**PEMBANGUNAN PERANGKAT LUNAK DATA *WAREHOUSE* DI SPACEMAN CLOTHING INDONESIA**

**Disusun untuk Menyelesaikan Mata Kuliah Skripsi**

**Semester Genap Tahun Akademik 2014/2015**

**OLEH:**

Ricky Azhari Pratama

10111260



**Program Studi Teknik Informatika**

**Fakultas Teknik Dan Ilmu Komputer**

**Universitas Komputer Indonesia**

**2015**

# DAFTAR ISI

[DAFTAR ISI 1](#_Toc419321019)

[DAFTAR GAMBAR 4](#_Toc419321020)

[DAFTAR TABEL 6](#_Toc419321021)

[BAB I PENDAHULUAN 9](#_Toc419321022)

[1.1 Latar Belakang Masalah 9](#_Toc419321023)

[1.2 Peruumusan Masalah 10](#_Toc419321024)

[1.3 Maksud dan Tujuan 10](#_Toc419321025)

[1.4 Batasan Masalah 11](#_Toc419321026)

[1.5 Metode Penelitian 11](#_Toc419321027)

[1.5.1 Metode Pengumpulan Data 12](#_Toc419321028)

[1.5.2 Metode Pembangunan Perangkat Lunak 12](#_Toc419321029)

[1.5.3 Metode Pembangunan Data *Warehouse* 14](#_Toc419321030)

[1.5.4 Metode Pemodelan *Fuzzy* Data *Warehouse* 15](#_Toc419321031)

[1.6 Sistematika Penulisan 17](#_Toc419321032)

[BAB II TINJAUAN PUSTAKA 18](#_Toc419321033)

[2.1 Profil Perusahaan 18](#_Toc419321034)

[2.1.1 Visi dan Misi Perusahaan 18](#_Toc419321035)

[**2.1.1.1** **Visi** 18](#_Toc419321036)

[**2.1.1.2** **Misi** 19](#_Toc419321037)

[2.1.2 Struktur Organisasi 19](#_Toc419321038)

[2.1.3 Deskripsi Kerja 20](#_Toc419321039)

[2.2 Landasan Teori 21](#_Toc419321040)

[2.2.1 *Database* 21](#_Toc419321041)

[**2.2.1.1** **Skema Diagram** 22](#_Toc419321042)

[2.2.2 Data *Warehouse* 22](#_Toc419321043)

[**2.2.2.1** **Karakteristik Data *Warehouse*** 22](#_Toc419321044)

[2.2.2.2 Arsitektur Data *warehouse* 23](#_Toc419321045)

[2.2.2.3 Skema *Snowflake* 24](#_Toc419321046)

[2.2.2.4 *ETL ( Extract, Transform, Loading)* 24](#_Toc419321047)

[2.2.2.5 *OLAP (On-Line Analytical Processing)* 25](#_Toc419321048)

[2.2.3 *Fuzzy Data Warehouse* 27](#_Toc419321049)

[2.2.3.1 Konsep *Fuzzy Data Warehouse* 27](#_Toc419321050)

[2.2.3.2 Meta Model *Fuzzy Data Warehouse* 28](#_Toc419321051)

[2.2.3.3 Tipe Konsep *Fuzzy* 29](#_Toc419321052)

[2.2.4 *Object Oriented Analysis and Design (OOAD)* 31](#_Toc419321053)

[BAB III ANALISIS DAN PERANCANGAN DATA WAREHOUSE 33](#_Toc419321054)

[3.1 Analisis Data *Warehouse* 33](#_Toc419321055)

[3.1.1 Analisis Masalah 33](#_Toc419321056)

[3.1.2 Analisis Sumber Data 33](#_Toc419321057)

[3.1.3 Analisis Kebutuhan Informasi Strategis 42](#_Toc419321058)

[3.1.4 Analisis Data *Staging* 42](#_Toc419321059)

[3.1.5 Analisis *Fuzzy Dimensional* 51](#_Toc419321060)

[3.1.6 Analisis Dimensi dan Fakta Bisnis 55](#_Toc419321061)

[**3.1.6.1** **Dimensi** 57](#_Toc419321062)

[**3.1.6.2** **Fakta** 60](#_Toc419321063)

[3.2 Analisis OLAP dan Reporting Tools 67](#_Toc419321064)

[3.2.1 Analisis OLAP 67](#_Toc419321065)

[3.3 Analisis Kebutuhan 67](#_Toc419321066)

[3.3.1 Analisis Kebutuhan Non Fungsional 68](#_Toc419321067)

[3.3.1.1 Analisis Perangkat Keras 68](#_Toc419321068)

[3.3.1.2 Analisis Perangkat Lunak 69](#_Toc419321069)

[3.3.1.3 Analisis Perangkat Pikir 70](#_Toc419321070)

[3.3.2 Analisis Kebutuhan Fungsional 71](#_Toc419321071)

[3.3.2.1 *Use Case Diagram* 71](#_Toc419321072)

[3.3.2.2 *Use Case Scenario* 71](#_Toc419321073)

[3.3.2.3 *Activity Diagram* 77](#_Toc419321074)

[3.4 Analisis Model 80](#_Toc419321075)

[3.4.1 Realisasi *Use Case* Tahap Analisis 80](#_Toc419321076)

[3.4.2 Kelas Diagram Keseluruhan 83](#_Toc419321077)

[3.4.3 Analisis Kelas 83](#_Toc419321078)

[3.5 Model Perancangan 85](#_Toc419321079)

[3.5.1 Realisasi *Use Case* Tahap Perancangan 85](#_Toc419321080)

[3.5.2 Perancangan Kelas 90](#_Toc419321081)

[3.4 Perancangan Antarmuka 93](#_Toc419321082)

[BAB IV 94](#_Toc419321083)

[BAB V 95](#_Toc419321084)

[DAFTAR PUSTAKA 95](#_Toc419321085)

# DAFTAR GAMBAR

[Gambar 1 *Waterfall model* [3] 13](file:///D:\MK\VIII\Skripsi\Masterpieces\Dokument\Laporan%201.1.docx#_Toc419382597)

[Gambar 2 *Kimball Life Cycle Data Track Diagram* [4] 15](file:///D:\MK\VIII\Skripsi\Masterpieces\Dokument\Laporan%201.1.docx#_Toc419382598)

[Gambar 3 Metode Pemodelan *Fuzzy* Data *Warehouse* 17](file:///D:\MK\VIII\Skripsi\Masterpieces\Dokument\Laporan%201.1.docx#_Toc419382599)

[Gambar 4 Struktur organisasi Spaceman Clothing Indonesia 19](file:///D:\MK\VIII\Skripsi\Masterpieces\Dokument\Laporan%201.1.docx#_Toc419382600)

[Gambar 5 Komponen arsitektur dalam *three major areas* 24](file:///D:\MK\VIII\Skripsi\Masterpieces\Dokument\Laporan%201.1.docx#_Toc419382601)

[Gambar 6 Model meta *fuzzy data warehouse* [5] 30](#_Toc419382602)

[Gambar 7 *Open end fuzzy concept* [5] 31](#_Toc419382603)

[Gambar 8 *Limited fuzzy concept* [5] 31](#_Toc419382604)

[Gambar 9 *Adaptive fuzzy concept* [5] 32](#_Toc419382605)

[Gambar 10 Skema diagram *OLTP* 35](#_Toc419382606)

[Gambar 13 Skema dimensional fakta belanja produksi 51](#_Toc419382607)

[Gambar 15 Skema dimensional fakta pembayaran konsumen 51](#_Toc419382608)

[Gambar 14 Skema biaya produksi 52](#_Toc419382609)

[Gambar 16 Skema fakta pemesanan produk 53](#_Toc419382610)

[Gambar 17 skema fakta pengirman produk 54](#_Toc419382611)

[Gambar 18 Skema fakta kepopuler *design* produk 55](#_Toc419382612)

[Gambar 19 Skema fakta kepopuleran produk 56](#_Toc419382613)

[Gambar 20 Skema fakta *size* produk 56](#_Toc419382614)

[Gambar 11 Kurva pemetaan himpunan fuzzy jumlah produk 67](#_Toc419382615)

[Gambar 12 Kurva pemetaan himpunan fuzzy pembayaran konsumen 69](#_Toc419382616)

[Gambar 21 *Use case diagram data warehouse* pada perusahaan Spaceman 74](#_Toc419382617)

[Gambar 22 *Activity diagram ETL* 80](#_Toc419382618)

[Gambar 23 *Activity diagram creating OLAP* 81](#_Toc419382619)

[Gambar 24 *Activity diagram creating report* 81](#_Toc419382620)

[Gambar 25 *Activity diagram print* 82](#_Toc419382621)

[Gambar 26 *Acrivity diagram extraction* 82](#_Toc419382622)

[Gambar 27 *Activity diagram transforming* 83](#_Toc419382623)

[Gambar 28 Diagram kelas pada fungsional ETL 84](#_Toc419382624)

[Gambar 29 Diagram Kelas pada fungsional *Creating OLAP* 84](#_Toc419382625)

[Gambar 30 Kelas diagram pada fungsional *Creating Report* 85](#_Toc419382626)

[Gambar 31 Kelas digram pada fungsional *print* 85](#_Toc419382627)

[Gambar 32 Kelas diagram pada fungsional *extraction* 85](#_Toc419382628)

[Gambar 33 Kelas diagram pada fungsional *transforming* 86](#_Toc419382629)

[Gambar 34 Diagram kelas keseluruhan 86](#_Toc419382630)

[Gambar 35 *Sequence diagram* pada *use case ETL* 89](#_Toc419382631)

[Gambar 36 *Sequence diagram* pada *use case creating OLAP* 90](#_Toc419382632)

[Gambar 37 *Sequence diagram* pada *use case creating report* 90](#_Toc419382633)

[Gambar 38 *Sequence diagram* pada *use case print* 91](#_Toc419382634)

[Gambar 39 *Sequence diagram* pada *use case extraction* 92](#_Toc419382635)

[Gambar 40 *Sequence diagram* pada *use case* *transforming* 92](#_Toc419382636)

# DAFTAR TABEL

[Tabel 1 Struktur data tabel kota 36](#_Toc419381402)

[Tabel 2 Struktur data tabel konsumen 36](#_Toc419381403)

[Tabel 3 Struktur data tabel order\_service 36](#_Toc419381404)

[Tabel 4 Struktur data tabel pemesanan 37](#_Toc419381405)

[Tabel 5 Struktur data tabel perencanaan 37](#_Toc419381406)

[Tabel 6 Struktur data tabel pengiriman 38](#_Toc419381407)

[Tabel 7 Struktur data tabel metode\_pengiriman 38](#_Toc419381408)

[Tabel 8 Struktur data tabel pembayaran 38](#_Toc419381409)

[Tabel 9 Struktur data tabel order\_size 39](#_Toc419381410)

[Tabel 10 Struktur data tabel size 39](#_Toc419381411)

[Tabel 11 Struktur data tabel design\_order 39](#_Toc419381412)

[Tabel 12 Struktur data tabel design 40](#_Toc419381413)

[Tabel 13 Struktur data tabel produksi 40](#_Toc419381414)

[Tabel 14 Struktur data tabel job 40](#_Toc419381415)

[Tabel 15 Struktur data staff\_produksi 41](#_Toc419381416)

[Tabel 16 Struktur data tabel bagian 41](#_Toc419381417)

[Tabel 17 Struktur data tabel satuan 41](#_Toc419381418)

[Tabel 18 Struktur data tabel detail\_belanja 42](#_Toc419381419)

[Tabel 19 Struktur data tabel belanja 42](#_Toc419381420)

[Tabel 20 Struktur data tabel kredit\_belanja 42](#_Toc419381421)

[Tabel 21 Struktur data tabel produk 43](#_Toc419381422)

[Tabel 22 Dimensi bisnis Perusahaan Spaaceman 44](#_Toc419381423)

[Tabel 23 Struktur data dim\_design 45](#_Toc419381424)

[Tabel 24 Hirarki design 46](#_Toc419381425)

[Tabel 25 Sturktur data dim\_job 46](#_Toc419381426)

[Tabel 26 Struktur data dim\_konsumen 46](#_Toc419381427)

[Tabel 27 Struktur data dim\_kredit 46](#_Toc419381428)

[Tabel 28 Struktur data dim\_pengiriman 47](#_Toc419381429)

[Tabel 29 Sturktur data dim\_produk 47](#_Toc419381430)

[Tabel 30 Struktur data dim\_size 47](#_Toc419381431)

[Tabel 31 Struktur data dim\_staff\_produksi 48](#_Toc419381432)

[Tabel 32 Struktur data dim\_waktu 48](#_Toc419381433)

[Tabel 33 Struktur data fact\_belanja 48](#_Toc419381434)

[Tabel 34 Struktur data fact\_biaya\_produksi 49](#_Toc419381435)

[Tabel 35 Struktur data fact\_pembayaran\_konsumen 50](#_Toc419381436)

[Tabel 36 Sturktur data fact\_pemesaan 51](#_Toc419381437)

[Tabel 37 Sturktur data fact\_pengiriman\_produk 52](#_Toc419381438)

[Tabel 38 Struktur data pada fact\_pop\_design 53](#_Toc419381439)

[Tabel 39 Struktur data fact\_pop\_produk 54](#_Toc419381440)

[Tabel 40 Struktur data fact\_size\_produk 54](#_Toc419381441)

[Tabel 41 Ekstraksi tabel *OLTP* 55](#_Toc419381442)

[Tabel 42 Proses *Cleaning* dalam transformasi 57](#_Toc419381443)

[Tabel 43 *Conditioning* tabel yang sudah di *cleaning* 62](#_Toc419381444)

[Tabel 44 Pemesanan produk oleh konsumen 65](#_Toc419381445)

[Tabel 45 Kepopuleran produk berdasarkan pemetaan fungsi himpunan 66](#_Toc419381446)

[Tabel 46 Spesifikasi Kebutuhan Fungsional 68](#_Toc419381447)

[Tabel 47 Spesifikasi Kebutuhan Non-Fungsional 69](#_Toc419381448)

[Tabel 48 Spesifikasi perangkat keras yang dibutuhkan 69](#_Toc419381449)

[Tabel 49 Spesifikasi perangkat keras yang ada saat ini 70](#_Toc419381450)

[Tabel 50 Spesifikasi perangkat lunak yang dibutuhkan 70](#_Toc419381451)

[Tabel 51 Spesifikasi perangkat lunak yang ada saat ini 71](#_Toc419381452)

[Tabel 52 Spesifikasi kebutuhan perangkat prikir 71](#_Toc419381453)

[Tabel 53 Spesifikasi perangkat pikir yang ada 71](#_Toc419381454)

[Tabel 54 *Use case scenario ETL* 73](#_Toc419381455)

[Tabel 55 *Use case scenario creating OLAP* 74](#_Toc419381456)

[Tabel 56 *Use case scenario creating report* 75](#_Toc419381457)

[Tabel 57 *Use case scenario print* 75](#_Toc419381458)

[Tabel 58 *Use case scenario extraction* 76](#_Toc419381459)

[Tabel 59 *Use case scenario transforming* 77](#_Toc419381460)

[Tabel 60 Jenis kelas yang digunakan di dalam sistem 85](#_Toc419381461)

[Tabel 61 Tanggung jawab kelas 85](#_Toc419381462)

[Tabel 62 Identifikasi perancangan kelas pada *use case ETL* 87](#_Toc419381463)

[Tabel 63 Identifikasi perancangan kelas pada *use case creating OLAP* 87](#_Toc419381464)

[Tabel 64 Identifikasi perancangan kelas pada *use case creating report* 88](#_Toc419381465)

[Tabel 65 Identifikasi perancangan kelas pada *use case print* 89](#_Toc419381466)

[Tabel 66 Identifikasi perancangan kelas pada *use case extraction* 89](#_Toc419381467)

[Tabel 67 Identifikasi perancangan kelas pada *use case transforming* 90](#_Toc419381468)

[Tabel 68 Perancangan operasi pada kelas *user kontrol* 91](#_Toc419381469)

[Tabel 69 perancangan operasi pada kelas *ETL* 91](#_Toc419381470)

[Tabel 70 perancangan operasi kelas dimensi 92](#_Toc419381471)

[Tabel 71 Perancangan atribut kelas dimensi 92](#_Toc419381472)

[Tabel 72 perancangan operasi pada kelas fakta 92](#_Toc419381473)

[Tabel 73 perancangan atribut pada kelas fakta 93](#_Toc419381474)

[Tabel 74 Perancangan operasi pada kelas *OLAP* 93](#_Toc419381475)

[Tabel 75 Perancangan operasi pada kelas koneksi 93](#_Toc419381476)

[Tabel 76 Perancangan atribut pada kelas koneksi 94](#_Toc419381477)

# BAB I PENDAHULUAN

# 1.1 Latar Belakang Masalah

Spaceman Clothing Indonesia merupakan salah satu perusahaan yang berkembang di Bandung yang bergerak di bidang jasa pengadaan barang hasil olahan konveksi, dimana produk yang dihasilkannya berupa pakaian seperti kemeja, kaos, jaket, dan berbagai seragam instansi ataupun seragam lainnya yang beralamat di jalan Cigadung Raya Timur No. 5 Kecamatan Cimeunyan Bandung. Perusahaan ini bertujuan untuk memberikan pelayanan kepada konsumen dengan memproduksi produk yang dipesan konsumen dengan baik dan terjaga kualitasnya. Konsumen yang telah menggunakan jasa perusahaan ini cukup beragam dan tersebar di wilayah Indonesia baik itu distro clothing, lembaga instansi, maupun sekolah. Perusahan ini telah berdiri selama enam tahun dan memiliki tiga bangunan utama, yaitu bangunan administrasi dan perencanaan, bangunan produksi dan pengendalian kualitas produk, dan bangunan desain produksi.

Sistem yang telah berjalan di dalam perusahaan ini dapat digolongkan sudah menggunakan teknologi dengan menerapkan penggunaan komputer dalam pendataan secara terpisah di setiap bagian dalam perusahaan ini dengan penginputan data secara manual. Kendala proses sistem yang berjalan di temukan seperti desain data yang kurang efektif untuk di analisis, Sulitnya menganalisis laporan dikarenakan laporan yang ada tidak dapat menampilkan informasi dari berbagai aspek fakta penting yang disebut multidimensional yang diperlukan, seperti menganalisa laporan pembayaran konsumen yang memesan produk dalam periode waktu tertentu, kerugian dalam proses produksi dalam periode waktu tertentu, atau keuntungan yang dihasilkan dalam proses produksi dalam periode waktu tertentu. Kendala juga terjadi ketika perbandingan data yang di hasilkan dalam keperluan analisis tidak dapat menyampaikan informasi yang semestinya. Seperti pada kendala pelunasan pembayaran produk yang dipesan konsumen, bila suatu konsumen mengalami telat dalam pelunasan maka akan menimbulkan rentang pembayaran terhadap tanggal semestinya yang akan digunakan dalam pembuatan laporan. Dalam data yang dihasilkan pada periode tertentu, rentang pembayaran yang seharusnya menginformasikan akan menjadi kurang optimal.

