini memberikan gambaran tentang seberapa besar kontribusi masing-masing kategori terhadap total penjualan. Di bagian bawah, terdapat diagram lingkaran (**pie chart**) yang memvisualisasikan persentase penjualan berdasarkan kategori produk. Diagram ini menunjukkan bahwa **Bikes** memiliki pangsa terbesar dengan 86.2%, diikuti oleh **Components** (10.7%), **Clothing** (1.9%), dan **Accessories** (1.2%).

### 3.3.7. Dashboard Chart Purchasing



Dashboard ini menampilkan **Total Pembelian Berdasarkan Tahun**, yang memberikan informasi tentang total biaya pembelian dari tahun 2011 hingga 2014. Di bagian atas, ditampilkan total pembelian untuk setiap tahun, dengan angka yang terus meningkat, dari \$946,361.04 pada tahun 2011 hingga mencapai \$81,976,765.77 pada tahun 2014. Informasi ini menunjukkan tren peningkatan pengeluaran pembelian setiap tahunnya. Di bagian bawah, terdapat grafik area yang memvisualisasikan total pembelian dari tahun ke tahun. Grafik ini memberikan gambaran yang jelas tentang kenaikan pembelian secara signifikan, terutama dari tahun 2012 ke tahun 2014.

### 3.3.8. Dashboard Chart Product Stock



Dashboard ini menampilkan **Total Pembelian per Product Category**, memberikan informasi tentang total pengeluaran pembelian untuk masing-masing kategori produk. Di bagian atas, ditampilkan total pembelian untuk kategori **Components** (\$106,250,054), **Accessories** (\$40,943,238), dan **Clothing** (\$1,069,771), memberikan gambaran kontribusi setiap kategori terhadap total pengeluaran. Di bagian bawah, terdapat diagram lingkaran (**pie chart**) yang memvisualisasikan persentase pembelian berdasarkan kategori produk. **Components** menyumbang 71.66% dari total pembelian, diikuti oleh **Accessories** (27.62%) dan **Clothing** (0.72%). Diagram ini interaktif, sehingga pengguna dapat mengklik kategori tertentu untuk melihat rincian pembelian berdasarkan bulan.

### 3.3.9. Dashboard OLAP

The screenshot shows the UAS DWO dashboard with a sidebar on the left containing links for Dashboard, Sales Territory, Sales Order, Sales Product, Purchasing, Product Stock, and OLAP. A blue 'Logout' button is at the bottom of the sidebar. The main content area is titled 'Tampilan OLAP DATABASE DWO - Using Mondrian'. It displays a license notice from Eclipse Public License v1.0, followed by 'OLAP DB ADVENTUREWORKS:' and two options: 'OLAP SALES' and 'OLAP PURCHASE'. The background of the main area is light gray.

Dashboard ini merupakan halaman utama untuk **OLAP Database DWO**, yang menggunakan Mondrian sebagai engine OLAP. Halaman ini memberikan akses ke dua opsi analisis utama, yaitu **OLAP SALES** dan **OLAP PURCHASE**, yang masing-masing menghubungkan pengguna ke analisis data penjualan dan pembelian. Tampilan ini memanfaatkan Mondrian untuk melakukan pengolahan data secara multidimensi, memungkinkan pengguna mengeksplorasi data secara interaktif melalui cube OLAP. Dengan Mondrian, pengguna dapat melakukan berbagai analisis, seperti drill-down, roll-up, slicing, dan dicing pada data penjualan dan pembelian, sehingga membantu dalam pengambilan keputusan yang lebih mendalam dan strategis. Dashboard ini menjadi pusat navigasi utama untuk akses ke analisis berbasis OLAP di database **AdventureWorks**.

The screenshot shows the UAS DWO dashboard with the same sidebar and 'Logout' button. The main content area is titled 'Tampilan OLAP DATABASE DWO - Using Mondrian' and contains a sub-section titled 'Query SALES using Mondrian OLAP'. It features a toolbar with various icons for data analysis. Below the toolbar is a table titled 'Measures' with columns for Time, Customer, Employee, Product, Territory, TotalAmount, and Quantity. The table data is as follows:

Time	Customer	Employee	Product	Territory	TotalAmount	Quantity
All Times	All Customers	All Employee	All Product	All Territory	220,747,779	549,828
+2011	All Customers	All Employee	All Product	All Territory	25,064,844	25,694
+2012	All Customers	All Employee	All Product	All Territory	62,686,317	126,822
+2013	All Customers	All Employee	All Product	All Territory	86,734,490	259,028
+2014	All Customers	All Employee	All Product	All Territory	46,262,128	138,284

Below the table, there is a section labeled 'Slicer' with a link to 'back to index'.

Tampilan ini menunjukkan hasil **OLAP SALES** menggunakan Mondrian, yang menampilkan data penjualan berdasarkan dimensi seperti waktu, pelanggan, karyawan, produk, dan wilayah. Data disajikan dalam bentuk tabel interaktif, di mana pengguna dapat melihat metrik utama seperti **SalesAmount** (jumlah penjualan) dan **Quantity** (kuantitas produk yang terjual) untuk setiap tahun. Tabel ini memungkinkan eksplorasi mendalam melalui fitur drill-down pada dimensi seperti waktu (misalnya, per tahun atau total semua waktu). Pengguna dapat menganalisis performa penjualan dari tahun ke tahun, di mana tahun 2013 menunjukkan kontribusi tertinggi dengan **SalesAmount** sebesar \$86,734,90.

Time	Vendor	Employee	Product	Shipping Method	Warehouse	Measures
All Times	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	Total Cost
2011	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	148,263,062
2012	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	946,361
2013	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	11,807,358
2014	All Vendor	All Employee	All Product	All Shipping Method	All Warehouse	53,532,577

Tampilan ini menunjukkan hasil **OLAP PURCHASING** menggunakan Mondrian, yang menyajikan data pembelian berdasarkan dimensi seperti waktu, vendor, karyawan, produk, metode pengiriman, dan gudang. Data disajikan dalam bentuk tabel interaktif, yang memungkinkan pengguna melihat metrik utama seperti **Total Cost** (total biaya pembelian) dan **Quantity** (jumlah barang yang dibeli) untuk setiap kategori dan periode waktu. Tabel ini dilengkapi dengan fitur drill-down, memungkinkan pengguna menggali lebih dalam detail pembelian berdasarkan dimensi tertentu. Misalnya, total biaya pembelian sepanjang waktu adalah \$148,263,062 dengan jumlah barang sebanyak 3,985,315 unit, dan rincian tahunan dapat dilihat untuk tahun 2011 hingga 2014. Tahun 2014 mencatat total pembelian tertinggi sebesar \$81,976,766 dengan jumlah barang 2,193,695 unit.

## **LAMPIRAN**

**Link Github**

<https://github.com/adelialiae/UAS-DWO-PAR-B-2024.git>