PEMBANGUNAN PERANGKAT LUNAK DATA MART PT. FOXIMAS MANDIRI PADA DIVISI PRODUKSI

Irwan Ruchwana SB[1]

[1] Program Studi Teknik Informatika
Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonesia
Jl. Dipatiukur No 112-116 Bandung - Indonesia
E-mail : ruchwana@gmail.com

ABSTRAK

PT. Foximas Mandiri merupakan perusahaan swasta yang bergerak dibidang produksi sepatu kulit. Dalam pembuatan laporan akhir produksi masih mengalami kesulitan karena semua dokumen produksi masih belum terkomputerisasi, dengan banyaknya data yang tidak terintegrasi membuat pihak manajer produksi kesulitan dalam menganalisa perkembangan produksi dan kesulitan dalam pencarian data yang akan dijadikan suatu informasi strategis.

Berdasarkan permasalahan yang ada pada PT. Foximas Mandiri, maka perlu dibangun sebuah perangkat lunak *data mart*, dimana data laporan akhir produksi dapat dibuat dengan sebaik mungkin dan data-data yang ada dapat terintegrasi dengan baik untuk memudahkan perusahaan dalam mengontrol hasil produksi dan dapat di manfaatkan semaksimal mungkin untuk memberikan kemudahan dalam pencarian data sesuai dengan kebutuhan manajer produksi agar data yang diperlukan dapat menyajikan informasi secara cepat dan detail, dalam bentuk tabel multidimensi dan grafik yang nantinya dapat dianalisis sehingga membantu dalam pengambilan keputusan. Pembangunan perangkat lunak *data mart* menggunakan UML (*Unified Modeling Language*), *data mart* adalah bagian dari *data warehouse* yang membantu dalam pembuatan laporan dan analisis data pada suatu unit.

Berdasarkan hasil pengujian *black box* dan pengujian wawancara dapat di simpulkan bahwa pembangunan perangkat lunak *data mart* dapat memudahkan dalam mendapatkan suatu informasi strategis secara cepat dan detail yang dibutuhkan oleh manajer produksi PT. Foximas Mandiri untuk mempercepat proses penganalisaan produksi ke depannya yang akan membantu pengambilan keputusan oleh manajer produksi dan membantu memudahkan manajer produksi dalam pembuatan laporan akhir produksi yang multidimensi sesuai dengan kebutuhan PT. Foximas Mandiri.

Kata kunci : PT. Foximas Mandiri, Informasi Strategis, *Data Mart.*

1. PENDAHULUAN

PT. Foximas Mandiri merupakan perusahaan swasta yang bergerak dibidang produksi sepatu kulit. PT. Foximas Mandiri dipercaya sebagai produsen pembuat sepatu kulit yang di kirim ke seluruh outlet sepatu kulit di seluruh indonesia dan memenuhi permintaan pembuatan sepatu militer bagi instansi pemerintah terutama TNI dan Polri.

Saat ini pembuatan laporan akhir produksi masih mengalami kesulitan karena semua dokumen produksi masih belum terkomputerisasi, dengan banyaknya data yang tidak terintegrasi membuat pihak manajer produksi kesulitan dalam menganalisa perkembangan produksi dan kesulitan dalam pencarian data yang akan dijadikan suatu informasi strategis. Dalam menyajikan informasi yang memiliki periode waktu akan membutuhkan waktu yang cukup lama dalam menyajikannya dikarenakan oleh pencarian data dan pemrosesannya yang lambat serta belum adanya suatu sistem yang dapat menyajikan informasi yang cepat dan detail. Untuk itu perlu dibangun sebuah perangkat lunak *data mart*, dimana data laporan akhir produksi dapat dibuat dengan sebaik mungkin dan data-data yang ada dapat terintegrasi dengan baik untuk memudahkan perusahaan dalam mengontrol hasil produksi dan dapat di manfaatkan semaksimal mungkin untuk memberikan kemudahan dalam pencarian data sesuai dengan kebutuhan manajer produksi agar data yang diperlukan dapat menyajikan informasi secara cepat dan detail, dalam bentuk tabel multidimensi dan grafik yang nantinya dapat dianalisis sehingga membantu dalam pengambilan keputusan. Tujuan dari membangun perangkat lunak *data mart* pada PT. Foximas Mandiri ini adalah untuk :

1. Membantu manajer produksi dalam menampilkan kebutuhan informasi strategis secara cepat dan ringkas, serta dapat menyajikan informasi yang multidimesi dengan tabel dan grafik yang memudahkan dalam menganalisa produksi perusahaan untuk membantu pengambilan keputusan yang dibuat oleh manajer produksi.
2. Membantu pihak manajer produksi dalam pembuatan laporan akhir produksi yang multidimensi.

Batasan masalah yang diterapkan dalam pembangunan perangkat lunak *data mart* ini antara lain, yaitu :

1. Data yang diambil adalah data yang dimiliki oleh Divisi Produksi yaitu data produksi dan data SPK.
2. Pemodelan untuk pembangunan perangkat lunak menggunakan UML (*Unified Modeling Language*).
3. Pengujian data *mart* menggunakan OLAP (*On-Line Analytical Processing*).
4. Data untuk pengujian *data mart* yaitu dari tahun 2013 sampai dengan tahun 2014
5. Proses pembuatan *data mart* dengan proses ETL (*Extract Transform Loading*).
6. Pembangunan perangkat lunak menggunakan *visual studio* 2012
7. DBMS menggunakan *microsoft SQL server* 2012

1.1 Data Mart

Data mart merupakan suatu bagian dari *data warehouse* yang mendukung pembuatan laporan dan analisis data pada suatu unit. *Data mart* berisi informasi yang relefan bagi pengguna yang ingin mengambil keputusan.