Sebagai solusi dari kendala dari permasalahan yang ada, maka perlu dibangunnya perangkat lunak data *warehouse* untuk memudahkan pimpinan dalam menganalisa data untuk mendapatkan informasi mengenai kepopuleran produk yang dipesan konsumen maupun pembayaran konsumen dalam pemesanan barang yang telah di lakukan oleh konsumen. Solusi kendala juga membutuhkan suatu metode agar mendapatkan informasi yang optimal. Dalam penelitian ini metode yang diterapkan dalam pembangunan data *warehouse* adalah menggunakan logika fuzzy. Penerapan logika fuzzy dalam penelitian ini dapat membantu pimpinan lebih mudah dalam mengartikan suatu informasi dan lebih intuitif dalam suatu penganalisisan [1].

# 1.2 Peruumusan Masalah

Berdasarkan latar belakang masalah maka dirumuskan masalah yaitu bagaimana membangun perangkat lunak *data warehouse* dengan menggunakan konsep *fuzzy dimensional*.

# 1.3 Maksud dan Tujuan

Adapun maksud yang ingin dicapai dalam menyelesaikan permasalahan adalah dengan membangun perangkat lunak data *warehouse* yang dapat mengelola data dari berbagai sumber secara valid dan akurat.

Adapun tujuan untuk dicapai dalam penyelesaian masalah yang telah dipaparkan adalah:

1. Membantu pimpinan perusahan dalam menganalisis suatu informasi dengan mengoptimalkan desain data dalam perusahan seperti :
   1. Kepopuleran produk yang dipesan oleh konsumen.
   2. Pemesanan produk yang dilakukan oleh konsumen
   3. Pembayaran terhadap pesanan yang dilakukan oleh konsumen.
   4. Pengiriman produk yang digunakan untuk mengirim produk ke konsumen.
   5. Pembelanjaan produksi terhadap pemesanan konsumen.
   6. Biaya produksi pada pesanan konsumen.
   7. Kepopuleran desain yang digunakan pada produk.
   8. *Size* yang telah dipesan per produk.

2. memodelkan data secara multidimensional yang dapat digunakan secara dinamis sesuai dengan informasi yang dibutuhkan.

3. Menerapkan metode logika *fuzzy* dalam pembangunan perangkat lunak.

# 1.4 Batasan Masalah

Batasan masalah dalam pembangun perangkat lunak data *warehouse* ini adalah sebagai berikut:

1. *Data Base Management System*  yang digunakan adalah Microsoft SQL Server 2014.
2. *Fuzzy Data warehouse* yang digunakan berbasis *meta table structure*.

3. Penganalisisan data *warehouse* menggunakan OLAP (*On-line Analytical Processing* ).

4. Analisis pembangunan perangkat lunak menggunakan *Object Oriented*.

# 1.5 Metode Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah metode deskriptif komparatif. Metode ini merupakan salah satu metode penelitian deskriptif yang bertujuan mencari jawaban atau fakta secara mendasar dengan menganalisis faktor-faktor penyebab terjadinya ataupun munculnya suatu fenomena [2]. Metode ini menggunakan teknik korelasi, yaitu meneliti ketergantungan dalam hubungan antar variabel [2].

# 1.5.1 Metode Pengumpulan Data

**1. Wawancara**

Wawancara merupakan teknik pengumpulan data yang dilakukan dengan cara komunikasi langsung dengan responden, dimana komunikasi berupa pertanyaan dan jawaban yang bertujuan untuk menemukan permasalahan yang harus diteliti [2].

1. **Kuesioner**

Kuesioner merupakan teknik pengumpulan data dimana responden mengisi pertanyaan atau pernyataan yang disediakan dan di kembalikan kembali ke peneliti setelah terisi semua [2].

1. **Observasi**

Teknik perngumpulan data dengan cara mengamati langsung pola prilaku individu dalam situasi tertentu guna mendapatkan informasi tentang fenomena yang diinginkan [2].

1. **Dokumen**

Teknik pengumpulan data dengan cara mempelajari dan mengkaji catatan peristiwa yang sudah berlalu. Dalam penelitian ini dokumen yang di amati adalah catatan transaksi perusahaan baik itu berupa laporan, catatan keuangan maupun kepegawaian.

# 1.5.2 Metode Pembangunan Perangkat Lunak

Metode yang digunakan dalam pembangunan perangkat lunak dalam penelitian ini menggunakan *waterfall model* atau *linear sequential model.* Model ini menyediakan pendekatan yang sistematis dan berurutan bagi pengembang perangkat lunak. Tahap – tahap pengembangang perangkat lunak dengan model ini adalah :

1. *Requirements Analysis and Definition*

Proses pengumpulan kebutuhan secara lengkap dalam tahap konsultasi dengan pengguna sistem yang dikemudian digunakan sebagai bahan analisis dan pendefinisian spesifikasi sistem [3].

2. *System and Software Design*

Proses *software design* mengalokasikan segala persayaratan yang dibutuhkan baik dalam sisi perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan [3]. *Software design* merupakan proses yang melibatkan suatu identifikasi dan penggambaran abstrak dalam perwujudan dasar sistem dan relasinya [3].

3. *Implementation and Unit Testing*

Tahap ini merupakan tahap merealisasikan desain sistem sebagai serangkaian program atau unit program [3]. Sedangkan *unit testing* tahap untuk memverifikasi setiap unit terhadap spesifikasi kebutuhan sistem [3].

4. *Integration and System Testing*

Tahap pengitegrasian unit program menjadi satu program individu dan diuji sebagai sistem yang lengkap untuk memastikan bahwa persayaratan perangkat lunak telah terpenuhi [3]

5. *Operation and Maintenance*

Tahap sistem di-instalasi dan di pakai sebagai praktek untuk memenuhi kebutuhan yang seharusnya [3]. *Maintenance* merupakan proses mengoreksi kesalahan yang tidak ditemukan pada tahap siklus sebelumnya [3].

Requirements definition

System and software design

Implementation and unit testing

Integration and System Testing

Operation and Maintenance

Gambar 1 *Waterfall model* [3]

# 1.5.3 Metode Pembangunan Data *Warehouse*

Metode yang digunakan dalam pembangunan data *warehouse* adalah metode *kimball*  yang memiliki tahapan umum dalam pembangunan data warehouse. Tahap-tahap dalam metode *kimball* yang digunakan dalam pembangunan data *warehouse* adalah sebagai berikut :

1. *Business Requirements Definition*

Tahap ini merupakan tahap kolaborasi atau diskusi dengan pengguna bisnis untuk menentukan dan memahami kebutuhan yang diinginkan dalam mencapai tujuan data *warehousing* [4]. Tahap ini hanya fokus pada pengumpulan kebutuhan-kebutuhan bisnis yang diperlukan [4].

1. *Dimensional Modelling*

Tahap ini merupakan tahap pemodelan dengan malakukan *review* terhadap kebutuhan bisnis pengguna, pemanfaatan alat pemodelan, pembangunan konvensi penamaan, koordinasi waktu yang dibutuhkan, dan perancangan pemodelan *dimensional* [4].

1. *Physical Design*

Tahap ini merupakan tahap pengembangan *dimensional*  model dan di dokumentasikan melalui sumber awal menuju ke targer pemetaan yang diterjemahkan ke dalam *phisical database* [4]. Tahap ini melakukan penavigasian pada model data dalam bentuk elemen tabel dan nama kolom yang mengandung arti dalam suatu bisnis, membangun pemodelan *database* yang bersifat fisik yang akan digunakan dan mendukung dalam pembangunan sistem ETL, mempersiapkan perencanaan inisalisasi index dalam tabel, dan merancang agregasi dalam suatu relasi pada perancangan fisik [4]. Hal ini bertujuan untuk untuk memberikan pemuatan data, pemeliharaan, dan kinerja *query* menjadi lebih efisien.

1. *ETL Design and Development*

Tahap ini merupakan tahap pemudelan dan pengembangan proses yang dibutuhkan dalam *warehousing* dengan pertimbangan terhadap data yang dibutuhkan dalam suatu bisnis, kualitas suatu data, kemanan data, integrasi data, *latency*  pada pengiriman sumber data yang harus dikirim ke sistem data *warehouse*, dan melakukan pengarsipan data [4]

1. *Deployment*

Tahp ini merupakan tahap penyajian penyebaran data yang dilakukan pada tahap sebelumnya. Dalam tahapan ini hal-hal yang bersifat kritis adalah melakukan pengujian sistem *end-to-end*, jaminan kualitas data, pengolahan operasi, kinerja sistem, dan pengujian kegunaan suatu sistem [4].

Gambar 2 *Kimball Life Cycle Data Track Diagram* [4]

Business Requirements Definition

Dimensional Modeling

Physical Design

ETL Design & Development

Deployment

# 1.5.4 Metode Pemodelan *Fuzzy* Data *Warehouse*

Dalam pembangunan suatu *fuzzy* data *warehouse*, dibutuhkan suatu metode yang dapat mentransformasi suatu data *crisp* dalam data *warehose* menjadi  *fuzzy* data *warehouse* dimana nilai masukan awal pada metode ini adalah data *warehouse* klasik dan keluarannya merupakan  *fuzzy*  data  *warehouse* [5]*.* Adapun langkah dapal metode ini adalah sebagai berikut:

1. *Defining Classification Elements*

Tahapan ini merupakan tahapan yang mendefensiskan kalsifikasi element yang akan digunakan pada tahapan selanjutnya dalam pembangunan suatu model *fuzzy* data *warehouse*. Tahapan ini melibatkan tiga langkah, yaitu:

1. *Identify Target Attribute*

Langkah ini merupakan langkah untuk mengidentifikasikan atribut target yang memiliki nilai-nilai yang akan diklasifikan secara *fuzzy* [5].

1. *Identify Linguistic Term*

Langkah ini merupakan langkah untuk mengidentifikasi *linguistik term* yang digunakan untuk mengklasifikasi target atribut dan menentukan nilai target atribut yang harus diklasifikan dalam suatu *lingusitic term* [5]*.*

1. *Define Membership Function*

Langkah ini merupakan langkah untuk mendefinisikan fungsi keanggotaan untuk setiap istilah linguistik yang telah ditetapkan [5]. Langkah ini dilakukan sedemikian rupa sehingga nilai dari atribut target dapat dipetakan melalui skala 0 sampai 1 [5].

1. *Building Fuzzy DWH Model*

Tahap ini merupakan tahap menggunakan element klasifikasi yang didapatkan pada tahap sebelumnya dalam membangun model *fuzzy* data *warehouse* [5]. Tahap ini melibatkan tiga langkah, yaitu:

1. *Model FCT*

Langkah ini merupakan langkah untuk memodelkan suatu tabel atau atribut klasifikasi pada setiap set dari *linguistic term* [5].

1. *Model FMT*

Langkah ini merupakan langkah untuk membuat tabel keanggotaan *fuzzy* [5]*.* Atribut yang ada di dalam tabel ini adalah atribut identifikasi (*primary key*), atribut identifikasi dari target atribut, atribut untuk klasifikasi *fuzzy*, dan atribut derajat keanggotaan dari target atribut [5]. Nilai dari atribut derajat keanggotaan dikalkulasikan berdasarkan fungsi keanggotaan yang telah ditetapkan [5].

1. *Relate FCT, FMT & TA*

Langkah ini merupakan langkah untuk merelasikan atau menghubungkan tabel keanggotaan fuzzy dan pengkalisifikasiaannya dengan tabel yang berisi atribut target [5]. Dalam perelasian ini, dibutuhkan penambahan kunci identifikasi dari target atribut yang direlasikan pada tabel *fuzzy* sebagai kunci tamu [5].

*Defining Classification Elements*

*Identify Target Attribute*

*Identify Linguistic Term*

*Identify Membership Function*

*Building Fuzzy DWH Model*

*Model FCT*

*Model FMT*

*Relate FCT, FMT & TA*

Gambar 3 Metode Pemodelan *Fuzzy* Data *Warehouse*

# 1.6 Sistematika Penulisan

Sistematika penulisan penelitian ini disusun agar mendapatkan gambaran umum tentang penelitian yang dijalankan. Sistematika penulisan penelitian ini adalah sebagai berikut :

**BAB I PENDAHULUAN**

Bab ini menguraikan tentang latar belakang permasalahan, mencoba merumuskan masalah, menentukan maksud dan tujuan, menentukan metodologi peneltian, serta sistematika penulisan.

**BAB II TINJAUAN PUSTAKA**

Bab ini membahas berbagai konsep dasar dan teori-teori yang berkaitan dengan topik penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis permasalahan serta tinjauan terhadap penelitian-penelitian serupa yang telah pernah dilakukan sebelumnya termasuk sintesisnya. Membahas tentang konsep dasar serta teori-teori yang berkaitan dengan topik penelitian dan yang melandasi rancang bangun sistem.

**BAB III ANALISIS DAN PERANCANGAN SISTEM**

Pada Bab ini berisi tentang analisis data *warehouse*, analisis *OLAP* dan *reporting tool*, analisis kebutuhan fungsional dan *non* fungsional, dan perancangan sistem.

**BAB IV IMPLEMENTASI DAN PENGUJIAN**

Pada Bab ini berisi pembahasan implementasi serta penjelasan tentang teknik dan strategi pengujian sistem yang digunakan.

**BAB V KESIMPULAN DAN SARAN**

Pada Bab ini berisi tentang kesimpulan dan saran untuk pengembangan sistem selanjutnya.

# BAB II TINJAUAN PUSTAKA

# 2.1 Profil Perusahaan

Spaceman Clothing Indonesia merupakan perusahaan yang bergerak dalam bidang jasa penyediaan barang hasil olahan konveksi seperti kaos, kemeja, polo, jaket, dan lain-lain. Perusahaan ini didirikan sejak tahun 2010 yang hingga saat ini terus berkembang dan tetap aktif melayani setiap pemesenan konsumen di seluruh wilayah Indonesia yang menggunakan jasa dari perushaan ini. Perusahaan ini beralamat di jalan Cigadung Raya Timur No. 5 Kecamatan Cimeunyan Bandung.

# 2.1.1 Visi dan Misi Perusahaan

Adapan visi dan misi dari perusahaan ini adalaha sebagai berikut.

1. Visi

Menjadi perusahaan yang berskala nasional dan internasional yang mengedepankan kualitas, inovasi, dan profesionalisme dalam setiap kinerja team dalam upaya manjadi yang terdepan di bidangnya.

1. Misi
2. Mengangkat *prestise (image)* produk lokal sehingga dapat diperhitungkan dalam kancah industri global.
3. Menjadi perusahaan *clothing manufacture* yang menghasilkan produk berkualitas, inovatif, dan mengikuti perkembangan zaman yang dikonsumsi oleh pasar lokal regional, nasional, dan internasional.
4. Menjadi perusahaan yang dapat menyerap lebih banyak tenaga kerja profesional yang kompeten. Sehingga turut berpartisipasi dalam program pemerintah dalam penekanan tingkat jumlah pengangguran.

# 2.1.2 Struktur Organisasi

Struktur Organasi merupakan pola hubungan antar bagian dari instansi atau menggambarkan dengan jelas pemisahan kegiatan pekerja antara bagian yang satu dengan bagian yang lain dalam suatu instnasi. **Gambar 2** menjelaskan struktur organisasi yang ada di Spaceman Clothing Indonesia.

Gambar 4 Struktur organisasi Spaceman Clothing Indonesia

Owner

Pengawas Produksi

Administrasi dan Perencanaan

Kepala Penjahit

Kepala Kualitas dan desain

Staff Lapangan

Staff Penjahit

Staff Sablon

Staff Cutting

# 2.1.3 Deskripsi Kerja

a. Owner

Owner Memiliki fungsi sebagai pemimpin dan pengelola perusahaan dan bertugas untuk mewujudkan dan menjalankan visi dan misi, mengelola perkembangan perusahaan, keuangan, dan hal-hal yang berkaitan untuk perusahaan.

b. Pengawas Produksi

Pengawas Produksi memiliki fungsi sebagai mengawasi alur produksi terhadap suatu order dan memiliki tanggung jawab mengontrol proses order konsumen yang sedang dikerjakan dan mengontrol perpindahan bahan dari satu proses ke proses yang lain, dan mengontrol alat dan bahan yang dibutuhkan dalam suatu proses produksi terhadap suatu order.

c. Administrasi dan Perencanaan

Administrasi dan Perencanaan memiliki fungsi sebagai pengelola hal-hal bersifat dokumentasi dan estimasi terhadap suatu order dari konsumen dan memiliki tanggung jawab untuk membuat suat estimasi order, keperluan pembelanjaan bahan, pengecekan pembayaran, dan pembuatan laporan setiap order produksi.

d. Kepala Penjahit

Kepala penjahit memiliki fungsi sebagai pengelola distribusi tugas staff penjahit dan memiliki tanggung jawab untuk mengawas dan membantu staff penjahit dalam mengerjakan suatu produk yang di pesan oleh konsumen, memeriksa kelengkapan bahan yang akan dijahit, mengontrol kelengkapan peralatan yang digunakan selama menjahit, dan mengontrol kelayakan mesin untuk digunakan menjahit.

e. Kepala Kualitas dan Desain

Kepala kualitas dan design memiliki fungsi sebagai pengelola distribusi tugas staff sablon dan memiliki tanggung jawab untuk mengawas dan membantu staff sablon dalam mengerjakan suatu produk, membuat kalkir dari desain yang akan digunakan dalam proses sablon, mengontrol kelengkapan bahan yang digunakan, mengontrol kelayakan alat yang digunakan, dan mencocokkan warna yang akan digunakan.

f. Staff Penjahit

Staff penjahit memiliki fungsi sebagai penjahit yang menjaga kualitas dalam pengerjaannya dan memiliki tanggung jawab untuk mengambil tugas yang di berikan oleh kepala penjahit.

g. Staff sablon

Staff sablon memiliki fungsi sebagai tenaga keraja yang mengerjakan desain dari pakaian yang mempertahankan kualitas dalam pengerjaannya dan memiliki tanggung jawab untuk mengambil tugas yang di berikan oleh kepala kualitas dan desain.

h. Staff Cuting

Staff cuting memilki fungsi sebagai pemotong bahan yang digunakan dalam suatu pesanan sesuai dengan kebutuhan yang diperlukan dan memiliki tanggung jawab untuk membuat pola potong baru, memotong bahan yang digunakan seoptimal mungkin dengan penggunaan bahan, dan mengontrol ketersediaan bahan untuk dipotong.

i. Staff Lapangan

Staff lapangan memiliki fungsi sebagai tenaga kerja yang digunakan dalam proses pengadaan alat dan bahan yang di perlukan dalam suatu pesanan konsumen dan memiliki tanggung jawab untuk membeli alat dan bahan yang diperlukan, mengontrol biaya yang tersedia dalam pembelian alat dan bahan, dan memiliki alternatif untuk mengambil keputusan dalam pembelian alat dan bahan.

# 2.2 Landasan Teori

# 2.2.1 *Database*

*Database* merupakan sekumpulan data saling berhubungan yang mampu menyediakan informasi yang relevan untuk suatu organisasi atau *enterprise* [6]*.* Sedangkan alat untuk mengelola dan mengakses *database* disebut *DBMS ( Database Management System)*. Tujuan utama dari *DBMS* adalah menyediakan cara untuk menyimpan dan mengambil informasi dari *database* dengan baik, nyaman, dan efisien [6].

* + - 1. **Skema Diagram**

Skema diagram merupakan gambaran dari skema database bersamaan dengan kunci primer dan depedensinya dengan kunci tamu yang setiap relasinya di wakili sebagai kotak yang di lengkapi dengan nama relasi dan atribut [6]. Atribut yang muncul sebagai atribut kunci primer akan digaris bawahi di dalam kotak dan atribut sebagai kunci tamu muncul dalam atribut di dalam kotak yang memiliki panah dari atribut kunci tamu ke kunci primer sebagai hubungan referensi [6].

# 2.2.2 Data *Warehouse*

Data *warehouse* merupakan suatu sistem yang menyajikan suatu informasi, didalam sistem tersebut terdapat integrasi dan transformasi terhadap data dari berbagai sistem operasional yang ada baik data yang bersumber dari internal maupun eksternal kedalam bentuk suatu informasi yang relevan untuk penunjang keputusan dalam starategi bisnis [7]. Dalam pengartian lain Data *warehouse* juga merupakan *repository* (atau arsip) informasi yang dikumpulkan dari berbagai sumber, disimpan di dalam skema yang *unified* di dalam aplikasi tunggal yang akan disimpan dalam waktu yang lama dan memungkinkan untuk mengakses data historis, sehingga menyediakan informasi untuk pendukung keputusan dari data *warehouse* [6]*.*

* + - 1. **Karakteristik Data *Warehouse***

Karakteristik data didalam data *warehouse* adalah sebagai berikut:

1. Berorientasi subjek

Didalam data *warehouse*, data dibentuk berdasarkan subjek bisnis bukan berdasarkan fungsi pada sistem operasional pada aplikasi tertentu [7]. Sehingga data fokus terhadap suatu subjek yang di anggap kritikal untuk dijadikan informasi sebagai penunjang keputusan yang akan digunakan sebagai strategi bisnis yang diterapkan [7].

1. Terintegrasi

Data didalam data *warehouse* bersumber dari beberapa sistem dimana sumber datanya berada di database, file, atau segementasi data yang berbeda [7]. Syarat integrasi terhadap sumber data dapat dipenuhi dengan cara memperbaiki data yang tidak konsistensi, melakukan standarisasi tehadap elemen data, dan memastikan maksud dari nama-nama variabel yang digunakan di dalam data yang berasal dari setiap sumber yang digunakan [7].

1. *Time Variant*

Data di dalam data *warehouse* merupakan data yang bersumber dari data pada periode yang lalu dan sekarang, dimana setiap struktur data di dalam data *warehouse*  mengandung elemen waktu dikarenakan data di dalam data *warehouse*  dimaksudkan untuk sebagai bahan analisis dan penunjang keputusan yang mengandung tidak hanya sekedar data di masa sekarang, tetapi mengandung data pada masa lampau juga [7].

1. *NonVolatile*

Data operasional dipindahkan ke dalam data *warehouse* dalam spesifikasi interval tertentu, tergantung dari kebutuhan bisnis yang diinginkan [7]. Di dalam data *warehouse* data diekstraksi sebagai *snapshot* dalam periode waktu, bisa dalam periode perminggu, perbulan, atupun pertahun [7]. Setiap data transaksional tidak dapat melakukan pembaharuan, penginputan, ataupun menghapus data secara *real-tme* di dalam data *warehouse* seperti yang dapat dilakukan di sistem operasional melainkan melakukan *refreshing* dari data operasional yang kemudian dimuat kembali ke dalam data *warehouse*.

# 2.2.2.2 Arsitektur Data *warehouse*

Arsitektur Data warehouse merupakan struktur yang menyajikan semua komponen yang terlibat didalam data *warehouse* secara bersamaan [7]. Didalam data *warehouse*, arsitektur termasuk data yang terintegrasi sebagai satuan yang terpusat, semua kebutuhan untuk persiapan data dan penyimpanannya, dan arah penyajian informasi dari data *warehouse*  sehingga menghasilkan suatu aturan, prosedur, dan fungsional untuk memungkinkan data *warehouse* bekerja dan memenuhi kebutuhan bisnis [7]. Salah satu arsitektur yang dapat digunakan adalah arsitektur *three major areas* yang meliputi *data acquisition, data storage,*  dan *information delivery* [7]*.*

*Internal*

*Data Staging*

*Data Source*

*External*

*production*

*Archived*

***Data Acquisition***

*Data warehouse*

*DBMS*

*Data Marts*

*MDDB*

*Meta data*

***Data Storage***

*OLAP*

***Information Delivery***



*Data mining*

*Report / Query*

Gambar 5 Komponen arsitektur dalam *three major areas*

# 2.2.2.3 Skema *Data Warehouse*

Dalam memodelkan suatu sistem data  *warehouse* menggunakan teknik pemodelan *dimensional* atau juga disebut *dimensional model* [7]. Pemodelan ini berguna untuk menetapkan untuk membuat tabel dimensi, tabel fakta dan relasinya [7]. Hasil dari pemodelan ini akan membentuk skema yaitu:

1. *Star Schema*

Skema ini merupakan skema dasar yang terbentuk dari teknik perancangnan dasar untuk data *warehouse*. Skema ini menggambarkan suatu tabel fakta yang ditampilkan di tengah-tengah diagram dan dikelilingi oleh dimensi-dimensi yang berelasi pada tabel fakta [7]. Relasional antar tabel dimensi dengan tabel fakta merupakan relasi satu ke banyak [7]. Skema ini sering digunakan dalam pembuatan sistem data *warehosue* dikarenekan lebih mudah dimengerti oleh pengguna, mengoptimalkna navigasi melalui suatu *database*, dan skema yang paling cocok dalam melakukan *qurey* *processing* [7]*.*

1. *Snowflake Schema*

*“Snowflaking”* merupakan metode normalisasi tabel dimensi dalam skema STAR dimana setelah semua tabel dimensi telah benar-benar dinormalisasikan, struktur yang dihasilkan menyerupai struktur salju dengan tabel fakta di tengah [7]. Prinsip dari *snowflaking* adalah menormalisasi tabel dengan menghapus atribut kardinalitas rendah dan membentuk tabel terpisah. Alasan melakukan *snowflaking* juga adalah untuk penghematan ruang penyimpanan dalam satu tabel dan penelusuran atribut lebih terperinci dalam suatu tabel dimensi [7].

# 2.2.2.4 *ETL ( Extract, Transform, Loading)*

*ETL* merupakan himpunan fungsi yang dilakukan untuk mengubah dan membentuk kembali data ke dalam bentuk yang berbeda pada data di dalam sistem operasional yang disimpan di dalam data *warehouse* sebagai informasi yang relevan dan strategis [7]. Adapun kelompok himpunan *ETL* adalah ekstraksi data, transformasi, dan *loading* yang menjadi tahapan proses pengubahan dan pembentukan ulang data yang digunakan di dalam data *warehouse* [7]*.*

1. *Extraction*

Tahap extraction merupakan tahap untuk mengidentifikasi semua sumber data internal, menentukan proses komputasi dan sumber data mana yang akan di ekstrak, menentukan kompatibilitas struktur data jika dan hanya jika bila menggunakan sumber-sumber dari luar, dan mengindikasi metode untuk mengekstraksi data [7].

1. *Transform*

Tahap *transform* memiliki fungsi meliputi pemilihan input, pemisahan struktur input, normalisasi dan denormalisasi dari struktur data sumber, mengagregasi, mengkonversi, dan memecahkan nilai yang hilang [7].

1. *Loading*

Tahap *loading* merupakan tahap menginisialisasi *load* awal, menentukan seberapa sering suatu kelompok data harus tetap *up-to-date* dalam data *warehouse*, dan menentukan cara mengubah data yang akan dilaksanakan dalam periode waktu tertentu.

# 2.2.2.5 *OLAP (On-Line Analytical Processing)*

*OLAP* merupakan salah satu katergori teknologi perangkat lunak yang memungkinkan analis, manajer, dan eksekutif untuk menggali suatu data secara cepat, konsisten, dan memiliki akses yang interaktif dalam luasnya penggalian suatu informasi yang di transformasikan dari suatu *raw*  data menjadi dimensi fakta yang dapat dimengerti oleh pengguna [7].

1. kriteria *OLAP*

Suatu sistem *OLAP* harus memiliki kriteria yang dapat menghasilkan suatu prioritas yang akan dituju. Adapun kriteria dari *OLAP* adalah sebagai berikut.

1. *Multidimensional Conceptual View*

Suatu model data *multidimensional*  mangacu pada bagaimana peresepsi pengguna untuk memecahkan suatu masalah bisnis yang dapat menciptakan suatu analisis yang intuitif dan mudah digunakan [7].

1. *Transparency*

Teknologi, gudang data, arsitektur komputasi, dan data alami dari sumber ditransparasikan terhadap pengguna, dengan tujuan untuk dapat membantu meningkatkan ke-efisienan dan produktivitas pengguna [7].

1. *Accessibility*

Suatu akses yang dihasilkan untuk menampilkan data harus dapat menghasilkan hasil yang dapat di analis secara spesifik, dipresentasikan sebagai tunggal, konherensi, dan konsisten bagi pengguna yang melihatnya [7].

1. *Consistent Reporting Performance*

*OLAP* memastikan kekonsistenan penggunaan dimensi atau ukuran yang digunakan sehingga pengguna dapat melaporkan kinerja dengan efisien terhadap waktu baik itu waktu dijalankan, waktu respon, atau pemanfaat sistem setiap kali *query* digunakan.

1. *Client/Server Architecture*

Pembangunan *OLAP* sesuai dengan prinsip arsitektur klien atau server agar kinerja optimal, fleksibel, mampu beradaptasi, dan inter-operabilitas [7]. Sehingga pengguna tidak perlu menginputkan *query* lagi dalam pembentukan ulang [7].

1. *Generic Dimensionality*

*OLAP* memastikan terhadap setiap dimensi memiliki kesetaraan dalam struktur, kemampuan operasional, dan memiliki satu struktur yang logis [7].

1. *Multi User support*

*OLAP* memberikan dukungan pada *end user* untuk bekerja secara bersamaan dengan model analisis yang sama atau untuk model yang berbeda pada data yang sama [7].

1. *Intuitive Data Manipulation*

*OLAP* memungkinkan berbagai macam jalur konsolidasi re-orientasi pada pemanipulasian informasi seperti *drill-down, roll-up,* dan manipulasi yang dilakukan secara intuitif dan langsung [7].

1. *Flexible Reporting*

*OLAP* memberikan kebebasan pengguna untuk mengatur dengan mudah suatu kolom, baris dan sel dengan fasilitas manipulasi yang mudah, analisis, dan sisntesis informasi [7].

1. *Unlimited Dimensions and Agregation Levels.*

*OLAP* dapat mengakomodasi banyak dimensi data dalam suatu model umum analisis.

1. Karakteristik *OLAP*

Karakteristik dasar dari *OLAP* adalah sebagai berikut.

1. Memastikan pengguna memiliki suatu *view* yang *multidimensional* dan *logical* dalam suatu data warehouse [7].
2. Memfasilitasi *query* yang interaktif dan analisis yang kompleks untuk pengguna [7].
3. Memungkinkan user untuk melakukan *drill down* untuk mendapatkan detail yang lebih rinci atau melakukan *roll up* untuk proses agregasi dari suatu *metrics* dalam suatu dimensi tunggal ataupun dalam *multiple* dimensi [7].
4. Menghasilkan suatu *view* yang dapat menyajikan arti dari segala arah, termasuk *chart*  dan *graph* [7].

# 2.2.3 *Fuzzy Data Warehouse*

*Fuzzy dimension* merupakan dimensi atau fakta yang memuat dan mengidentifikasi unsur-unsur *fuzzy* terhadap data yang digunakan untuk kepentingan analisis [1]. Data yang digunakan harus dipertimbangkan guna untuk memberikan suatu manfaat terhadap proses bisnis [1]. Informasi dasar yang dibutuhkan dalam mendefinisikan variabel *fuzzy*  adalah tabel sumber dan target atribut, asosiasi, jenis atribut, sifat-sifatnya, dan cara perhitungannya [1].

Dalam suatu sistem data *warehouse*, konsep *fuzzy* dapat diintegrasikan sebagai struktur *meta table* tanpa mempengaruhi inti dari suatu data *warehouse* [5]. Dengan pendekatan *fuzzy* *data warehousing* memungkinkan untuk mengekstraksi dan menganalisis data secara bersamaan dalam bentuk klasik yang jelas dan dengan cara *fuzzy* [5].

# 2.2.3.1 Konsep *Fuzzy Data Warehouse*

Pengintegrasian konsep *fuzzy* menjadi *data warehouse* merupakan penganalisisan elemen-elemen dalam data *warehouse* yang bisa diklasifikasikan secara *fuzzy* dimana elemen yang dianalisis bisa saja suatu fakta di dalam tabel fakta ataupun suatu atribut dari suatu dimensi [5].

Target atribut (TA) merupakan suatu atribut dimensi atau fakta di tabel fakta yang diperlukan untuk diklasifikasi secara *fuzzy* [5]. Atrribut domain merupakan suatu set keemungkinan nilai atau rentang kemungkinan nilai dari suatu atribut dimensi atau fakta yang bisa di petakan dalam suatu *lingustic term* yang akan dipetakan kedalam himpunan kelas dalam suatu konsep *fuzzy* [5] . Pemodelan *fuzzy* data *warehouse* merupakan suatu kombinasi dari empat elemen yaitu tabel dimensi, tabel fakta, *fuzzy membership*, dan *fuzzy classification* [5]. Ada tiga tipe dalam memodelkan *fuzzy data warehouse* yang dapat digunakan untuk merancang konsep *fuzzy* [5], yaitu:

1. *Distinct Fuzzy Classes / Linguistic Terms*

*Linguistic terms* tunggal yang digunakan tanpa adanya pengulangan yang dilakukan terhadap lainnya, yang artinya satu atribut target hanya diperuntukkan terhadap satu kelas dan derajat keanggotaan *fuzzy-*nya terhadap relasi yang diukur oleh fungsi keanggotannya [5].

1. *Different Membership Degrees for the Same Linguistic Terms*

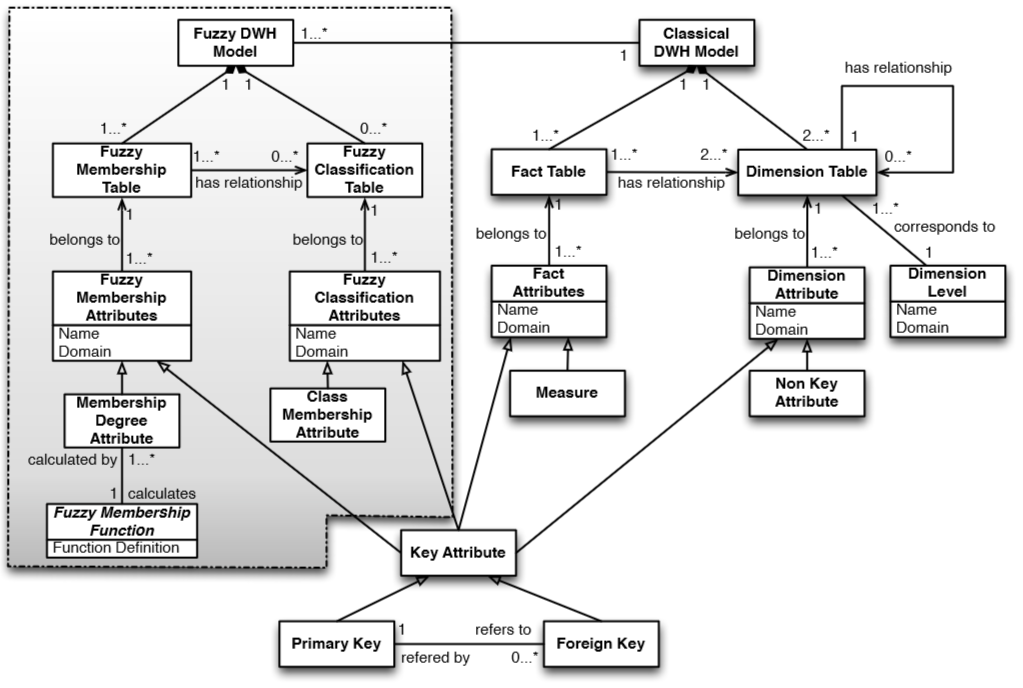
Suatu atribut target yang mungkin diperuntukkan pada satu *linguistic terms* tetapi juga memungkinkan untuk memiliki perbedaan kederajatan terhadap *linguistic terms*  yang ada [5].

1. *Different Linguistic Terms for a Target Attribut*

Suatu target atribut yang munking diperuntukkan pada *multiple linguistic terms*, sehingga dapat memiliki *multiple class*  dan memiliki *multiple* fungsi keanggotaannya [5].

# 2.2.3.2 Meta Model *Fuzzy Data Warehouse*

Suatu meta model mendefinisikan elemen-elemen dari konseptualisasi dan hubungan mereka [5]. Dalam *fuzzy data warehouse*, meta model dalam *data warehouse* saling terintegrasi dengan konsep *fuzzy* sebagai struktur meta tabel. Model meta *data warehouse* mengacu pada skema *data warehouse* yang terdiri dari satu atau lebih tabel fakta yang terletak di pusat untuk menyajikan informasi proses bisnis dan dua atau lebih tabel dimensi [5]. Model meta *fuzzy data warehouse*  mengacu pada konsep *fuzzy*  yang terintegrasi dengan suatu *data warehouse* dimana untuk setiap target atribut yang diidentifikasikan dalam model *fuzzy data warehouse* bisa lebih dari satu dan begitu juga sebaliknya [5].