Ada empat tugas yang dapat dilakukan dengan adanya *data mart* [2], empat tugas tersebut adalah sebagai berikut :

1. Pembuatan laporan
   Pembuatan laporan merupakan salah satu kegunaan *data mart* yang paling umum dilakukan. Dengan menggunakan *query* sederhana didapatkan laporan per hari, per bulan, per tahun, atau jangka waktu kapanpun yang diinginkan.
2. *On-Line Analytical Processing* (OLAP)
   Dengan adanya *data mart*, semua informasi baik detail maupun hasil *summary* yang dibutuhkan dalam proses analisa mudah didapat. OLAP merupakan konsep data multidimensi dan memungkinkan pada pemakai menganalisa data sampai mendetail, tanpa mengetikkan satupun perintah SQL.
3. *Data Mining*
   Data mining merupakan proses untuk menggali (*mining*) pengetahuan dan informasi baru dari data yang berjumlah banyak pada *data mart.*
4. Proses informasi *executive.*
   Data mart dapat membuat ringkasan informais yang penting dengan tujuan membuat keputusan bisnis, tanpa harus menjelajahi keseluruhan data. Dengan menggunakan *data mart* segala laporan telah diringkas dan dapat pula mengetehaui segala rinciannya secara lengkap, sehingga mempermudah proses pengambilan keputusan.

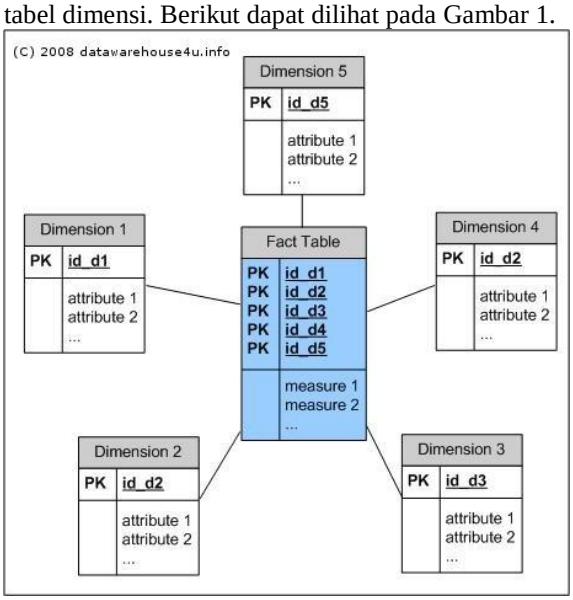
1.2 Model Dimensional Data Mart

Model dimensional pada *data mart* terdiri dari tabel fakta dan tabel dimensi,

Ada beberapa model skema yang terdapat pada pemodelan *data mart*, yaitu skema *star*, skema *snowflake*, dan skema *constellation*. Penjelasan dari masing-masing model skema adalah sebagai berikut :

1. *Star Schema*

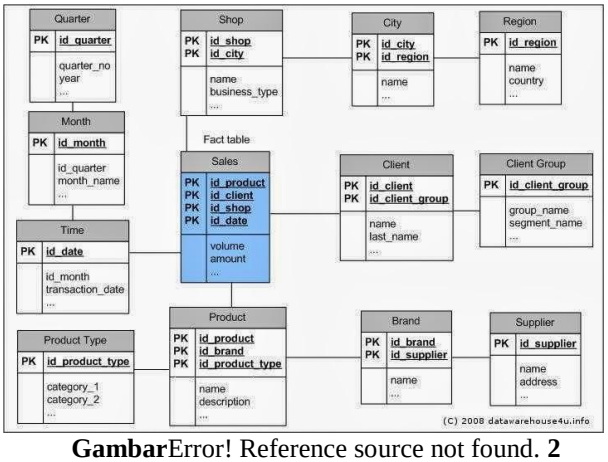
Skema ini mengikuti bentuk bintang, dimana terdapat satu tabel fakta di pusat bintang dengan beberapa tabel dimensi yang mengelilinginya. Semua tabel dimensi berhubungan dengan ke tabel fakta. Tabel fakta memiliki beberapa *primary key* dalam tabel dimensi. Berikut dapat dilihat pada Gambar 1.



Gambar 1 *Star Schema*

2. *Snowflake Schema*

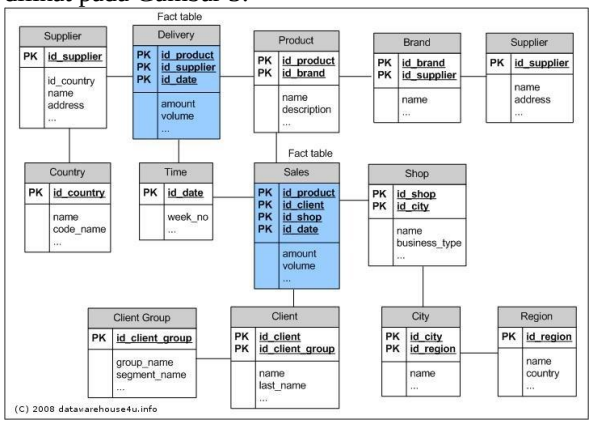
Snowflake Schema merupakan perluasan dari *star schema* dengan tambahan beberapa tabel dimensi yang tidak berhubungan secara langsung dengan tabel fakta. Tabel dimensi tersebut berhubungan dengan tabel dimensi yang lain. Berikut dapat dilihat pada Gambar**Error! Reference source not found.** 2



GambarError! Reference source not found. **2 *Snowflake Schema***

3. Skema *Constellation*

Pada skema ini terdapat beberapa tabel fakta yang menggunakan satu atau beberapa tabel dimensi. Berikut adalah contoh skema constellation dapat dilihat pada Gambar 3.