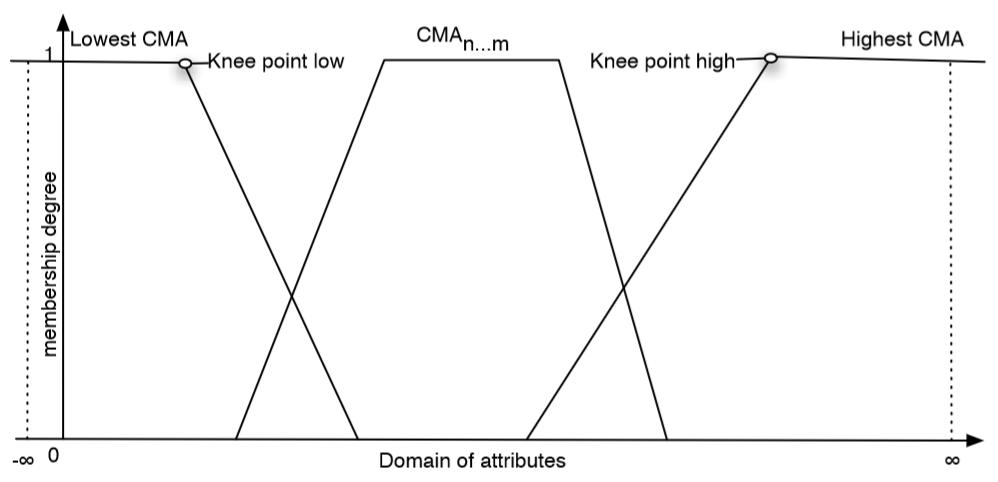
Gambar 6 Model meta *fuzzy data warehouse* [5]

# 2.2.3.3 Tipe Konsep *Fuzzy*

Konsep *fuzzy* harus benar-bnar bisa menangani rentang nilai yang dimiliki oleh atribut target [5]. Berdasarkan dengan bagaimana suatu rentang nilai harus digunakan, maka pendefinisian konsep *fuzzy* bisa jadi berbeda [5]. Konsep *fuzzy* dapat dibedakan menjadi tiga jenis, yaitu:

1. *Open End Fuzzy Concept*

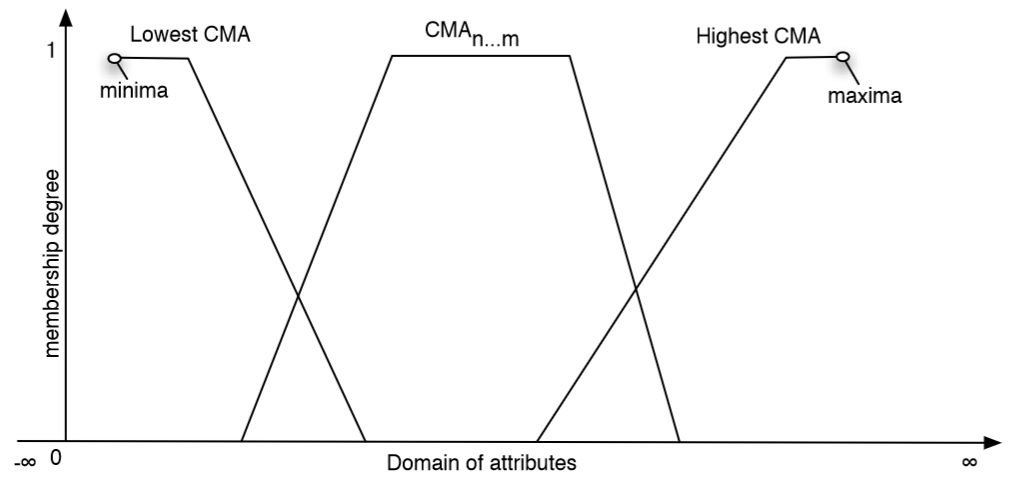
Konsep *fuzzy* yang memberikan keakuratan suatu klasifikasi antar titik yang mencakupi semua batasan dari nilai target atribut dan semua derajat keanggotaan [5]. Konsep ini membutuhkan banyak atribut derajat keanggotaan dalam tabel keanggotaan sebagai jumlah cartesian produk dari atribut kelas dan atribut target.



Gambar 7 *Open end fuzzy concept* [5]

1. *Limited Fuzzy Concept*

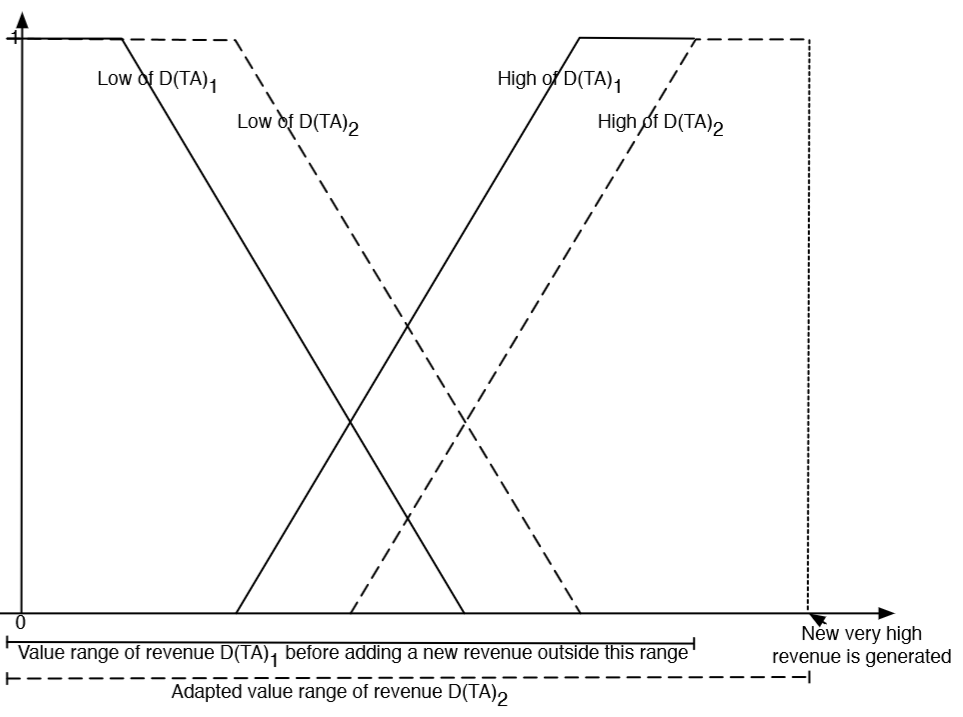
Konsep ini merupakan konsep yang terbatas dimana nilai target atribut yang didefinisikan di luar domain menjadi nilai yang tidak dianggap dan dibuang [5]. Nilai terendah dan nilai tertinggi pada nilai dari target atribut didefinisikan pada titik minima dan maksima [5].



Gambar 8 *Limited fuzzy concept* [5]

1. *Adaptive Fuzzy Concept*

Konsep ini merupkan konsep yang menggunakan atribut domain yang lengkap dimana konsep *fuzzy* tidak didefinsikan sebagai titik tetap atau sebagai titik minima dan maksima [5]. Atribut kelas didefinisikan secara relatif terhadap rentang nilai dan mencakup berbagai persentil yang dapat menyesuaikan diri dengan domain baru secara otomatis.



Gambar 9 *Adaptive fuzzy concept* [5]

# 2.2.4 *Object Oriented Analysis and Design (OOAD)*

OOAD merupakan metodologi dan teknik pengembangan sistem yang bebasis objek [8]. Pendekatan *OOAD* menggabungkan data dan proses menjadi entitas tunggal yang disebut objek [8]. Objek biasanya terkorespondensi terhadap suatu sistem informasi seperti kostumer, suppliers, atau kontrak [8]. Tujuan dari *OOAD*  adalah untuk membuat elemen sistem lebih *reusable*, *improving system quality* dan produktivitas terhadap analisis dan desain sistem [8]. Alat bantu yang digunakan untuk membantu pemodelan dalam *OOAD* adalah *UML* ( *Unified Modeling Language )*. Berikut ini adalah diagram yang digunakan dalam *UML.*

1. *Use Case*

*Use Case* merepukan *case* atau situasi yang digunakan oleh sistem untuk memenuhi satu atau lebih dari kebutuhan pengguna, *use case* menangkap bagian-bagian dari fungsional yang disediakan oleh sistem [9].

1. *Activity Diagram*

*Activity* *Diagram* memungkinkan dalam menentukan bagaimana sistem mencapai tujuan dalam tindakan-tindakan yang dilakukan dalam suatu proses yang telah di gambarkan pada *use case* [9].

1. *Class Diagram*

*Class Diagram*  merupakan model hubungan dari *class*  yang digunakan didalam sistem yang menggambarkan berbagai jenis objek yang ada pada sistem [9].

1. *Sequence Diagram*

*Sequence Diagram*  merupakan gambaran interaksi antar bagian-bagian yang ada di dalam sistem yang dipicu ketika suatu proses *use case* dijalankan dalam sistem [9].

# BAB III ANALISIS DAN PERANCANGAN DATA WAREHOUSE

# 3.1 Analisis Data *Warehouse*

Analisis data *warehouse* merupakan suatu analisis yang dilakukan untuk mendapatkan kejelasan mengenai proses yang digunakan dalam membangung sistem data *warehouse* yang baik dan benar. Langkah dalam analisis ini yaitu analisis masalah, analisis sumber data, analisis kebutuhan informasi strategis, analisis data *staging,* analisis *fuzzy dimensional*, dananalisis dimensi dan fakta.

# 3.1.1 Analisis Masalah

Masalah yang ada pada Perusahaan Spaceman sekarang adalah sebagai berikut:

1. Desain Data

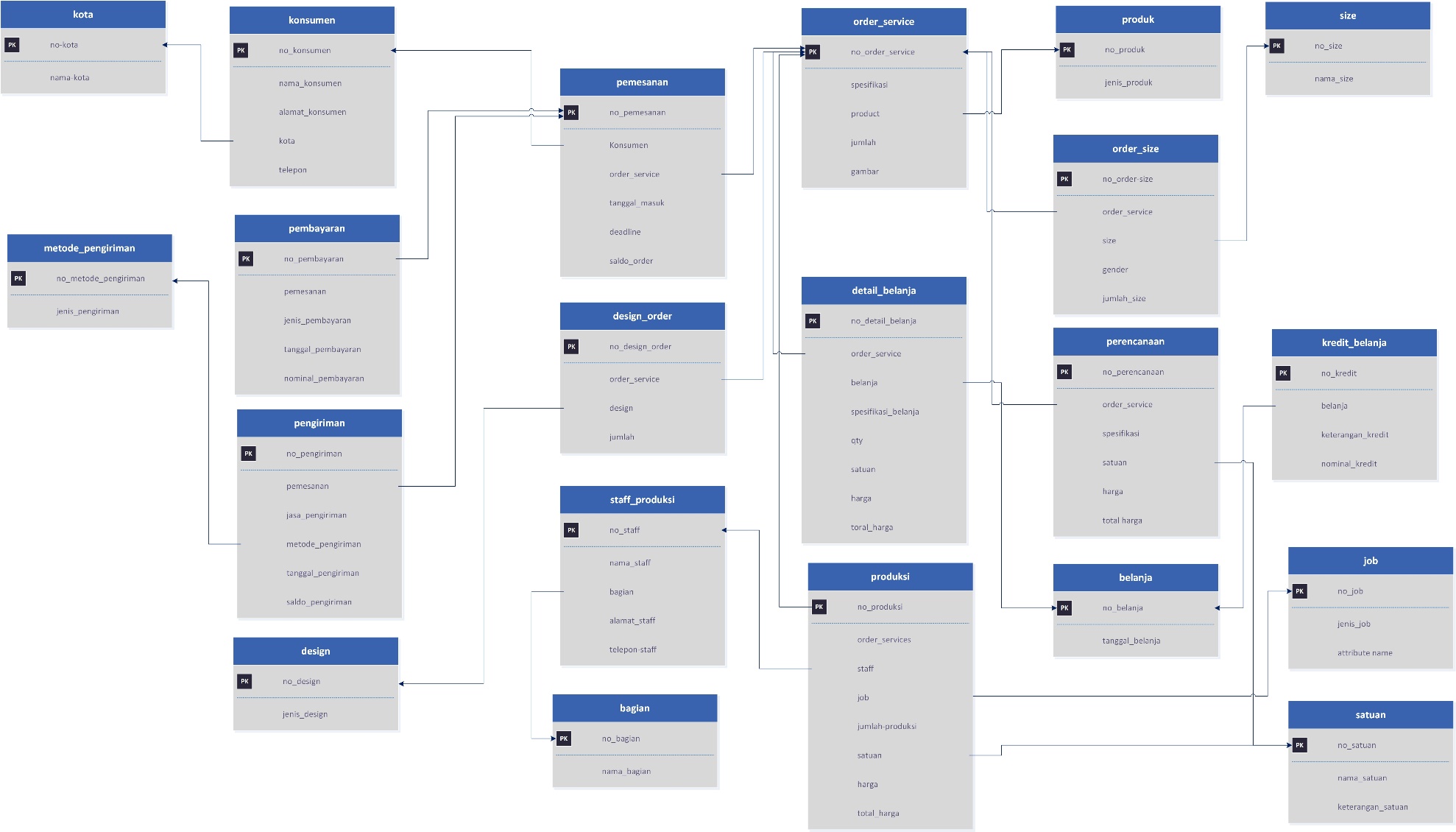
Desain data merupakan elemen penting dalam penunjang untuk mengambil keputusan. Desain data yang diterapkan di dalam perusahaan masih menggunakan konsep yang sederhana dan belum terintegrasi, sehingga sulit untuk melakukan suatu analisa pada informasi yang memerlukan desain yang dapat menampilkan banyak data dari data sekarang maupun data *historis*.

1. Pembuatan Laporan

Pembuatan laporan yang dilakukan dalam perusahaan dengan cara merekap data yang terlibat dalam transaksi suatu pemesanan konsumen yang kemudian dibuat laporan dengan bentuk laporan masing-masing. Masalah dalam pembuatan laporan ini adalah dengan merekap data yang terlibat sehingga harus melakukan pengecekan kembali dan hasil laporan yang tidak mendukung untuk dapat dilihat dari berbagai aspek yang terlibat dalam laporan tersebut.

# 3.1.2 Analisis Sumber Data

Analisis sumber data pada perusahan Spaceman berdasarkan pada database yang sudah ada. Berikut skema OLTP yang dibutuhkan:



Gambar 10 Skema diagram *OLTP*

1. Tabel kota

Tabel kota merupakan tabel untuk menyimpan data kota konsumen yang melakukan pemesanan produk.

Tabel 1 Struktur data tabel kota

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_kota | Integer |  | Primary Key | Auto\_increment |
| 2 | Nama\_kota | Varchar | 20 |  |  |

1. Tabel konsumen

Tabel konsumen merupakan tabel untuk menyimpan data profil konsumen yang melakukan pemesanan produk.

Tabel 2 Struktur data tabel konsumen

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_konsumen | Integer |  | Primary Key | Auto\_increment |
| 2 | Nama\_konsumen | Varchar | 25 |  |  |
| 3 | Alamat | Varchar | Max |  |  |
| 4 | Kota | Int |  | Foreign Key |  |
| 5 | Telepon | Varchar | 13 |  | Hanya bisa menginput numerical |

1. Tabel order\_service

Tabel order\_service merupakan tabel yang menyimpan data produk yang akan diorder oleh konsumen.

Tabel 3 Struktur data tabel order\_service

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_order\_service | Varchar | 21 | Primary Key | Kode order bebentuk  Space/ord/bulan/urutan order/tahun |
| 2 | Spesifikasi | Varchar | 100 |  |  |
| 3 | Produk | Integer |  | Foreign key |  |
| 4 | Jumlah | Integer |  |  |  |
| 5 | Gambar\_order | Varchar | 100 |  |  |

1. Tabel pemesanan

Tabel pemesanan merupakan tabel yang menyimpan data pemesanan konsumen yang memesan produk yang ada pada di tabel order\_service.

Tabel 4 Struktur data tabel pemesanan

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_pemesanan | Integer |  | Primary Key | Auto\_increment |
| 2 | Service\_order | Varchar | 21 | Foreign key |  |
| 3 | Konsumen | Int |  | Foreign key |  |
| 4 | Tanggal\_masuk | Date |  |  |  |
| 5 | Deadline | Date |  |  |  |
| 6 | Jumlah | Integer |  |  |  |
| 7 | Saldo\_order | Float |  |  |  |

1. Tabel perencanaan

Tabel perencanaan merupakan tabel yang menyimpan data estimasi pemesanan konsumen terhadap produk\_service yang dipesan.

Tabel 5 Struktur data tabel perencanaan

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_estimasi | Integer |  | Primary key | Auto\_increment |
| 2 | Service\_order | Varchar | 21 | Foreign key |  |
| 3 | Spesifikasi | Varchar | 100 |  |  |
| 4 | Jumlah | Float |  |  |  |
| 5 | Satuan | Integer |  | Foriegn key |  |
| 6 | Harga | Float |  |  |  |
| 7 | Total\_harga | Float |  |  |  |

1. Tabel pengiriman

Tabel pengiriman merupakan tabel yang menyimpan data transasksi pengiriman produk ke konsumen.

Tabel 6 Struktur data tabel pengiriman

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_pengiriman | Integer |  | Primary Key | Auto\_increment |
| 2 | Pemesanan | Integer |  | Foreign key |  |
| 3 | Jasa\_pengiriman | Varchar | 100 |  |  |
| 4 | Metode\_pengiriman | Int |  | Foreign key |  |
| 5 | Nominal\_pengiriman | Float |  |  |  |

1. Tabel metode\_pengiriman

Tabel metode\_pengiriman merupakan tabel yang digunakan untuk menyimpan metode pengiriman produk ke konsumen.

Tabel 7 Struktur data tabel metode\_pengiriman

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_metode\_pengiriman | Integer |  | Primary Key | Auto\_increment |
| 2 | Nama\_metode\_pengiriman | Varchar | 100 |  |  |

1. Tabel pembayaran

Tabel pembayaran merupakan tabel yang digunakan untuk menyimpan data transaksi pembayaran pemesanan terhadap konsumen.

Tabel 8 Struktur data tabel pembayaran

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_pembayaran | Integer |  | Primary Key | Auto\_increment |
| 2 | Pemesanan | Int |  | Foreign key |  |
| 3 | Jenis\_pembayaran | Varchar | 100 |  |  |
| 4 | Nominal\_pembayaran | Float |  |  |  |
| 5 | Keterangan\_pembayaran | Enum | (“DP”,”Lunas”) |  |  |

1. Tabel order\_size

Tabel order\_size merupakan tabel yang digunakan untuk menyimpan data ukuran produk yang dipesan oleh konsumen.

Tabel 9 Struktur data tabel order\_size

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_order\_size | Integer |  | Primary Key | Auto\_increment |
| 2 | Service\_order | Varchar | 21 | Foreign key |  |
| 3 | Size | Integer |  | Foreign key |  |
| 4 | Gender | Enum | “male”,”female” |  |  |
| 5 | Jumlah | Integer |  |  |  |

1. Tabel size

Tabel size merupakan tabel yang digunakan untuk menyimpan data jenis ukuran yang akan digunakan pada produk.

Tabel 10 Struktur data tabel size

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_size | Integer |  | Primary Key | Auto\_increment |
| 2 | Nama\_size | Varchar | 7 |  |  |

1. Tabel design\_order

Tabel design\_order merupakan tabel yang digunakan untuk menyimpan data desain yang digunakan pada suatu produk yang dipesan oleh konsumen.

Tabel 11 Struktur data tabel design\_order

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_design\_order | Integer |  | Primary Key | Auto\_increment |
| 2 | Service\_order | Varchar | 21 | Foreign key |  |
| 3 | Design | Integer |  | Foreign key |  |
| 4 | Jumlah | Integer |  |  |  |

1. Tabel design

Tabel design merupakan tabel yang digunakan untuk menyimpan data jenis desain yang akan digunakan pada produk.

Tabel 12 Struktur data tabel design

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_design | Integer |  | Primary Key | Auto\_increment |
| 2 | Jenis\_design | Varchar | 200 |  |  |

1. Tabel produksi

Tabel produksi merupakan tabel yang digunakan untuk menyimpan data proses produksi yang dlakukan untuk pengadaan suatu produk.

Tabel 13 Struktur data tabel produksi

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_produksi | Integer |  | Primary Key | Auto\_increment |
| 2 | Service\_order | Varchar | 21 | Foreign key |  |
| 3 | Staff | Integer |  | Foreign key |  |
| 4 | Job | Integer |  | Foreign key |  |
| 5 | Jumlah | Float |  |  |  |
| 6 | Satuan | Integer |  | Foreign key |  |
| 7 | Harga | Float |  |  |  |
| 8 | Total\_harga | Float |  |  |  |

1. Tabel job

Tabel job merupakan tabel yang digunakan untuk menyimpan jenis pekerjaan yang dilakukan terhadap proses produksi untuk pengadaan produk.