Gambar 3 Skema *Constellation*

1.3 ETL (*Extract, Transform, Loading*)

Proses ETL (*Extract, Transform, Loading*) merupakan proses yang harus dilalui dalam pembentukan *data mart* [3]. Tujuan ETL adalah mengumpulkan, menyaring, mengolah dan menggabungkan data dari berbagai sumber untuk disimpan ke dalam *data mart*. Berikut adalah penjelasan dari tiap proses ETL :

1. Ekstraksi Data (*Extract*)

Ekstraksi data adalah proses pengambilan data dari berbagai sistem operasional, baik menggunakan *query* atau aplikasi ETL. Terdapat beberapa fungsi ekstraksi data, yaitu :

a. Ekstraksi data secara otomatis dari aplikasi sumber.
b. Penyaringan atau seleksi data hasil ekstraksi.
c. Pengiriman data dari berbagai *platform* aplikasi ke sumber data.
d. Perubahan format *layout* data dari format aslinya.
e. Penyimpanan dalam *file* sementara untuk penggabungan dengan hasil ekstraksi dari sumber lain.

2. Transformasi Data (*Transformation*)

Transformasi adalah proses dimana data hasil ekstraksi disaring dan diubah sesuai dengan aturan bisnis yang berlaku. Langkah-langkah dalam transformasi data adalah sebagai berikut :

a. Memetakan data input dari skema data aslinya ke skema *data mart*.
b. Melakukan konversi tipe data.
c. Membersihkan dan membuang data yang sama (duplikasi).
d. Memeriksa referensi data.
e. Mengisi nilai-nilai yang kosong dengan nilai *default*.
f. Menggabungkan data.

3. Pengisian Data (*Loading*)

Pengisian data adalah proses memasukkan data yang didapatkan dari hasil transformasi ke dalam *data mart*. Cara untuk memasukkan data adalah dengan menjalankan *SQL script* secara periodik.

1.4 OLAP (*On-Line Analytical Processing*)

OLAP (*On-Line Analytical Processing*) adalah teknologi yang memproses data ke dalam struktur multidimensi, menyediakan jawaban yang cepat untuk *query* analisis yang kompleks dengan tujuan untuk mengorganisir sejumlah data yang besar, agar bisa dianalisa dan dievaluasi dengan cepat serta menyediakan kecepatan dan fleksibelitas untuk mendukung analisa dalam waktu nyata [1].

Ada beberapa karakteristik OLAP, yaitu :

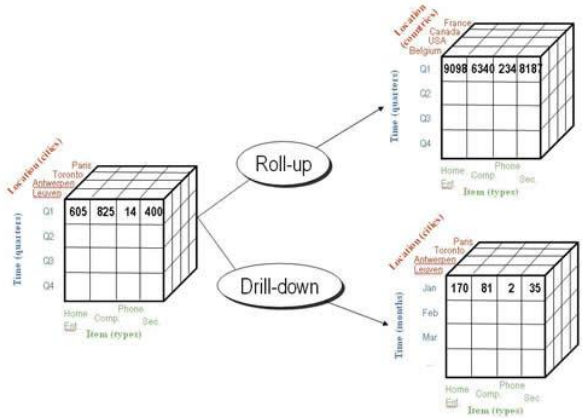
1. Mengijinkan pelaku bisnis melihat data dari sudut pandang *logical* dan multidimensional pada *data warehouse*.
2. Memfasilitasi *query* yang kompleks dan analisa bagi pengguna.
3. Mengijinkan pengguna melakukan *drill-down* untuk menampilkan data yang lebih detail atau *roll-up* untuk agregasi dari suatu dimensi atau beberapa dimensi.
4. Menyediakan proses kalkulasi dan perbandingan data.
5. Menampilkan hasil dalam bentuk tabel atau grafik.

Keuntungan dari OLAP, yaitu :

1. Mengingkatkan produktivitas pemakai akhir bisnis, pengembang IT, dan keseluruhan organisasi.
2. Pengawasan yang lebih dan akses tepat waktu terhadap informasi strategis dapat membuat pengambilan keputusan lebih cepat.
3. Mengurangi pengembangan aplikasi bagi staf IT dengan membuat pemakaian akhir dapat mengubah skema dan membuat model sendiri.
4. Penyimpanan pengawasan organisasi melalui integritas data koorporasi sebagai aplikasi OLAP tergantung pada *data warehouse* dan sistem OLTP untuk memperbaharui sumber tingkatan data.

OLAP dapat digunakan untuk melakukan seperti [1] :

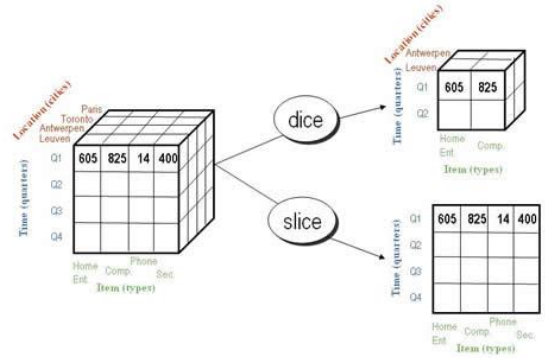
1. Konsolidasi (*roll-up*)
   Konsolidasi melibatkan pengelompokan data.
2. *Drill-down*
   Suatu bentuk yang merupakan kebalikan dari konsolidasi untuk menjabarkan data yang ringkas menjadi data yang lebih detail. Gambaran untuk *roll-up* dan *drill-down* dapat dilihat pada Gambar 4.



Gambar 4 *Roll-up* dan *Drill-down*

3. *Slicing* dan *dicing*

Menjabarkan pada kemampuan untuk melihat data dari sudut pandang. Gambaran untuk *slicing* dan *dicing* dapat dilihat pada Gambar 5.



Gambar 5 *Slicing* dan *Dicing*

1.5 SSIS (*SQL Server Integration Service*)

SSIS (*SQL Server Integration Services*) adalah suatu platform untuk membangun sistem yang handal untuk integrasi data, *extraction*, *transformation*, dan *loading* yang digunakan pada *data warehousing* [4].