Tabel 14 Struktur data tabel job

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_job | Integer |  | Primary Key | Auto\_increment |
| 2 | Jenis\_job | Varchar | 100 |  |  |

1. Tabel\_staff\_produksi

Tabel staff\_produksi merupakan tabel yang digunakan untuk menyimpan data profil pegawai yang bekerja di bagian produksi.

Tabel 15 Struktur data staff\_produksi

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_staff\_produksi | Integer |  | Primary Key | Auto\_increment |
| 2 | Nama staff | Varchar | 30 |  |  |
| 3 | Bagian | Integer |  | Foreign key |  |
| 4 | Alamat | Varchar | Max |  |  |
| 5 | Telepon | Varchar | 13 |  | Penginputan hanya boleh numerikal |

1. Tabel bagian

Tabel bagian merupakan tabel yang digunakan untuk menyimpan data bagian suatu pegawai di bagian produksi.

Tabel 16 Struktur data tabel bagian

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_bagian | Integer |  | Primary Key | Auto\_increment |
| 2 | Nama\_bagian | Varchar | 20 |  |  |

1. Tabel satuan

Tabel satuan merupakan tabel yang digunakan untuk menyimpan data satuan yang digunakan terhadap alat atau bahan yang menggunakan satuan.

Tabel 17 Struktur data tabel satuan

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_satuan | Integer |  | Primary Key | Auto\_increment |
| 2 | Nama satuan | Varchar | 5 |  |  |
| 3 | Keterangan | Varchar | 50 |  |  |

1. Tabel detail\_belanja

Tabel detail\_belanja merupaan tabel yang digunakan untuk menyimpan data pembelanjaan yang dibutuhkan untuk mengadakan suatu produk.

Tabel 18 Struktur data tabel detail\_belanja

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_detail\_belanja | Integer |  | Primary Key | Auto\_increment |
| 2 | Service\_order | Varchar | 21 | Foreign key |  |
| 3 | Belanja | Integer |  | Foreign key |  |
| 4 | Qty | Float |  |  |  |
| 5 | Satuan | Integer |  | Foreign key |  |
| 6 | Harga | Float |  |  |  |
| 7 | Total\_harga | Float |  |  |  |

1. Tabel belanja

Tabel belanja merupakan tabel yang digunakan untuk menyimpan data aktivitas belanja.

Tabel 19 Struktur data tabel belanja

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_belanja | Integer |  | Primary Key | Auto\_increment |
| 2 | Tanggal\_belanja | Date |  |  |  |

1. Tabel kredit\_belanja

Tabel kredit\_belanja merupakan tabel yang digunakan untuk menyimpan data kas perusahaan.

Tabel 20 Struktur data tabel kredit\_belanja

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_kredit | Integer |  | Primary Key | Auto\_increment |
| 2 | Belanja | Integer |  | Foreign key |  |
| 3 | Keterangan | Varchar | 200 |  |  |
| 4 | Nominal | Float |  |  |  |

1. Tabel produk

Tabel produk merupakan tabel yang digunakan untuk menyimpan jenis produk yang dapat dipesan oleh konsumen.

Tabel 21 Struktur data tabel produk

| **No.** | **Nama Field** | **Type** | **Value** | **Status** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | No\_produk | Integer |  | *Primary Key* | *Auto increment* |
| 2 | Nama\_produk | Varchar | 20 |  |  |

# 3.1.3 Analisis Kebutuhan Informasi Strategis

Berdasarkan hasil diskusi dengan owner perusahaan, maka dapat diketahui bahwa informasi strategis yang dibutuhkan adalah sebagai berikut:

* 1. Informasi kepopuleran produk yang dipesan oleh konsumen per bulan dalam periode per tahun.
  2. Informasi pada pemesanan produk yang dilakukan oleh konsumen dalam periode per tahun.
  3. Informasi persentase dalam pelunasan pembayaran terhadap pesanan yang dilakukan oleh konsumen per golongan dalam periode pertahun.
  4. Informasi biaya pengiriman produk yang digunakan untuk mengirim produk ke konsumen per bulan dalam periode per tahun.
  5. Informasi pembelanjaan produksi terhadap pemesanan konsumen per produk dalam periode per tahun.
  6. Informasi biaya produksi pada pesanan konsumen per produk dalam periode per tahun.
  7. Informasi kepopuleran desain yang digunakan pada produk per kelompok dalam periode per tahun.
  8. Informasi *size* yang telah dipesan per produk setiap bulan dalam periode per tahun.

# 3.1.4 Analisis Dimensi dan Fakta Bisnis

Berdasarkan pada *point* **3.1.3** yang telah membahas kebutuhan informasi strategis, maka dapat dibuat model dimensi yang dibutuhkan dalam data *warehouse*. **Tabel 22** menjelaskan dimensi bisnis yang dibutuhkan berdasarkan kebutuhan informasi strategisnya.

Tabel 22 Dimensi bisnis Perusahaan Spaaceman

|  |  |  |
| --- | --- | --- |
| **1** | **Kebutuhan Informasi Strategis** | Informasi kepopuleran produk yang dipesan oleh konsumen per bulan dalam perioder per tahun. |
| **Tabel Fakta** | Fact\_pop\_produk |
| **Tabel Dimensi** | 1. Dim\_waktu 2. Dim\_produk |
| **2** | **Kebutuhan Informasi Strategis** | Informasi pemesanan produk yang dilakukan oleh konsumen dalam periode per tahun. |
| **Tabel Fakta** | Fact\_pemesanan |
| **Tabel Dimensi** | 1. Dim\_waktu 2. Dim\_produk 3. Dim\_konsumen 4. Dim\_size |
| **3** | **Kebutuhan Informasi Strategis** | Informasi persentase dalam pelunasan pembayaran terhadap pesanan yang dilakukan oleh konsumen per golongan dalam periode pertahun |
| **Tabel Fakta** | Fact\_persentase\_pembayaran |
| **Tabel Dimensi** | 1. Dim\_waktu 2. Dim\_konsumen |
| **4** | **Kebutuhan Informasi Strategis** | Informasi biaya pengiriman produk yang digunakan untuk mengirim produk ke konsumen per bulan dalam periode per tahun. |
| **Tabel Fakta** | Fact\_pengiriman |
| **Tabel Dimensi** | 1. Dim\_waktu 2. Dim\_produk 3. Dim\_jasa\_kirim |
| **5** | **Kebutuhan Informasi Strategis** | Informasi belanja produksi per produk dalam periode per tahun |
| **Tabel Fakta** | Fact\_belanja |
| **Tabel Dimensi** | 1. Dim\_waktu 2. Dim\_produk |
| **6** | **Kebutuhan Informasi Strategis** | Informasi biaya produksi pada produk pesanan konsumen per staff dalam periode per tahun |
| **Tabel Fakta** | fact\_produksi |
| **Tabel Dimensi** | 1. Dim\_waktu 2. Dim\_staff\_produksi 3. Dim\_produk 4. Dim\_job |
| **7** | **Kebutuhan Informasi Strategis** | Informasi kepopuleran desain yang digunakan pada produk dalam periode per tahun. |
| **Tabel Fakta** | Fact\_pop\_design |
| **Tabel Dimensi** | 1. Dim\_waktu 2. Dim\_design 3. Dim\_produk |

Berdasarkan penjelasan yang ada pada **Tabel 22**, dapat diketahui tabel dimensi dan tabel fakta yang akan digunakan dalam sistem ini. Berikut penjelasan tabel dimensi dan tabel fakta yang digunakan.

* + - 1. **Dimensi**

Dimensi digunakan untuk menjelaskan entitas-entitas bisnis yang akan mendukung suatu fakta sehingga dapat di-analisis secara multidimensional. Berikut penjelasan dimensi yang digunakan dalam sistem:

1. Dim\_design

Dim\_design merupakan dimensi yang menjelaskan jenis design yang diterapkan pada produk. **Tabel 23** menjelaskan struktur data pada dim\_design.

Tabel 23 Struktur data dim\_design

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_design | Integer |  | *Primary Key* |
| Nama\_design | Varchar | 100 | *Not null* |

Dim\_design memiliki hirarki yang dijelaskan pada **tabel 24**.

Tabel 24 Hirarki design

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Nama\_design |
| 2 | No\_design |

1. Dim\_job

Dim\_job merupakan dimensi yang menjelaskan pekerjaan pada tahap produksi yang dilakukan oleh staff produksi. **Tabel 25** menjelaskan struktur data pada dim\_job.

Tabel 25 Sturktur data dim\_job

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_job | Integer |  | *Primary Key* |
| keterangan\_job | Varchar | 50 | *Not null* |

Dim\_job memiliki hiararki yang dijelaskan pada **Tabel 26.**

Tabel 26 Hirarki job

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Keterangan\_job |
| 2 | No\_job |

1. Dim\_konsumen

Dim\_konsumen merupakan dimensi yang menjelaskan profil konsumen yang telah memesan produk. **Tabel 27** menjelaskan struktur data pada dim\_konusmen.

Tabel 27 Struktur data dim\_konsumen

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_konsumen | Integer |  | *Primary Key* |
| Nama\_konsumen | Varchar | 50 | *Not null* |
| Nama\_kota | Varchar | 50 | *Not null* |

Dim\_konsumen memiliki hirarki yang dijelaskan pada **Tabel 28** dan **Tabel 29**.

Tabel 28 Hirarki kota

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Nama\_kota |
| 2 | Nama\_konsumen |

Tabel 29 Hirarki konsumen

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Nama\_konsumen |
| 2 | No\_konsumen |

1. Dim\_kredit

Dim\_kredit merupakan dimensi yang menjelaskan jenis pengeluaran terhadap pembelanjaan produksi. **Tabel 30** menjelaskan struktur data pada dim\_kredit.

Tabel 30 Struktur data dim\_kredit

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_kredit | Integer |  | *Primary Key* |
| Jenis\_kredit | Varchar | 50 | *Not null* |

Dim\_kredit memiliki hirarki yang dijelaskan pada **Tabel 31**.

Tabel 31 Hirarki kredit

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Jenis\_kredit |
| 2 | No\_kredit |

1. Dim\_pengiriman

Dim\_pengiriman merupakan dimensi yang menjelaskan jenis pengiriman yang digunakan dalam pengiriman produk ke konsumen. **Tabel 32** menjelaskan struktur data pada dim\_pengiriman.

Tabel 32 Struktur data dim\_pengiriman

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_pengiriman | Integer |  | *Primary Key* |
| Jenis\_pengiriman | Varchar | 100 | *Not null* |

Dim\_pengiriman memiliki hirarki yang dijelaskan pada **Tabel 33**.

Tabel 33 Hirarki pengiriman

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Jenis\_pengiriman |
| 2 | No\_pengiriman |

1. Dim\_produk

Dim\_produk merupakan dimensi yang menjelaskan produk yang dapat dipesan konsumen. **Tabel 34** menjelaskan struktur data pada dim\_produk.

Tabel 34 Sturktur data dim\_produk

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_produk | Integer |  | *Primary Key* |
| Nama\_produk | Varchar | 50 | *Not null* |

Dim\_produk memiliki hirarki yang dijelaskan pada **Tabel 35**.

Tabel 35 Hirarki produk

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Nama\_produk |
| 2 | No\_produk |

1. Dim\_size

Dim\_size merupakan dimensi yang menjelaskan ukuran yang digunakan pada produk yang dipesan oleh konsumen berdasarkan jenis kelaminnya. **Tabel 36** menjelaskan struktur data pada dim\_size.

Tabel 36 Struktur data dim\_size

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_size | Integer |  | *Primary Key* |
| Nama\_size | Varchar | 10 | *Not null* |

Dim\_size memiliki hirarki yang dijelaskan pada **Tabel 37**.

Tabel 37 Hirarki size

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Nama\_size |
| 2 | No\_size |

1. Dim\_staff\_produksi

Dim\_staff\_produksi merupakan dimensi yang menjelaskan profil staff\_produksi yang mengerjkalan pembuatan produk. **Tabel 38** menjelaskan struktur data pada dim\_staff\_produksi.

Tabel 38 Struktur data dim\_staff\_produksi

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| No\_staff | Integer |  | *Primary Key* |
| Nama\_staff | Varchar | 50 | *Not null* |
| Nama\_bagian | Varchar | 50 | *Not\_null* |

Dim\_staff\_produksi memiliki hirarki yang dijelaskan pada **Tabel 39** dan **Tabel 40**.

Tabel 39 Hirarki bagian

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Nama\_bagian |
| 2 | Nama\_staff |

Tabel 40 Hirarki Staff

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Nama\_staff |
| 2 | No\_staff |

1. Dim\_waktu

Dim\_waktu merupakan dimensi yang menjelaskan waktu, baik itu dalam segi waktu transaksi, belanja, maupun pengiriman. **Tabel 41** menjelaskan struktur data pada dim\_waktu.

Tabel 41 Struktur data dim\_waktu

| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| --- | --- | --- | --- |
| No\_waktu | Integer |  | *Primary Key* |
| Fulldate | Nvarchar | 50 | *Not null* |
| Hari | Nvarchar | 50 | *Not null* |
| Tanggal | Integer |  | *Not null* |
| Nama\_bulan | Nvarchar | 10 | *Not null* |
| Bulan | Integer |  | *Not null* |
| Tahun | Integer |  | *Not null* |

Dim\_waktu memiliki hirarki yang dijelaskan pada **Tabel 42**.

Tabel 42 Hirarki waktu

|  |  |
| --- | --- |
| **Level** | **Atribut** |
| 1 | Tahun |
| 2 | Bulan |
| 3 | Tanggal |

* + - 1. **Fakta**

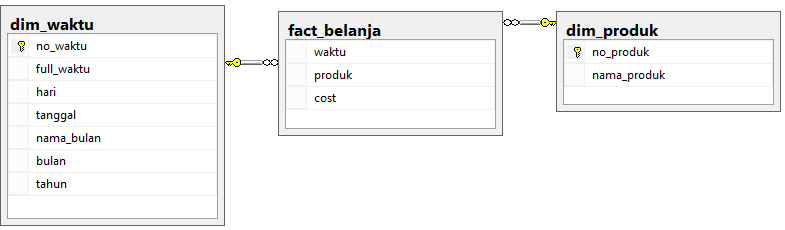
Fakta menjelaskan langsung nilai-nilai yang berkaitan dengan proses bisnsi yang bersifat multidimensional agar lebih mudah penggunan untuk menganalisisnya. Berikut penjelasan dan skema fakta yang digunakan dalam sistem ini:

1. Fact\_belanja

Fact\_belanja menjelaskan fakta pengeluaran dalam biaya pemebelian bahan dan alat untuk kepentingan pembuatan produk. **Tabel 43** menjelaskan struktur data pada fact\_belanja dan **Gambar 11** menjelaskan skema fact\_belanja terhadap dimensi yang terkait.

Tabel 43 Struktur data fact\_belanja

| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| --- | --- | --- | --- |
| Waktu | Integer |  | *Foreign key* |
| Produk | Integer |  | *Foreign key* |
| Cost | Float |  | *Not null* |



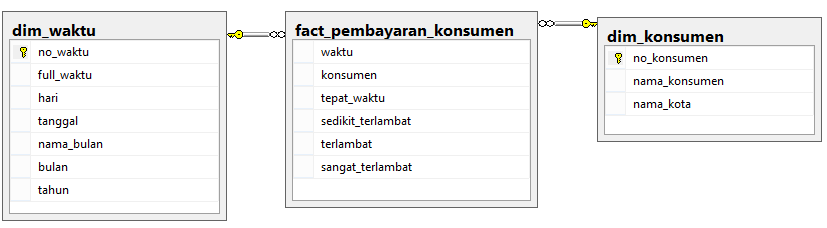
Gambar 11 Skema dimensional fakta belanja produksi

1. Fact\_pembayaran\_konsumen

Fact\_pembayaran\_konsumen menjelaskan pengelompokan rentang pembayaran konsumen terhadap tenggwat waktu yang ditetapkan dalam skala ke-abuan. **Tabel 44** menjelaskan sturktur data pada fakta pembayaran konsumen dan **Gambar 12** menjelaskan skema fakta terhadap dimensi yang terkait.

Tabel 44 Struktur data fact\_pembayaran\_konsumen

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| Waktu | Integer |  | *Foreign key* |
| Konsumen | Integer |  | *Foreign key* |
| Rentang\_pembayaran | Integer |  |  |



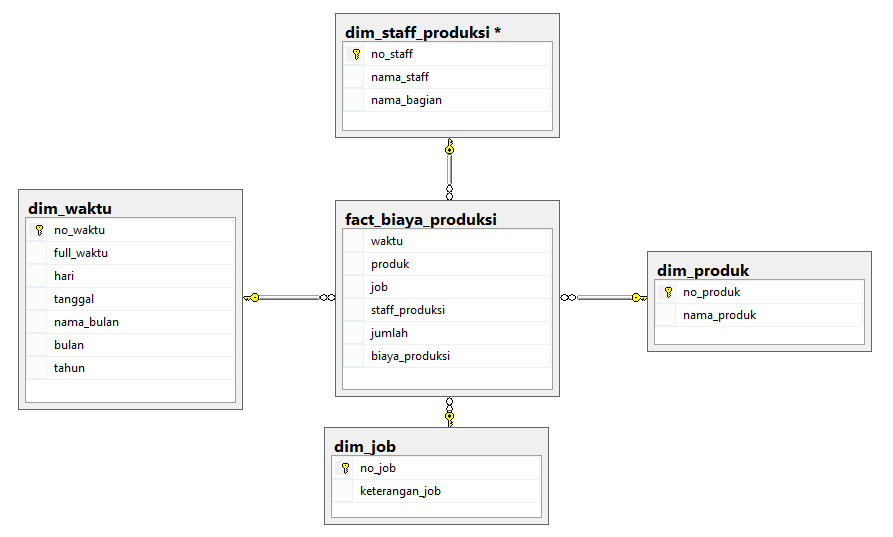
Gambar 12 Skema dimensional fakta pembayaran konsumen

1. Fact\_biaya\_produksi

Fact\_biaya\_produksi mejelaskan fakta biaya dalam proses pembuatan produk. **Tabel 45** menjelaskan struktur data pada fact\_biaya\_produksi dan **Gambar 13** menjelaskan skema fact\_biaya\_produksi terhadap dimensi yang terkait.

Tabel 45 Struktur data fact\_biaya\_produksi

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| Waktu | Integer |  | *Foreign key* |
| Produk | Integer |  | *Foreign key* |
| Job | Integer |  | *Foreign key* |
| Staff\_produksi | Integer |  | *Foreign\_key* |
| Jumlah | Integer |  | *Not null* |
| Biaya\_produksi | Float |  | *Not null* |



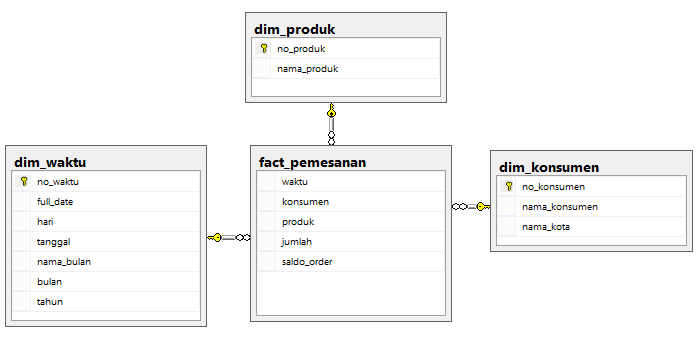
Gambar 13 Skema dimensional fakta biaya produksi

1. Fact\_pemesanan

Fact\_pemesanan menjelaskan fakta pemesanan produk yang dilakukan oleh konsumen. **Tabel 46**  menjelaskan struktur data pada fact\_pemesanan dan **Gambar 14** menjelaskan skema fakta terhadap dimensi yang terkait.