Arsitektur *SQL Server Integration Services* secara umum berisi berbagai macam komponen, diantaranya yaitu :

1. *SSIS Deginer*. Merupakan *tools* yang digunakan untuk membuat dan mengatur paket *integration service*. Pada *SQL Server 2012*, *tools* ini sudah terintegrasi dengan *Visual Studio 2010* yang merupakan bagian *project Bussiness Intelegence*.
2 *Runtime Engine*. Komponen ini berguna untuk menjalankan semua paket SSIS yang sudah dibuat.
3 *Task* dan *executable binary*.
4 *Data Flow Engine* dan *Data Flow*. Komponen *data flow* merupakan enkapsulasi *data flow engine* yang menyediakan *buffer* didalam memori dan bertugas memindahkan data dari sumber data ke tujuan data. Sedangkan *data flow* merupakan sumber data, tujuan data, dan transformasi.
5 *Integration Services service*. Memungkinkan *SQL Server Management Studio* dapat digunakan untuk melakukan monitoring paket SSIS dan mengatur SSIS *storage* yang digunakan.
6 *SQL Server Import* dan *Export Wizard*. *Tools* ini digunakan untuk melakukan *copy* data dari sumber ke tujuan data.

2 ISI PENELITIAN

Terdapat beberapa tahapan analisis dalam pembangunan perangkat lunak *data mart* ini, yaitu:

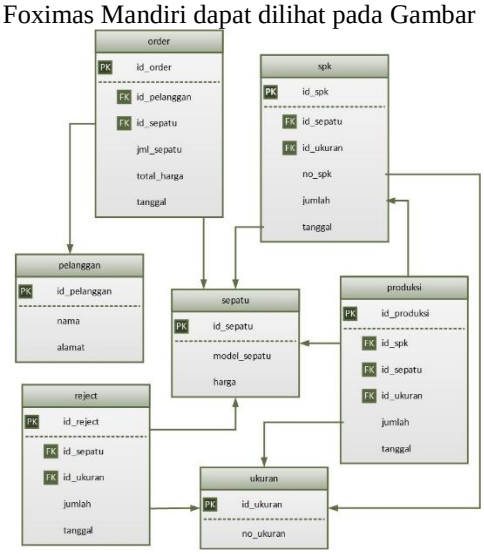
2.1 Analisis Kebutuhan Informasi

Analisis kebutuhan informasi merupakan tahap untuk menganalisis apa saja yang dibutuhkan oleh PT. Foximas Mandiri untuk *data mart* yang akan dibangun. Informasi yang akan disajikan secara detail. Berdasarkan hasil wawancara dengan manajer produksi, informasi yang dibutuhkan untuk pembangunan *data mart* adalah sebagai berikut :

1. Informasi jumlah sepatu yang diproduksi berdasarkan model sepatu pertanggal, perbulan dan pertahun.
2. Informasi jumlah sepatu yang diproduksi berdasarkan ukuran sepatu pertanggal, perbulan dan pertahun.
3. Informasi jumlah sepatu yang diproduksi berdasarkan model sepatu dan ukuran sepatu pertanggal, perbulan dan pertahun.
4. Informasi jumlah sepatu yang diproduksi berdasarkan no spk (surat perintah kerja) pertanggal, perbulan dan pertahun.
5. Informasi jumlah sepatu yang diproduksi berdasarkan no spk (surat perintah kerja) dan model sepatu pertanggal, perbulan dan pertahun.
6. Informasi jumlah sepatu yang diproduksi berdasarkan no spk (surat perintah kerja), model sepatu dan ukuran pertanggal, perbulan dan pertahun.
7. Informasi jumlah sepatu yang diorder berdasarkan nama pelanggan pertanggal, perbulan dan pertahun.
8. Informasi jumlah sepatu yang diorder berdasarkan model sepatu pertanggal, perbulan dan pertahun.
9. Informasi jumlah sepatu yang diorder berdasarkan nama pelanggan dan model sepatu pertanggal, perbulan dan pertahun.
10. Informasi jumlah sepatu reject berdasarkan model sepatu pertanggal, perbulan dan pertahun.
11. Informasi jumlah sepatu reject berdasarkan ukuran pertanggal, perbulan dan pertahun.
12. Informasi jumlah sepatu reject berdasarkan model sepatu dan ukuran pertanggal, perbulan dan pertahun.

2.2 Sumber Data

Pada penelitian ini sumber data yang digunakan yaitu dengan memanfaatkan OLTP yang terdapat di PT. Foximas Mandiri. Berikut diagram relasi OLTP PT. Foximas Mandiri dapat dilihat pada Gambar 6 :



Gambar 6 Diagram Relasi OLTP PT. Foximas Mandiri

2.3 Analisis Arsitektur Data Mart

Pada lapisan ini, data operasional akan diekstrak dengan proses ETL (*Extract Transform Loading*) ke dalam *data mart*.

1. *Extract*

Proses ini merupakan pemilihan data dari sumber data yang ada untuk pembuatan *data mart*, beberapa tabel yang akan di jadikan *data mart* yaitu tabel produksi, tabel ukuran, tabel sepatu, tabel spk, tabel order, tabel pelangggan, dan tabel reject. Atribut-atribut yang ada pada tabel yang akan diekstrak tidak ada perubahan masih tetap sama dengan sumber data yang ada pada OLTP. Proses ekstraksi data dari sumber data ke dalam *data mart* adalah sebagai berikut :

a. Proses *extract* pada tabel produksi

Proses *extract* pada tabel produksi, dilakukan proses pengambilan data dari database OLTP. Kolom yang di-*extract* adalah kolom id_produksi, id_spk, id_sepatu, id_ukuran, jumlah, dan tanggal. Hasil dari *extract* data pada tabel produksi dapat dilihat pada Tabel 1:

Tabel 1 *Extract* Produksi

Nama Tabel	Field
Tabel Produksi	id_produksi
	id_spk
	id_sepatu
	id_ukuran
	Jumlah
	Tanggal

b. Proses *extract* pada tabel ukuran

Proses *extract* pada tabel ukuran, dilakukan proses pengambilan data dari database OLTP. Kolom yang di-*extract* adalah kolom id_ukuran, dan no_ukuran. Hasil dari *extract* data pada tabel ukuran dapat dilihat pada Tabel 2 :

Tabel 2 *Extract* Ukuran

Nama Tabel	Field
Tabel Ukuran	id_ukuran
	no_ukuran

c. Proses *extract* pada tabel sepatu

Proses *extract* pada tabel sepatu, dilakukan proses pengambilan data dari database OLTP. Kolom yang di-*extract* adalah kolom id_sepatu, model_sepatu dan harga. Hasil dari *extract* data pada tabel sepatu dapat dilihat pada Tabel 3:

Tabel 3 *Extract* Sepatu

Nama Tabel	Field
Tabel Sepatu	id_sepatu
	model_sepatu
	Harga

d. Proses *extract* pada tabel order

Proses *extract* pada tabel order, dilakukan proses pengambilan data dari database OLTP. Kolom yang di-*extract* adalah kolom id_order, id_pelanggan, jumlah, dan banyak. Hasil dari *extract* data pada tabel order dapat dilihat pada Tabel 4:

Tabel 4 *Extract* Order

Nama Tabel	Field
Tabel Order	id_order
	id_pelanggan
	Id_sepatu
	Jml_sepatu
	Total_harga
	Tanggal

e. Proses *extract* pada tabel pelanggan

Proses *extract* pada tabel pelanggan, dilakukan proses pengambilan data dari database OLTP. Kolom yang di-*extract* adalah kolom id_pelanggan, nama, dan alamat. Hasil dari *extract* data pada tabel pelanggan dapat dilihat pada Tabel 5 :

Tabel 5 *Extract* Pelanggan

Nama Tabel	Field
Tabel pelanggan	id_pelanggan
	Nama
	Alamat

f. Proses *extract* pada tabel spk

Proses *extract* pada tabel spk, dilakukan proses pengambilan data dari database OLTP. Kolom yang di-*extract* adalah kolom id_spk, no_spk, dan tanggal.

Hasil dari *extract* data pada tabel spk dapat dilihat pada Tabel 6 :

Tabel 6 *Extract* Spk

Nama Tabel	Field
Tabel spk	id_spk
	Id_sepatu
	Id_ukuran
	no_spk
	Jumlah
	Tanggal

g. Proses *extract* pada tabel reject

Proses *extract* pada tabel reject, dilakukan proses pengambilan data dari database OLTP. Kolom yang di-*extract* adalah kolom id_reject, id_sepatu, id_ukuran, jumlah dan tanggal. Hasil dari *extract* data pada tabel spk dapat dilihat pada Tabel 7 :

Tabel 7 *Extract* Reject

Nama Tabel	Field
Tabel reject	Id_reject
	id_sepatu
	Id_ukuran
	Jumlah
	Tanggal

2. *Transform*

Proses *transform* yang dilakukan terdiri dari 2 proses, yaitu :

1. *Cleaning*

Proses *cleaning* dilakukan untuk membersihkan data yang tidak digunakan dari tabel yang sudah diekstrak. Berikut merupakan nama *field* yang dihilangkan dalam proses *cleaning* :

a) Pada tabel sepatu tidak memerlukan *field* harga.
b) Pada tabel pelanggan tidak memerlukan *field* alamat.
c) Pada tabel order tidak memerlukan *field* total_harga dan tanggal.
d) Pada tabel spk tidak memerlukan *field* tanggal.

2. *Conditioning*

Proses *conditioning* yaitu data yang sudah melalui proses pembersihan akan diubah format dari data oprasional menjadi format *data mart*. Untuk proses ini format yang diubah hanya nama tabel dan perubahan tanggal menjadi id_waktu. Berikut tabel-tabel yang di *conditioning* yaitu antara lain :

1) *Conditioning* tabel produksi

Pada proses *conditioning* tabel produksi akan diubah namanya menjadi tabel fact_produksi karena tabel produksi memiliki *measure* yaitu parameter yang digunakan dalam tabel fakta yang berupa data *numeric* atau data yang dapat di ukur. sedangkan untuk *field* tanggal menjadi id_waktu. Pada *field* tanggal akan menghasilkan tabel baru yaitu tabel dimensi waktu yaitu untuk menjelaskan informasi yang dibutuhkan berdasarkan waktu dalam penganalisaan data. Untuk lebih jelasnya dalam proses *conditioning* tabel produksi dapat dilihat pada Tabel 8 :

Tabel 8 *Conditioning* Tabel Produksi

Tabel Produksi	
No	**Field**
1	Id_produksi
2	id_sepatu
3	id_ukuran
4	Jumlah
5	Tanggal

→

Fact_produksi	
No	**Field**
1	Id_produksi
2	id_sepatu
3	id_ukuran
4	jumlah
5	id_waktu

Tabel Produksi	
Tanggal	Date

→

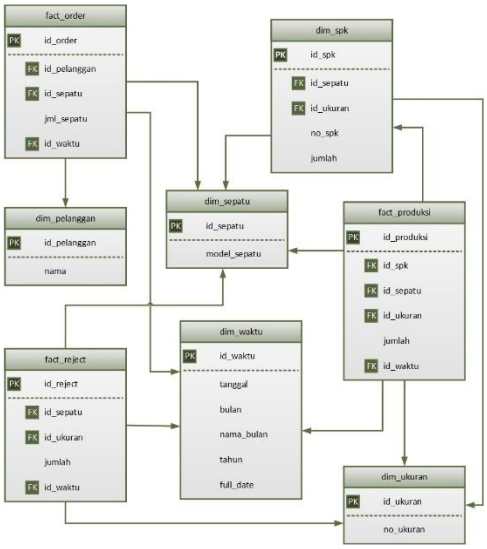
Dim_waktu	
id_waktu	integer
tanggal	integer
bulan	integer
nama_bulan	nvarchar(50)
tahun	integer
full_date	date

3. *Loading*

Setelah data di *extract* dan di *transform*, selanjutnya data tersebut dimasukkan ke dalam *data mart*. Proses *loading* pada aplikasi *data mart* akan dilakukan secara otomatis setelah proses *transform* selesai. Teknik yang digunakan adalah *update*, dimana proses ini akan langsung meng-*update data mart* tanpa merubah data yang sudah ada.