Tabel 46 Sturktur data fact\_pemesaan

| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| --- | --- | --- | --- |
| Waktu | Integer |  | *Foreign key* |
| Konsumen | Integer |  | *Foreign key* |
| Produk | Integer |  | *Foreign key* |
| Jumlah | Integer |  | *Not null* |
| Saldo\_order | Float |  | *Not null* |



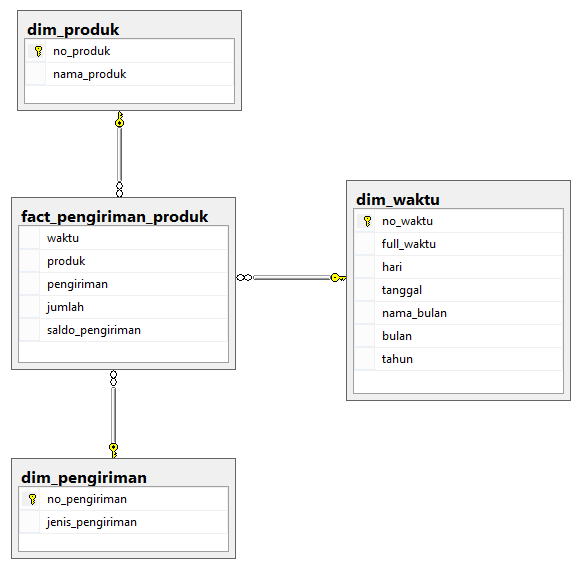
Gambar 14 Skema dimensional fakta pemesanan produk

1. Fact\_pengiriman\_produk

Fact\_pengiriman\_produk menjelaskan fakta biaya pengiriman suatu produk ke konsumen berdasarkan jenis pengiriman yang diterapkan. **Tabel 47** menjelaskan struktur data pada fact\_pengiriman\_produk dan **Gambar 15** menjelaskan skema fakta terhada dimensi yang tekait.

Tabel 47 Sturktur data fact\_pengiriman\_produk

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| Waktu | Integer |  | *Foreign key* |
| Produk | Integer |  | *Foreign key* |
| Pengiriman | Integer |  | *Foreign key* |
| Jumlah | Float |  | *Not null* |
| Saldo | Float |  | *Not null* |



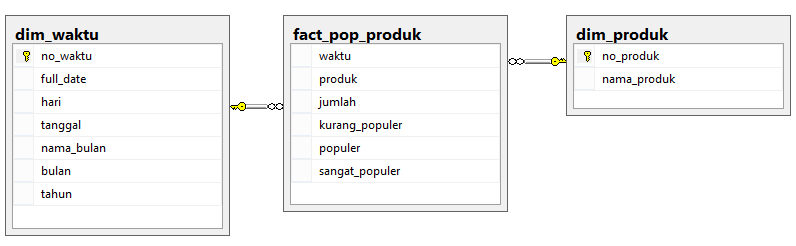
Gambar 15 skema dimensional fakta pengirman produk

1. Fact\_pop\_produk

Fact\_pop\_produk menjelaskan fakta kepopuler produk yang dipesan oleh konsumen. **Tabel 49** menjelaskan struktur data pada fact\_pop\_produk dan **Gambar 17** menjelaskan skema fakta terhadap dimensi yang terkait.

Tabel 49 Struktur data fact\_pop\_produk

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| Waktu | Integer |  | *Foreign key* |
| Produk | Integer |  | *Foreign key* |
| Jumlah | Integer |  | *Not null* |



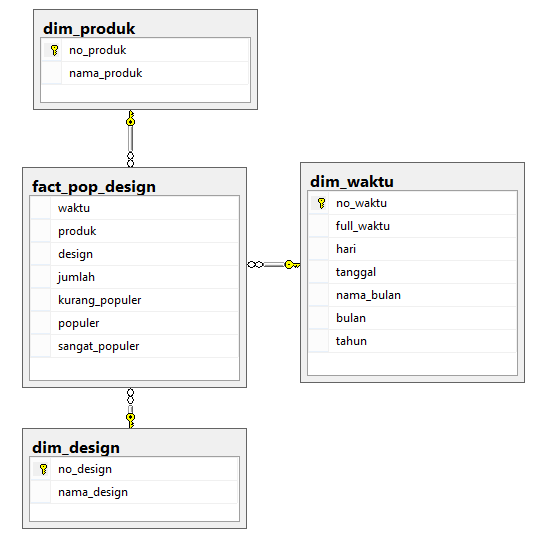
Gambar 16 Skema fakta kepopuleran produk

1. Fact\_pop\_design

Fact\_pop\_design menjelaskan kepopuleran suatu desain yang digunakan pada produk yang dipesan oleh konsumen yang digolong dalam derajat keanggotaannya. **Tabel 48** menjelaskan struktur data pada fact\_pop\_design dan **Gambar 16** menjelaskan skema fakta terhadap dimensi yang terkait.

Tabel 48 Struktur data pada fact\_pop\_design

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| Waktu | Integer |  | *Foreign key* |
| Produk | Integer |  | *Foreign key* |
| Design | Integer |  | *Foreign key* |
| Jumlah | Integer |  | *Not null* |



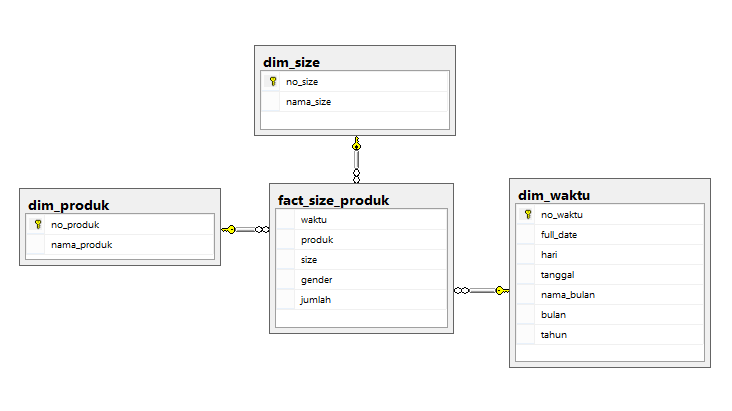
Gambar 17 Skema fakta kepopuler *design* produk

1. Fact\_size\_produk

Fact\_size\_produk menjelaskan fakta ukuran yang digunakan produk pada pemesanan konsumen. **Tabel 50** menjelaskan struktur data pada fact\_size\_produk dan **Gambar 18** menjelaskan skema fakta pada terhadap dimensi yang terkait.

Tabel 50 Struktur data fact\_size\_produk

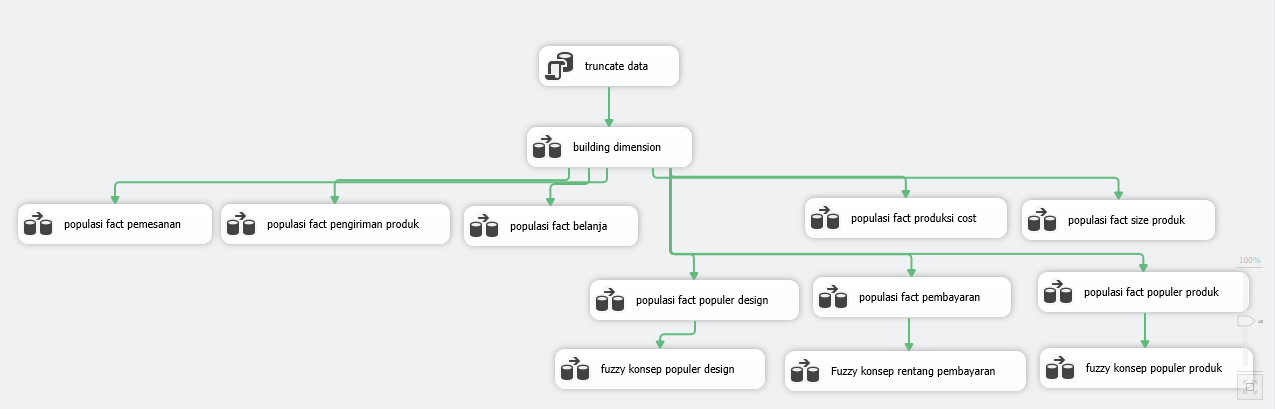
| **Nama Field** | **Tipe Data** | **Panjang** | **Keterangan** |
| --- | --- | --- | --- |
| Waktu | Integer |  | *Foreign key* |
| Produk | Integer |  | *Foreign key* |
| Size | Integer |  | *Foreign key* |
| Gender | Varchar(10) |  | *Not null* |
| Jumlah | Integer |  | *Not null* |



Gambar 18 Skema fakta *size* produk

# 3.1.5 Analisis Data *Staging*

Dalam tahapan ini, data operasional (*OLTP*) akan dilakaukan proses *data staging*  atau bisa juga disebut dengan proses ekstraksi, transformasi, dan *loading* (*ETL*) ke data *warehouse*. **Gambar 19** merupakan proses ETL untuk setiap pembuatan tabel dimensi dan populasi tabel fakta dengan menggunakan *Microsoft Visual Studio 2013 : Business Intelligence Project* dengan menggunakan *template* *Business Intelligence Integration Services.*



Gambar 19 *Control flow data staging*

Pada **Gambar 19** merupakan *control flow* dari proses *data* staging yang dilakukan untuk aliran data dari sumber ke tujuan, dan proses konsep *fuzzy* terhadap atribut target. Pada **Gambar 20** merupakan *toolbox* yang digunakan dalam pembuatan *control flow*.



Gambar 20 *Tools*  pembuatan *control flow* ETL

*Data flow task tools* merupakan alat untuk mendeskripsikan langkah aliran dari data. Sedangkan *execute SQL task* merupakan serangkaian perintah *SQL* yang dieksekusi. Berikut penjelasan dari setiap *task* yang dilakukan pada proses *data staging*.

1. *Truncate* *data*

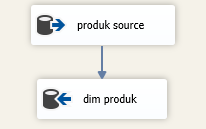
*Task* ini merupakan *execute SQL task* dimana dibangun perintah untuk membersihkan data dengan memperhatikan relasi antar tabel yang saling berhubungan di dalam *database data warehouse*.

1. *Building Dimension*

*Task* ini merupakan *data flow task* yang digunakan untuk mengalirkan data dari data sumber menuju ke tabel dimensi di *data warehouse.* Berikut aliran data masing-masing tabel dimensi dalam *data warehouse*.

1. Dimensi Produk

Dimensi produk dibentuk berdasarkan dari tabel produk yang ada di sumber. **Gambar 21** menggambarkan aliran data yang terjadi dalam dimensi Produk.

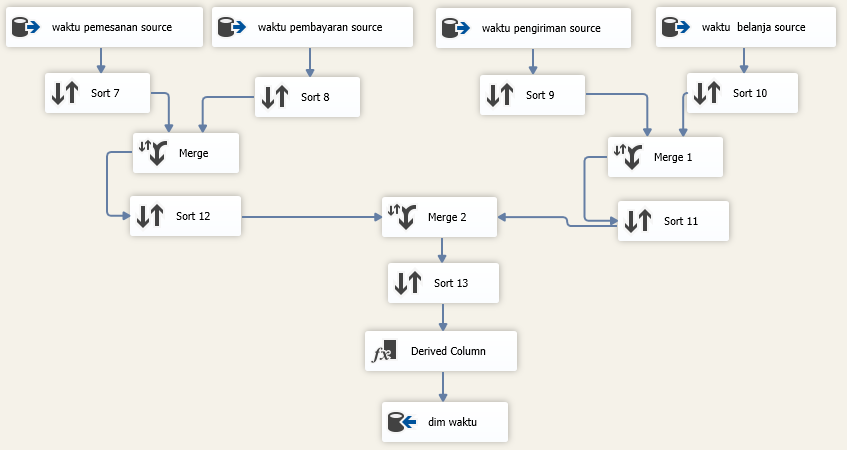


Gambar 21 *Data flow* dimensi produk

Adapun *step by step*  dari *data flow* dimensi produk ini adalah sebagai berikut:

1. Sumber data diambil dari tabel produk.
2. Data di-*extract* dengan memilih atribut yang akan digunakan didalam dimensi produk.
3. Hasil *extract* di petakan ke *destination* yaitu dimensi produk.
4. Dimensi Waktu

Dimensi waktu dibentuk sedemikian rupa berdasarkan dari tabel sumber yang memiliki atribut waktu. Masing-masing data di ekstraksi, transformasi, dan di muat ke dimensi waktu. **Gambar 22** menggambarkan aliran data yang terjadi pada dimensi waktu.

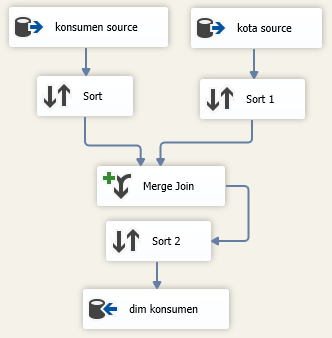


Gambar 22 *Data flow* dimensi waktu

Adapun *step by step* dalam aliran data menuju dimensi waktu adalah sebagai berikut.

1. Mengekstraksi data dari sumber yaitu, tabel pemesanan, tabel pembayaran, tabel pengiriman, dana tabel belanja.
2. Masing-masing data dari tabel digabung menjadi satu.
3. Mengecek data dan mensorting data, di tahapan ini data yang bersifat redudansi akan dihapus dan akan di ambil satu sebagai perwakilan data.
4. Melakukan transformasi data dengan pemecahan, *fulldate*, tanggal, bulan, dan tahun.
5. Hasil transformasi di muat ke dalam dimensi waktu.
6. Dimensi Konsumen

Dimensi konsumen dibentuk dari tabel konsumen dan kota sumber. **Gambar 23** menggambarkan aliran data yang terjadi pada dimensi konsumen.

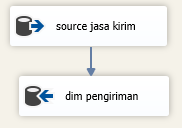


Gambar 23 *Data flow* dimensi konsumen

Adapun *step by step*  dalam aliran data menuju dimensi konsumen adalah sebagai berikut:

1. Mengktraksi data dari sumber yaitu tabel konsumen dan tabel kota
2. Melakukan penggabungan antara tabel konsumen dengan tabel kota berdasarkan identitas relasinya.
3. Memuat data ke dimensi konsumen.
4. Dimensi pengiriman

Dimensi pengiriman dibentuk dari tabel jasa\_kirim. **Gambar 24** menggambarkan aliran data yang terjadi pada dimensi pengiriman.

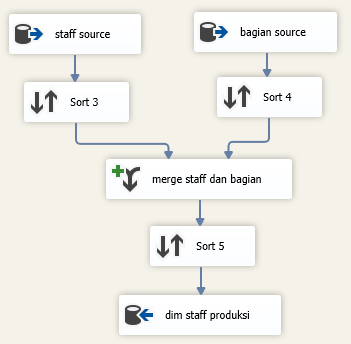


Gambar 24 *Data flow* dimensi pengiriman

Adapun *step by step*  dari aliran menuju dimensi pengiriman ini adalah sebagai berikut:

1. Mengktraksi data dari sumber yaitu tabel metode pengiriman.
2. Data yang akan digunakan dimasukkan ke dalam dimensi pengiriman.
3. Dimensi Staff Produksi

Dimensi staff produksi dibentuk dari tabel staff produksi dan tabel bagian dari sumber. **Gambar 25** menggambarkan aliran data yang terjadi pada dimensi staff produksi.

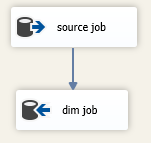


Gambar 25 *Data flow* dimensi staff produksi

Adapun *step by step*  aliran data menuju dimensi staff produksi adalah sebagai berikut:

1. Mengktraksi data staff produksi dan bagian dari sumber.
2. Menggabungkan data staff dan bagian berdasarkan relasinya.
3. Memuat data ke dimensi staff produksi.
4. Dimensi Job

Dimesi job dibentuk dari tabel job yang ada di sumber. **Gambar 26** menggambarkan aliran data yang terjadi pada dimensi job.

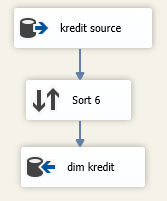


Gambar 26 *Data flow* dimensi job

Adapun *step by step* aliran data menuju dimensi job adalah sebagai berikut:

1. Mengkstrak data dari tabel job sumber.
2. Menentukan dan memuat atribut yang digunakan untuk dimuat ke dalam dimensi job.
3. Dimensi Kredit

Dimensi kredit dibentuk dari tabel kredit yang ada di sumber. **Gambar 27** menggambarkan aliran data yang terjadi di dimensi kredit.

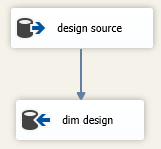


Gambar 27 *Data flow* dimensi kredit

Adapun *step by step* aliran data menuju dimensi kredit adalah sebagai berikut:

1. Mengkstrak data dari tabel kredit.
2. Membersihkan data dari redudansi.
3. Memuat data ke dimensi kredit.
4. Dimensi Design

Dimensi design dibentuk dari tabel design yang ada disumber. **Gambar 28** menggambarkan aliran data yang terjadi di dimensi design.

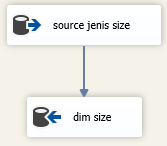


Gambar 28 *Data flow* dimensi design

Adapun *step by step* aliran data menuju dimensi design adalah sebagai berikut:

1. Mengkstrak data design dari sumber.
2. Menentukan dan memuat data ke dalam dimensi design.
3. Dimensi Size

Dimensi size dibentuk dari tabel size yang ada di sumber. **Gambar 29** menggambarkan aliran data yang terjadi pada dimensi size.