1. Analisis *Data Mart Layer*

Pada lapisan ini, data yang sudah melalui proses ETL akan disimpan pada sebuah penyimpanan *logic* yang tersentralisasi, yaitu *data mart*. Tabel-tabel yang akan dibutuhkan dalam pembuatan perancangan *data mart*, yaitu tabel fakta produksi, fakta order dan tabel fakta reject. Selain itu terdapat tabel dimensi yang akan digunakan bersama dalam beberapa tabel fakta, yaitu tabel dimensi sepatu, tabel dimensi ukuran, tabel dimensi pelanggan, tabel dimensi spk dan tabel dimensi waktu. Skema yang digunakan dalam pembangunan *data mart* adalah skema *constellation*. Berikut skema *constellation* dalam pembangunan *data mart* dapat dilihat pada Gambar 7 :



Gambar 7 Skema *Constellation*

Tabel 9 Penjelasan Skema *Data Mart*

No	Nama Tabel	Jenis Tabel
1	Fact_Produksi	Fakta
2	Fact_order	Fakta
3	Fact_reject	Fakta
4	Dim_Sepatu	Dimensi
5	Dim_Ukuran	Dimensi
6	Dim_Pelanggan	Dimensi
7	Dim_Waktu	Dimensi
8	Dim_Spk	Dimensi

2.4 Spesifikasi Kebutuhan Perangkat Lunak

Analisis spesifikasi kebutuhan perangkat lunak berisi deskripsi dari kebutuhan perangkat lunak yang akan dibangun baik kebutuhan fungsional maupun kebutuhan non fungsional. Tabel spesifikasi kebutuhan perangkat lunak dapat dilihat pada Tabel 10 dan Tabel 11 :

Tabel 10 Spesifikasi Kebutuhan Perangkat Lunak Fungsional

Kode Kebutuhan	Kebutuhan
SKPL-F-1	Administrator dan manajer produksi dapat mengakses aplikasi.
SKPL-F-2	Sistem dapat melakukan *update* data pada *data mart*.
SKPL-F-3	Sistem dapat melakukan proses ETL (*Extract Transform Loading*).
SKPL-F-4	Manajer produksi dapat melakukan analisis terhadap *data mart*.
SKPL-F-5	Sistem dapat menampilkan hasil analisis berupa tabel dan grafik.
SKPL-F-6	Sistem dapat memberikan laporan tercetak.

Tabel 11 Spesifikasi Kebutuhan Perangkat Lunak Non Fungsional

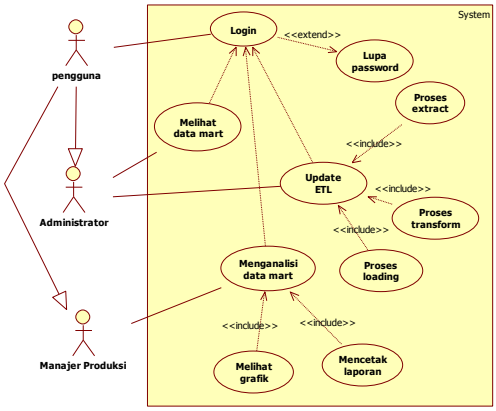
Kode Kebutuhan	Kebutuhan
SKPL-NF-1	Sistem dapat berjalan pada *platfrom windows*.
SKPL-NF-2	Sistem dapat berjalan pada *hardware* berspesifikasi menengah.
SKPL-NF-3	Bahasa pemrograman yang digunakan adalah C#.

2.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional bertujuan untuk menguraikan proses kegiatan dan kebutuhan yang akan diterapkan pada sistem yang akan dibangun.

2.5.1 Use Case Diagram

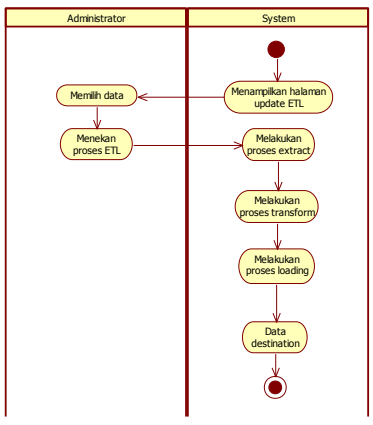
Use case diagram merupakan gambaran fungsionalitas atau perilaku dari sistem. Berikut *use case diagram* pada perangkat lunak dapat dilihat pada Gambar 8:



Gambar 8 *Use Case Diagram*

2.5.2 Activity Diagram

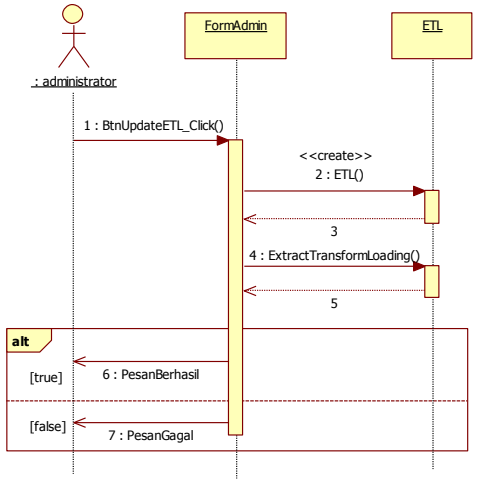
Activity diagram memodelkan aliran kerja dari urutan aktivitas dalam suatu proses yang mengacu pada *use case diagram* yang ada. *Activity diagram update* ETL merupakan aktifitas yang dilakukan oleh *user* untuk memasukkan data baru ke dalam *data mart*. *Activity diagram* dari *update* ETL dapat dilihat pada Gambar 9 :