Gambar 29 *Data flow* dimensi size

Adapun *step by step* aliran data menuju dimensi size adalah sebagai berikut:

1. Mengkestrak data dari sumber.
2. Menentukan dan memuat data ke dimensi size.
3. Populasi Fakta

*Task* ini merupakan *data flow task* yang digunakan untuk mempopulasikan data dari data sumber menuju ke tabel fakta di *data warehouse.* Hal ini dilakukan untuk mendapatkan nilai bisnis yang dilakukan di tabel sumber untuk dijadikan nilai bisnis di tabel fakta. Selain itu Tabel fakta juga di bentuka berdasarkan hasil *lookup*  dari tabel dimensi yang memiliki relasi dengan tabel fakta agar mendapatkan kunci dai tabel dimensi yang akan dijadikan sebagai kunci tamu di tabel fakta. Berikut aliran data masing-masing tabel fakta dalam *data warehouse.*

1. Fakta Pemesanan Produk

Fakta pemesanan produk menunjukkan bisnis pemesanan produk yang terbentuk dari tabel pemesanan dan tabel *service order* yang ada di sumber dan hasil *lookup* dari tabel dimensi yang telah dibentuk. **Gambar 30**  menggambarkan aliran data yang terjadi di fakta pemesanan produk. Adapun *step by step* aliran data menuju fakta pemesanan produk adalah sebagai berikut:

1. Mengekstrak data dari tabel pemesanan dan tabel *service order*.
2. Menggabungkan kedua tabel berdasarkan relasinya
3. Melakukan *derived kolom* waktu menjadi data yang cocok untuk dilakukan *lookup*.
4. Melakukan *lookup* ke dimensi waktu, dimensi konsumen, dan dimensi produk.
5. Memuat data hasil *lookup*  ke dalam tabel fakta pemesanan produk.
6. Fakta Pengiriman Produk

Fakta pengiriman produk menunjukkan bisnis pengiriman suatu produk yang terbentuk dari tabel pengiriman, pemesanan, *order service,* dan metode pengiriman. **Gambar 31** menggambarkan aliran data yang terjadi pada fakta pengiriman produk. Adapun *step by step*  aliran data menuju fakta pengiriman produk adalah sebagai berikut:

1. Mengekstrak data dari tabel pengiriman, pemesanan, *order service*, dan metode pengiriman.
2. Menggabungkan masing-masing tabel berdasarkan relasinya.
3. Melakukan *derive* terhadap tanggal pengiriman
4. Melakukan *lookup* data pada dimensi waktu, dan dimensi produk.
5. Memuat data hasil *lookup* ke fakta pengiriman produk.
6. Fakta Belanja

Fakta belanja menunjukkan bisnis pengeluaran perusahan dalam pembelian alat dan bahan yang terbentuk dari tabel detail belanja, belanja, kredit, dan *service order*. **Gambar 32** menggambarkan aliran data yang terhadi pada fakta belanja. Adapun *step by step* aliran data menuju fakta belanja adalah sebagai berikut:

1. Mengekstrak data dari tabel detail belanja, belanja, kredit, dan *service order*.
2. Menggabungkan seluruh tabel berdasarkan dengan relasinya.
3. Melakukan *derive* waktu dan nominal.
4. Melakukan *lookup* data pada dimensi waktu, dimensi produk, dan dimensi kredit.
5. Memuat data hasil *lookup* ke fakta belanja.
6. Fakta Biaya Produksi

Fakta biaya produksi menunjukkan bisnis biaya sumber daya terhadap proses produksi yang terbantuk dari tabel produksi, *order service*, dan pemesanan. **Gambar 33** menggambarkan aliran data yang terjadi pada fakta biaya produksi. Adapun *Step by step* aliran data menuju fakta biaya produksi adalah sebagai berikut:

1. Mengekstrak data dari tabel produksi, *order service*, dan pemesanan
2. Menggabungkan data dari masing-masing tabel berdasarkan relasinya.
3. Melakukan *derive* waktu.
4. Melakukan *lookup* data pada dimensi waktu, dimensi produk, dimensi job, dan dimensi staff produksi.
5. Memuat data hasil *lookup* ke fakta biaya produksi.
6. Fakta Populer Produk

Fakta populer produk menujukkan bisnis kepopuleran suatu produk terhadap konsumen yang terbentuk dari tabel *service order* dan tabel pemesanan. Fakta yang dihasilkan dari tabel fakta ini akan dijadikan sebagai atribut target dalam konsep *fuzzy.* **Gambar 34** menunjukkan aliran data yang terjadi pada fakta populer produk. Adapun *step by step* aliran datanya adalah sebagai berikut:

1. Mengkstrak data dari tabel *order service*  dan pemesanan.
2. Menggabungkan data dari masing-masing tabel berdasarkan relasinya.
3. Melakukan *derive* waktu.
4. Melakukan *lookup* data pada dimensi waktu dan dimensi produk.
5. Memuat data hasil *lookup* ke fakta populer produk
6. Fakta Populer Design

Fakta populer design menunjukkan bisnis kepopuleran suatu design yang di gambarkan pada produk terhadap konsumen yang terbentuk dari tabel *design order*,*service order,* dan pemesanan. Fakta yang dihasilkan pada tabel fakta ini akan dijadikan sebagai target atribut dalam penggunaan konsp *fuzzy*. **Gambar 35** menggambarkan aliran data yang terjadi pada fakta kepopuleran produk. Adapun *step by step* aliran data fakta kepopuleran produk adalah sebagai berikut:

1. Mengekstrak data dari tabel *design order, service order,* dan pemesanan.
2. Menggabungkan data dari masing-masing tabel berdasarkan relasinya.
3. Melakukan *derive* waktu.
4. Melakukan *lookup* data pada dimensi waktu, dimensi produk, dimensi size.
5. Memuat data hasil *lookup* ke fakta kepopuleran produk.
6. Fakta Pembayaran

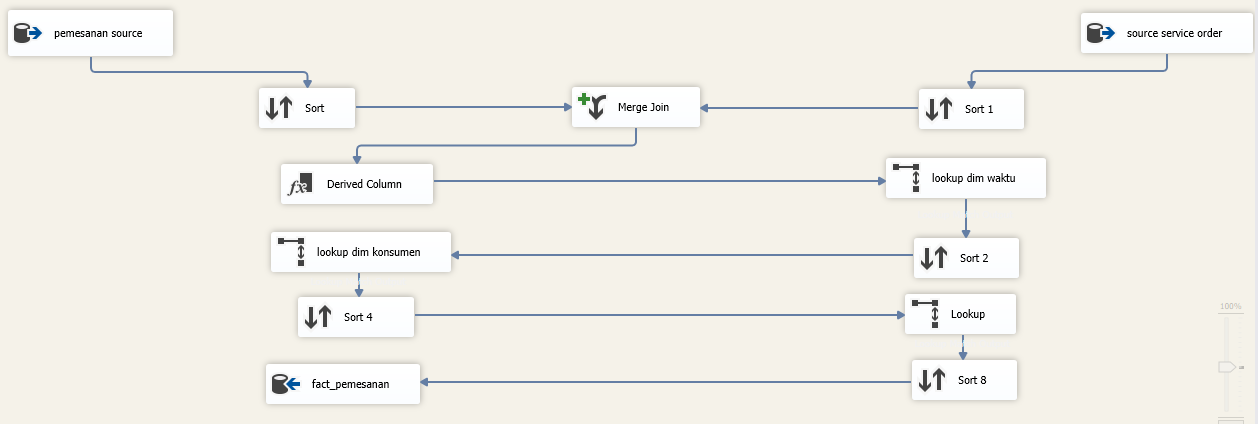
Fakta pembayaran menunjukkan bisnis pada rentang pemabayaran konsumen yang terbentuk dari tabel pemesanan, dan pembayaran. **Gambar 36** menunjukkan aliran data yang terjadi pada fakta pembayaran. Adapun *stepby step* aliran datanya adalah sebagai berikut:

1. Mengkstrak data dari tabel pemesanan, dan pembayaran.
2. Menggabungkan data dari masing-masing tabel berdasarkan relasinya.
3. Melakukan *derive* waktu.
4. Melakukan *lookup* data pada dimensi waktu dan dimensi konsumen.
5. Melakukan *derive* rentang pembayaran.
6. Memuat hasil data ke fakta pembayaran.
7. Fakta size Produk

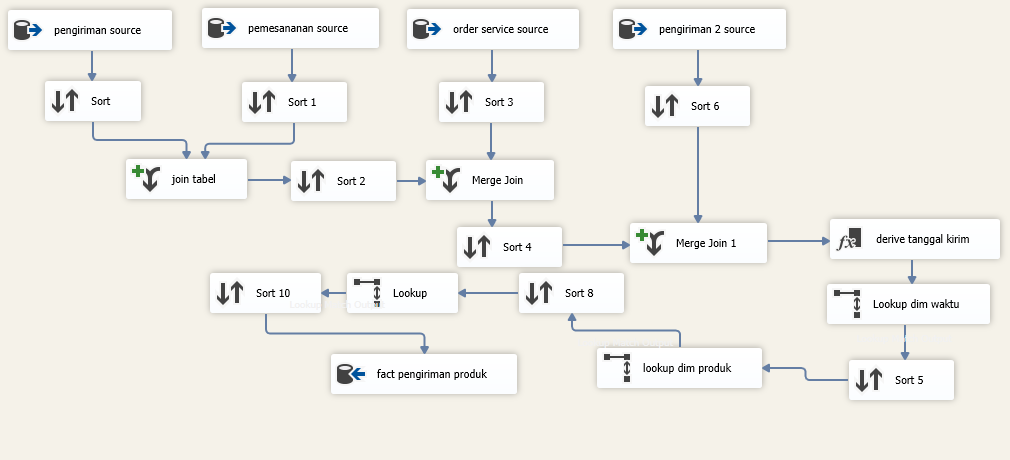
Fakta size produk menunjukkan bisnis ukuran yang digunakan produk yang tebentuk dari tabel *size order, service order,* dan pemesanan. **Gambar 37** menggambarkan aliran data yang terjadi pada fakta size produk. Adapun *step by step* aliran data menuju fakta size produk adalah sebagai berikut:

1. Mengekstrak data dari tabel *size order, service order,* dan pemesanan.
2. Menggabungkan data pada masing-masing tabel berdasarkan relasinya.
3. Melakukan *derive*  waktu.
4. Melakukan *lookup* data pada dimensi waktu, dimensi produk, dan dimensi size.
5. Memuat data hasil *lookup* ke fakta size produk.

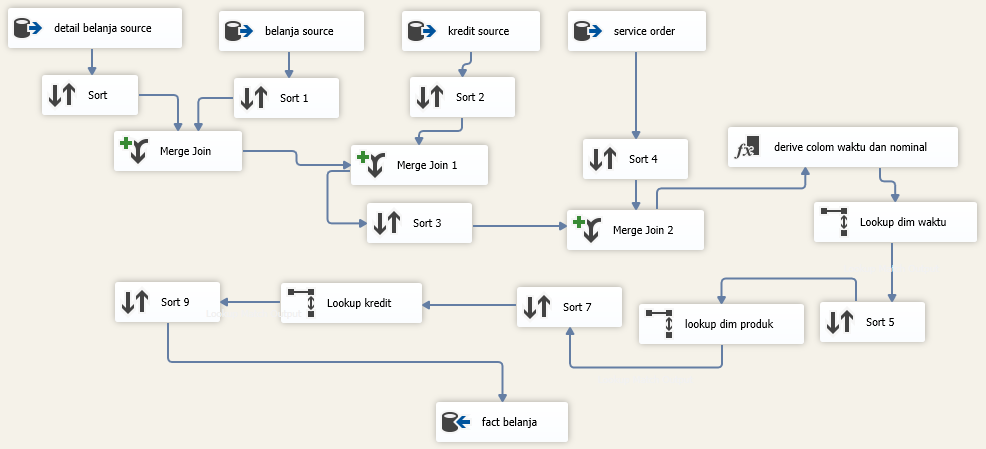
Pada proses *data staging* dalam suatu *data warehouse* akan berhenti disini, tetapi pada **Gambar 19** setelah mendefinisikan aliran data pada tabel fakta, beberapa tabel fakta memiliki *data flow task*  yaitu konsep *fuzzy. Task* *fuzzy* konsep yang ada pada **Gambar 19** akan dibahas pada point **3.1.6**.



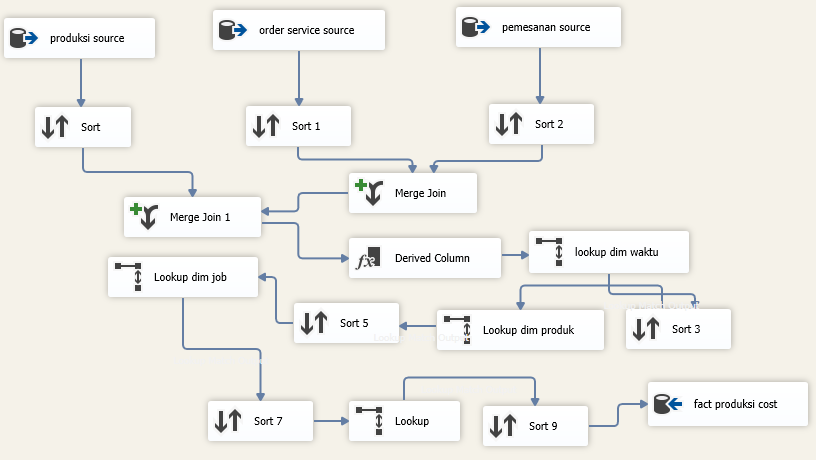
Gambar 30 *Data flow* fakta pemesanan produk



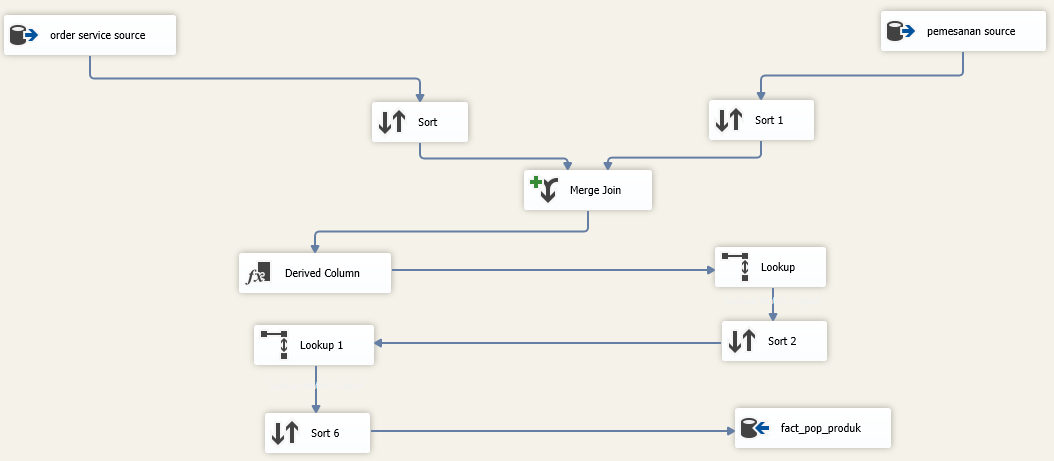
Gambar 31 *Data flow* fakta pengiriman produk



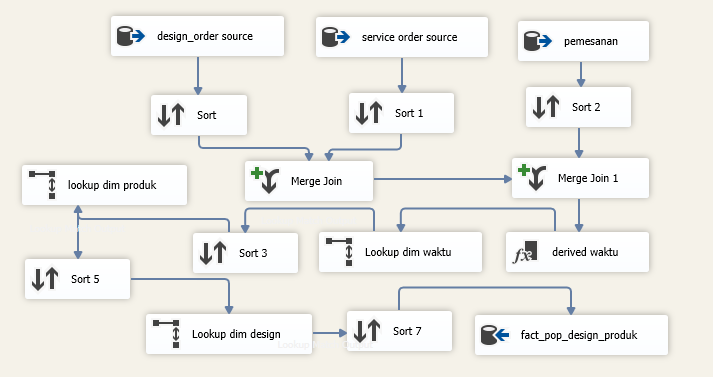
Gambar 32 *Data flow* fakta belanja



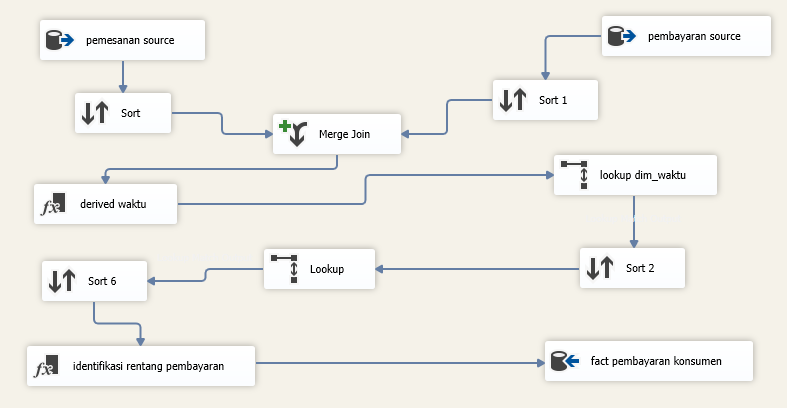
Gambar 33 *Data flow* fakta biaya produksi



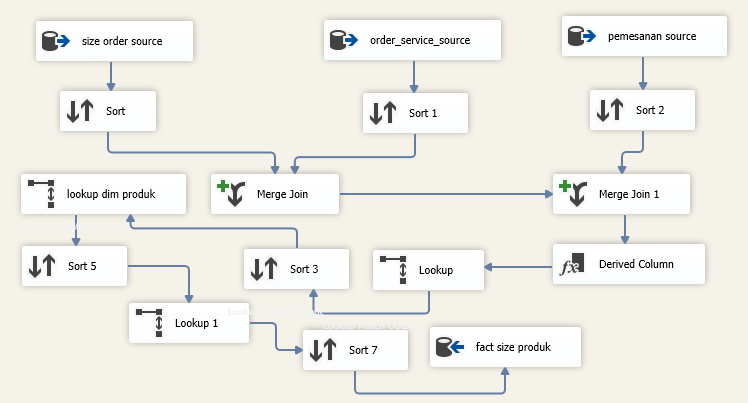
Gambar 34 *Data flow* fakta kepopuleran produk



Gambar 35 *Data flow* fakta kepopuleran design produk



Gambar 36 *Data flow* fakta pembayaran



Gambar 37 *Data flwo* fakta *size* produk

# 3.1.6 Analisis *Fuzzy Dimensional*

Analisis  *fuzzy dimensional* digunakan untuk menjelaskan hasil transformasi yang menghasilkan derajat keanggotaan yang diklasifikasikan seperti pada fakta persentasi\_pembayaran, pop\_design, dan pop\_produk.

Dalam proses transformasi tabel fakta\_pop\_produk terbentuk klasisfikasi terhadap jumlah produk di dalam tabel fakta. Pengklasifikasian tersebut terdiri dari kurang populer, cukup populer, dan sangat populer. Bila jumlah suatu produk di petakan dalam bentuk persentasi terhadap jumlah masing-masing produk yang dipesan oleh konsumen, maka dapat dibuat himpunannya seperti berikut:

1. Himpunan *fuzzy* kurang populer = 0 - 36
2. Himpunan *fuzzy* cukup populer = 20 – 60
3. Himpunan *fuzzy* sangat populer = 44 – ∞

Adapun fungsi keanggotannya adalah sebagai berikut:

Keterangan :

*x =* jumlah suatu produk yang dipesan oleh konsumen.

P = Pemetaan kurva menaik.

R = Pemetaan kurva menurun.

Berdasarkan fungsi keanggotaannya dapat dibangun kurva pemetaan kepopuleran suatu produk dalam periode waktu tertentu. Berikut **Gambar 5**  menunjukkan kurva pemetaan keanggotaan *fuzzy* kepopuleran produk.