Gambar 9 *Activity Diagram Update* ETL

2.5.3 Sequence Diagram

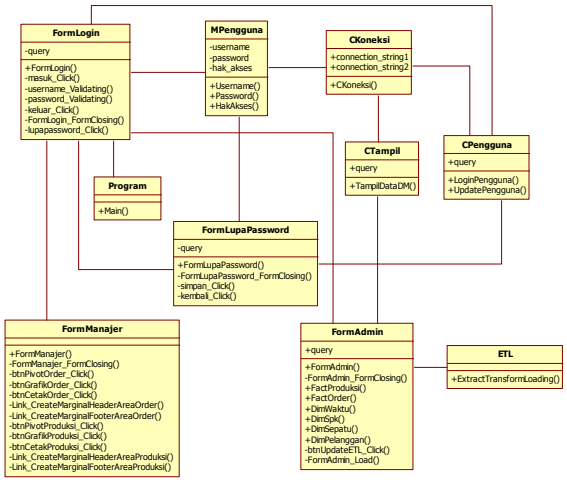
Sequence diagram menggambarkan interaksi antar masing-masing objek pada setiap *use case* dalam urutan waktu. *Sequence diagram* untuk *update* ETL dapat dilihat pada Gambar 10 :



Gambar 10 *Sequence Diagram Update* ETL

2.5.4 Class Diagram

Class diagram menggambarkan struktur dan hubungan antar objek-objek yang ada pada sistem. Struktur ini meliputi atribut-atribut dan metode-metode yang ada pada masing-masing kelas. *Class diagram* dari perangkat lunak *data mart* yang akan dibangun dapat dilihat pada Gambar 11:



Gambar 11 Class Diagram

2.6 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan aplikasi DBMS *Microsoft SQL Server 2012*. Implementasi basis data dalam bahasa *SQL* adalah sebagai berikut :

Tabel 12 Implementasi Tabel dim_pelanggan

Dimensi Pelanggan

```
CREATE TABLE [dbo].[dim_pelanggan](
	[id_pelanggan] [int] NOT NULL,
	[nama] [varchar](30) NOT NULL,
 CONSTRAINT [PK_dim_pelanggan] PRIMARY KEY
CLUSTERED
(
	[id_pelanggan] ASC
)WITH (PAD_INDEX = OFF,
STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS
= ON) ON [PRIMARY]
) ON [PRIMARY]

GO
```

Tabel 13 Implementasi Tabel dim_sepatu

Dimensi Sepatu

```
CREATE TABLE [dbo].[dim_sepatu](
	[id_sepatu] [int] NOT NULL,
	[model_sepatu] [varchar](10) NOT NULL,
 CONSTRAINT [PK_dim_sepatu] PRIMARY KEY
CLUSTERED
(
	[id_sepatu] ASC
)WITH (PAD_INDEX = OFF,
STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY
= OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS
= ON) ON [PRIMARY]
) ON [PRIMARY]

GO
```

Tabel 14 Implementasi Tabel fact_produksi

Fakta Produksi

```
CREATE TABLE [dbo].[fact_produksi](
	[id_spk] [int] NOT NULL,
	[id_sepatu] [int] NOT NULL,
	[id_ukuran] [int] NOT NULL,
	[jumlah] [int] NOT NULL,
	[id_waktu] [int] NOT NULL
) ON [PRIMARY]

GO

ALTER TABLE [dbo].[fact_produksi]  WITH CHECK
ADD  CONSTRAINT [FK_fact_produksi_dim_sepatu]
FOREIGN KEY([id_sepatu])
REFERENCES [dbo].[dim_sepatu] ([id_sepatu])
GO

ALTER TABLE [dbo].[fact_produksi] CHECK
CONSTRAINT [FK_fact_produksi_dim_sepatu]
GO

ALTER TABLE [dbo].[fact_produksi]  WITH CHECK
ADD  CONSTRAINT [FK_fact_produksi_dim_spk]
FOREIGN KEY([id_spk])
REFERENCES [dbo].[dim_spk] ([id_spk])
GO

ALTER TABLE [dbo].[fact_produksi] CHECK
CONSTRAINT [FK_fact_produksi_dim_spk]
GO

ALTER TABLE [dbo].[fact_produksi]  WITH CHECK
ADD  CONSTRAINT [FK_fact_produksi_dim_ukuran]
FOREIGN KEY([id_ukuran])
REFERENCES [dbo].[dim_ukuran] ([id_ukuran])
GO
```

```
ALTER TABLE [dbo].[fact_produksi] CHECK
CONSTRAINT [FK_fact_produksi_dim_ukuran]
GO

ALTER TABLE [dbo].[fact_produksi]  WITH CHECK
ADD  CONSTRAINT [FK_fact_produksi_dim_waktu]
FOREIGN KEY([id_waktu])
REFERENCES [dbo].[dim_waktu] ([id_waktu])
GO

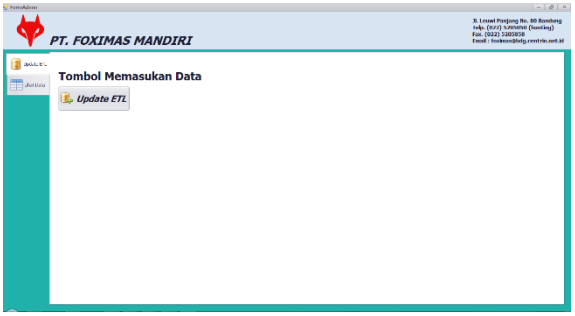
ALTER TABLE [dbo].[fact_produksi] CHECK
CONSTRAINT [FK_fact_produksi_dim_waktu]
GO
```

2.7 Implementasi Antar Muka

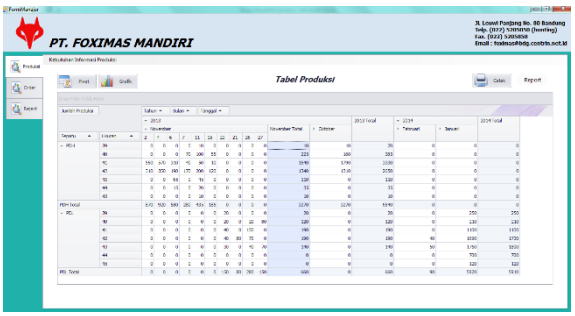
Implementasi antar muka pada perangkat lunak ini terdiri dari beberapa halaman. Berikut implementasi antar muka perangkat lunak *data mart*.



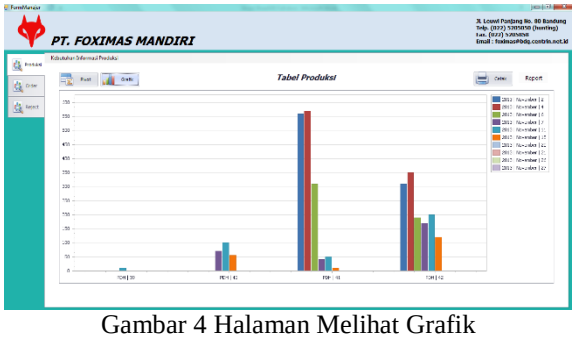
Gambar 1 Halaman Login



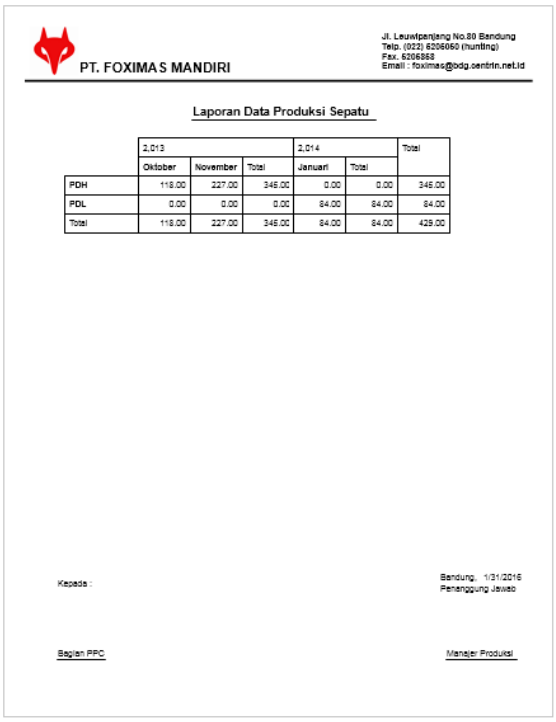
Gambar 2 Halaman *Update* ETL



Gambar 3 Halaman Menganalisis *Data Mart*



Gambar 4 Halaman Melihat Grafik



	2,013			2,014		Total
	Oktober	November	Total	Januari	Total	
PDH	118.00	227.00	345.00	0.00	0.00	345.00
PDL	0.00	0.00	0.00	84.00	84.00	84.00
Total	118.00	227.00	345.00	84.00	84.00	429.00

Gambar 5 Halaman Mencetak Laporan

3. PENUTUP

3.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan mengenai analisis, perancangan, implementasi, dan pengujian terhadap perangkat lunak *data mart* untuk divisi produksi di PT. Foximas Mandiri ini, maka dapat disimpulkan sebagai berikut :

1. Penerapan *Data Mart* memberikan kemudahan dalam mendapatkan informasi strategis dengan cepat yang diperlukan oleh manajer produksi PT. Foximas Mandiri, dan dapat membantu proses pengambilan keputusan sehingga mempercepat proses penganalisaan yang dibuat oleh manajer produksi untuk penanganan produk yang sudah diproduksi.
2. Penerapan *Data Mart* membantu dalam pembuatan laporan akhir produksi yang dibutuhkan oleh manajer produksi di PT. Foximas Mandiri.

3.2 Saran

Berikut ini setelah pembangunan perangkat lunak *data mart* pada PT. Foximas Mandiri maka beberapa saran dalam pengembagan ke depannya adalah sebagai berikut :

1. Sistem tidak hanya menganalisis Divisi Produksi saja, tetapi menganalisis seluruh divisi yang ada di PT. Foximas Mandiri guna untuk memenuhi informasi strategis perusahaan yang lebih besar.
2. Untuk pengembangan perangkat lunak *data mart* pada PT. Foximas Mandiri dapat dikembangkan dengan menambahkan divisi-divisi yang belum ada dan pembuatan format laporan akhir di sesuaikan dengan divisi masing – masing untuk dijadikan perangkat lunak data *warehouse*.

DAFTAR PUSTAKA

[1] N. Sharma, A. Iyer, R. Bhattacharya, N. Modi and W. Crivelini, Getting Started With Data Warehousing, Canada: IBM, 2012.

[2] M. Golfarelli and S. Rizzi, Data Warehouse Design, Modern Principles and Methodologies, New York: Mc Graw Hill, 2009.

[3] R. Kimball and M. Ross, The Data Warehouse Toolkit, Indianapolis: John Wiley & Sons, Inc, 2013.

[4] B. Knight, D. Knight, M. Davis and W. Snyder, Microsoft SQL Server 2012 Integration Services, Indianapolis: John Wiley & Sons, Inc, 2013.