Jumlah pemesanan produk dalam periode tertentu

persentase

AA

B

C

Gambar 38 Kurva pemetaan himpunan fuzzy jumlah produk

Keterangan :

A = Himpunan *fuzzy* kurang populer

B = Himpunan *fuzzy* cukup populer

C = Himpunan *fuzzy* sangat populer

Sebagai contoh, konsumen memsan produk dalam *Order Service* pada suatu transaksi sebagai berikut:

Tabel 54 Pemesanan produk oleh konsumen

| **Produk** | **Jumlah** |
| --- | --- |
| Polo | 25 |
| T-shirt | 52 |
| Kemeja | 24 |
| Jacket Jumper | 12 |
| Jacket Parasit | 12 |
| Jacket Jeans | 0 |
| Total | 125 |

Jadi menurut fungsi keanggotaannya, maka setiap produk akan dipetakan sebagai berikut:

Tabel 55 Kepopuleran produk berdasarkan pemetaan fungsi himpunan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Produk** | **Jumlah** | **Kurang Populer** | **Cukup Populer** | **Sangat Populer** |
| Polo | 25 | 0.69 | 0.31 | 0 |
| T-shirt | 52 | 0 | 0.5 | 0.5 |
| Kemeja | 24 | 0.75 | 0.25 | 0 |
| Jacket Jumper | 12 | 0.75 | 0 | 0 |
| Jacket Parasit | 12 | 0.75 | 0 | 0 |
| Jacket Jeans | 0 | 0 | 0 | 0 |

**Tabel 26** merupakan penjelasan hasil penerapan fuzzy dimensional sehingga dapat di-analisis berdasarkan kepopuleran suatu produk berdasarkan pemesanan konsumen. Pemetaan fungsi keanggotan berdasarkan klasifikasinya dapat juga diterapkan dalam pencarian fakta kepopuleran suatu design dalam produk. Sedangkan untuk penerapan fuzzy dimensional pembayaran konsumen, perlu di buat himpunan keanggotaan yang lain berdasarkan rentang waktu dari tanggal pembayaran dengan *dead line* pemesanan, berikut penjelasan himpunan keanggotaan untuk persentasi pembayaran konsumen.

1. Tepat waktu = 0 – 10
2. Sedikit terlambat = 7- 20
3. Terlambat = 16 – 30
4. Sangat terlambat = 25 - ∞

Bedasarkan himpunan keangotaan , dapat dibuat fungsi keanggotaannya sebagai berikut:

Berdasarkan fungsi keanggotaannya dapat dibangun kurva pemetaan persentasi pembayaran konsumen terhadap pemesanan yang dilakukan. Berikut **Gambar 7**  menunjukkan kurva pemetaan keanggotaan *fuzzy* pembayaran konsumen.

Rentang pembayaran kostumer

persentase

a

b

c

d

Gambar 39 Kurva pemetaan himpunan fuzzy pembayaran konsumen

Keterangan :

a = Pembayaran konsumen tepat waktu

b = Pembayaran konsumen sedikit terlambat

c = Pembayaran konsumen terlambat

d = Pembayaran konsumen sangat terlambat

# 3.2 Analisis OLAP dan Reporting Tools

Begitu selesai data staging, maka data *warehouse* telah terbentuk dan siap di lakukan analisis *OLAP* & *reporting tools*. Analisis *OLAP* & *reporting tools*  merupakan analisis yang dilakukan untuk menentukan bentuk *OLAP* & *reproting tools*  yang sesuai dalam penunjang pengambilan keputusan.

# 3.2.1 Analisis OLAP

OLAP mempresentasikan banyak data dalam bentuk multidimensi agar menjadi lebih mudah untuk melakukan analisis terhadap suatu informasi. Dalam penelitian ini, Metode OLAP yang digunakan adalah metode *Pivoting* yang memungkinkan pengguna untuk dapat melihat suatu nilai dalam tata – letak yang berbeda-beda demi kepentingan proses analisis dan penunjang pengambilan keputusan.

# 3.3 Analisis Kebutuhan

Analisis kebutuhan digunakan untuk menjelaskan spesifikasi yang dibutuhkan dalam pembangunan perangkat lunak dan batasan-batasan dalam pengimplementasian sistem. **Tabel 2** menjelaskan spesifikasi kebutuhan fungsional pada sistem yang dibangun dan **Tabel 3** menjelaskan spesifikasi kebutuhan non-fungsional.

Tabel 56 Spesifikasi Kebutuhan Fungsional

|  |  |
| --- | --- |
| **Kode Kebutuhan** | **Keterangan** |
| SKPL-F-1 | User dapat melakukan proses ETL di dalam sistem. |
| SKPL-F-2 | User dapat membuat OLAP cube sesuai keinginan. |
| SKPL-F-3 | User dapat membuat laporan didalam sistem. |
| SKPL-F-4 | User dapat melakukan pencetakan pada laporan. |

Tabel 57 Spesifikasi Kebutuhan Non-Fungsional

|  |  |
| --- | --- |
| **Kode Kebutuhan** | **Keterangan** |
| SKPL-NF-1-1 | Proses Ekstraksi pada data dilakukan secara otomatis didalam sistem saat user melakukan proses *ETL*. |
| SKPL-NF-1-2 | Proses Transformasi pada data dilakukan secara otomatis didalam sistem saat user melakukan proses *ETL*. |
| SKPL-NF-2-1 | Proses pembuatan *cube* *OLAP* dilakukan secara otomatis dalam penyesuaian kebutuhan *user*. |
| SKPL-NF-3-1 | Laporan yang di tampilkan oleh user sesuai dengan *cube OLAP* yang dilihat oleh *user*. |
| SKPL-NF-4-1 | Data yang diprint menggunakan *tamplate* yang telah ditentukan sesuai kebutuhan user. |

# 3.3.1 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional dibutuhkan untuk dapat menentukan fitur-fitur dari spesifikasi yang dibutuhkan oleh sistem. Spesifikasi ini meliputi elemen atau perangkat yang dibutuhkan untuk sistem yang dibangun sampai sistem siap diimplementasikan.

# 3.3.1.1 Analisis Perangkat Keras

Aplikasi berjalan dengan optimal apabila didukung dengan perangkat keras yang memadai. Spesifikasi perangkat keras dalam sistem yang dibangun adalah sebagai berikut:

Tabel 58 Spesifikasi perangkat keras yang dibutuhkan

|  |  |  |
| --- | --- | --- |
| **No** | **Perangkat keras** | **Spesifikasi** |
| 1 | Processor | 2.0 GHz |
| 2 | RAM | 2 GB |
| 3 | Hardisk | 128 GB |
| 4 | VGA | 128 bit kapasitas 512MB |
| 5 | Keyboard | Standard |
| 6 | Mouse | Standard |

Sedangkan spesifikasi perangkat keras yang ada saat ini adalah sebagai berikut:

Tabel 59 Spesifikasi perangkat keras yang ada saat ini

|  |  |  |
| --- | --- | --- |
| **No** | **Perangkat keras** | **Spesifikasi** |
| 1 | Processor | 3.2 GHz |
| 2 | RAM | 4 GB |
| 3 | Hardisk | 1024 GB |
| 4 | VGA | 128 bit kapasitas 2GB |
| 5 | Keyboard | Standard |
| 6 | Mouse | Standard |

Berdasarkan perbandingan analisis perangkat keras yang dibutuhkan dengan analisis perangkat keras yang ada saat ini, maka dapat disimpulkan bahwa perangkat keras yang telah ada saat ini telah memadai kebutuhan spesifikasi perangkat keras untuk menjalani sistem yang dibangun.

# 3.3.1.2 Analisis Perangkat Lunak

Analisis perangkat lunak digunakan untuk memenuhi spesifikasi perangkat lunak yang dibutuhkan untuk mendukung dalam pengimpelemntasian sistem yang dibangun. Adapun kebutuhn minimum perangkat lunak pendukung yang dibtuhkan untuk mendukung sistem adalah sebagai berikut:

Tabel 60 Spesifikasi perangkat lunak yang dibutuhkan

|  |  |  |
| --- | --- | --- |
| **No** | **Perangkat lunak** | **Spesifikasi** |
| 1 | Sistem Operasi | Windows 7 |
| 2 | Sql Server | Sql server 2014 |
| 3 | Runtime.net | Runtime.net 3.0,3.5,4.0,4.5 |

Sedangkan spesifikasi perangkat lunak yang ada saat ini adalah sebagai berikut.

Tabel 61 Spesifikasi perangkat lunak yang ada saat ini

|  |  |  |
| --- | --- | --- |
| **No** | **Perangkat lunak** | **Spesifikasi** |
| 1 | Sistem Operasi | Windows 8.1 |
| 2 | Sql Server | - |
| 3 | Runtime.net | Runtime.4.5 |

Berdasarkan perbandingan analisis dari perangkat lunak yang dibutuhkan dengan perangkat lunak yang ada saat ini, dibutuhkannya pembaharauan fitur perangkat lunak *framework* *runtime.net*  dari versi 3.0 hingga versi 4.5 dan aplikasi SQL *server 2014* untuk digunakan sebagai *database* sistem.

Tabel 62 Spesifikasi kebutuhan perangkat prikir

|  |  |
| --- | --- |
| **Pengguna Sistem** | Owner |
| **Tingkat Keterampilan yang harus dimiliki** | 1. Memahami konsep database secara dasar. 2. Menganalisa data 3. Pengoperasian komputer |
| **Pengalam yang harus dimiliki** | 1. Menggunakan sistem dengan konsep database 2. Menggunakan sistem *warehousing* |

Sedangkan perangkat pikir yang ada saat ini adalah sebagai berikut.

Tabel 63 Spesifikasi perangkat pikir yang ada

|  |  |
| --- | --- |
| **Pengguna Sistem** | Owner |
| **Tingkat Keterampilan yang harus dimiliki** | 1. Memahami konsep database 2. Menganalisa data 3. Pengoperasian komputer |
| **Pengalam yang harus dimiliki** | 1. Menggunakan sistem dengan konsep database. |

Berdasarkan perbandingan **Tabel 50**  dan **Tabel 51**, dapat disimpulkan penggunan sistem harus melakukan pelatihan tentang konsep sistem warehousing agar dapat menggunakan sistem yang dibangun secara optimal.

# 3.3.1.3 Analisis Perangkat Pikir

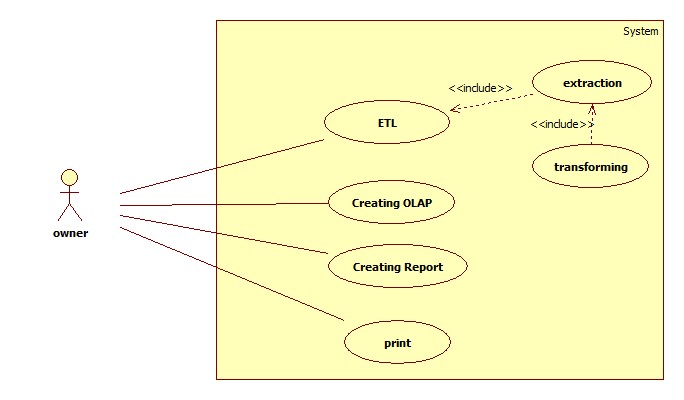
Analisis perangkat pikir digunakan untuk memmpersiapkan perangkat pikir yang pakar dalam menggunakan sistem yang dibangun agar sistem dapat berjalan dengan optimal. Adapun spesifikasi kebutuhan perangkat pikir pada sistem ini adalah sebagai berikut.

# 3.3.2 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional dilakukan untuk memodelkan fungsi sistem yang dibangun. Dalam penelitian ini, pemodelan dalam pembangunan sistem menggunakan *OOAD( Object Oriented Analysis and Design)*.

# 3.3.2.1 *Use Case Diagram*

*Use Case Diagram* digunakan dalam pemodelan untuk menjelaskan terhadap apa yang akan dilakukan oleh pengguna di dalam sistem. **Gambar 6** merupakan *use case diagram* pada perusahaan Spaceman yang digunakan untuk pemodelan terhadap apa saja yang dilakukan *owner* perushaan di dalam sistem.



Gambar 40 *Use case diagram data warehouse* pada perusahaan Spaceman

# 3.3.2.2 *Use Case Scenario*

*Use case scenario* digunakan untuk menjelaskan setiap *use case*  yang akan dilakukan oleh *user*. *Use case scenario* juga menjelaskan tahapan tahapan terhadap proses yang dilakukan oleh *user*,kondisi awal sebelum melakukan proses, kondisi sukses atau gagalnya suatu proses, dan pemicu terhadap proses yang dijalankan. Berikut merupakan *use case scenario* dari sistem yang dibangun.

1. *ETL*

*Use case scenario*  pada *use case* *ETL* menjelaskan proses, kondisi dan *user* pada saat melakukan proses *ETL* dan menjelaskan langkah-langkah proses *ETL* yang dideskripsikan pada **Tabel 53**.

Tabel 64 *Use case scenario ETL*

|  |  |  |
| --- | --- | --- |
| ***Use Case name*** | *ETL* | |
| ***Goal In Context*** | User melakukan proses *ETL* di dalam sistem. | |
| ***Preconditions*** | Sistem memiliki *data source* dan *data destination* | |
| ***Successful End Condition*** | Sistem berhasil memproses *ETL* | |
| ***Failed End Condition*** | Sistem gagal memproses *ETL* | |
| ***Primary Actors*** | User | |
| ***Secondary Actors*** | - | |
| ***Trigger*** | User meminta sistem melakukan proses *ETL* | |
| ***Main Flow*** | ***Step*** | ***Action*** |
| 1 | User meminta sistem melakukan proses ETL |
| 2 | Sistem membaca data source. |
| 3 | Sistem melakukan proses ekstraksi |
| 4 | Sistem melakukan proses transforming |
| 5 | Sistem memuat data ke data *destination*. |
| 6 | Menampilkan pesan status proses berhasil. |
| ***Extensions*** | ***Step*** | ***Branching Action*** |
| 2.1 | Sistem tidak menemukan data *source*. |
| 2.2 | Sistem membatalkan proses ETL |
| 2.3 | Menampilkan pesan status proses gagal. |

2. *Creating OLAP*

*Use case scenario*  pada *use case* *Creating OLAP* menjelaskan proses, kondisi dan *user* pada saat melakukan proses *Creating OLAP* dan menjelaskan langkah-langkah proses *Creating OLAP* yang dideskripsikan pada **Tabel 54**.

Tabel 65 *Use case scenario creating OLAP*

|  |  |  |
| --- | --- | --- |
| ***Use Case name*** | *Creating OLAP* | |
| ***Goal In Context*** | User membuat *OLAP.* | |
| ***Preconditions*** | Sistem memiliki *data destination* untuk pembuatan *OLAP*. | |
| ***Successful End Condition*** | Sistem berhasil memproses *OLAP* | |
| ***Failed End Condition*** | Sistem gagal memproses *OLAP* | |
| ***Primary Actors*** | User | |
| ***Secondary Actors*** | - | |
| ***Trigger*** | User meminta sistem membuat *OLAP*. | |
| ***Main Flow*** | ***Step*** | ***Action*** |
| 1 | User meminta sistem melakukan pembuatan *OLAP*. |
| 2 | Sistem membaca *data destination*. |
| 3 | Sistem membuat sturktur OLAP *cube* |
| 3 | Sistem menampilkan *OLAP.* |
| ***Extensions*** | ***Step*** | ***Branching Action*** |
| 2.1 | Sistem tidak menemukan *data* *destination* |
| 2.2 | Sistem membatalkan proses *creating OLAP* |
| 2.3 | Menampilkan pesan status proses gagal. |

3. *Creating Report*

*Use case scenario*  pada *use case* *Creating Report* menjelaskan proses, kondisi dan *user* pada saat melakukan proses *Creating Report* dan menjelaskan langkah-langkah proses *Creating Report* yang dideskripsikan pada **Tabel 55**.

Tabel 66 *Use case scenario creating report*

|  |  |  |
| --- | --- | --- |
| ***Use Case name*** | *Creating Report* | |
| ***Goal In Context*** | User membuat laporan | |
| ***Preconditions*** | Sistem memiliki *data destination* untuk pembuatan laporan | |
| ***Successful End Condition*** | Sistem berhasil memproses pembuatan laporan. | |
| ***Failed End Condition*** | Sistem gagal memproses pembuatan laporan | |
| ***Primary Actors*** | User | |
| ***Secondary Actors*** | - | |
| ***Trigger*** | User meminta sistem membuatlaporan | |
| ***Main Flow*** | ***Step*** | ***Action*** |
| 1 | User meminta sistem melakukan pembuatan laporan. |
| 2 | Sistem membaca *data destination*. |
| 3 | Sistem menampilkan laporan |
| ***Extensions*** | ***Step*** | ***Branching Action*** |
| 2.1 | Sistem tidak menemukan *data* *destination* |
| 2.2 | Sistem membatalkan proses pembuatan laporan. |
| 2.3 | Menampilkan pesan status proses gagal. |

4. *Print*

*Use case scenario*  pada *use case* *Print* menjelaskan proses, kondisi dan *user* pada saat melakukan proses *Print* dan menjelaskan langkah-langkah proses *Print* yang dideskripsikan pada **Tabel 56**.

Tabel 67 *Use case scenario print*

|  |  |  |
| --- | --- | --- |
| ***Use Case name*** | *Print* | |
| ***Goal In Context*** | User dapat memprint data. | |
| ***Preconditions*** | Sistem memiliki *template* untuk data yang ingin di cetak. | |
| ***Successful End Condition*** | Sistem berhasil memproses print data | |
| ***Failed End Condition*** | Sistem gagal memproses print data | |
| ***Primary Actors*** | User | |
| ***Secondary Actors*** | - | |
| ***Trigger*** | User meminta sistem melakukan *print*  data | |
| ***Main Flow*** | ***Step*** | ***Action*** |
| 1 | User meminta sistem melakukan proses *print*. |
| 2 | Sistem memuat *template* untuk di *print.* |
| 3 | Sistem memprint data. |
| ***Extensions*** | ***Step*** | ***Branching Action*** |
| 2.1 | Sistem tidak dapat memuat *template* |
| 2.2 | Sistem membatalkan proses *print*. |
| 2.3 | Menampilkan pesan status proses gagal. |

5. *Extraction*

*Use case scenario*  pada *use case* *print* menjelaskan proses, kondisi dan *user* pada saat melakukan proses *extraction* yang ada di dalam proses *ETL* dan menjelaskan langkah-langkah proses *print* yang dideskripsikan pada **Tabel 57**.

Tabel 68 *Use case scenario extraction*

|  |  |  |
| --- | --- | --- |
| ***Use Case name*** | *Extraction* | |
| ***Goal In Context*** | Sistem dapat melakukan ekstraksi data | |
| ***Preconditions*** | Sistem telah memiliki data *source* | |
| ***Successful End Condition*** | Sistem berhasil melakukan ekstraksi data | |
| ***Failed End Condition*** | Sistem gagal melakukan ekstraksi data | |
| ***Primary Actors*** | User | |
| ***Secondary Actors*** | - | |
| ***Trigger*** | User meminta untuk melakukan proses ETL | |
| ***Main Flow*** | ***Step*** | ***Action*** |
| 1 | Sistem membuka data *source* |
| 2 | Sistem melakukan ektraksi pada data *source* |
| 3 | Sistem berhasil melakukan proses *extraction* |
| ***Extensions*** | ***Step*** | ***Branching Action*** |
| 1.1 | Sistem gagal membuka data |
| 1.2 | Sistem membatalkan proses *extraction*. |
| 1.3 | Menampilkan pesan status proses gagal. |

6*. Transforming*

*Use case scenario*  pada *use case* *print* menjelaskan proses, kondisi dan *user* pada saat melakukan proses *extraction* yang ada di dalam proses *ETL* dan menjelaskan langkah-langkah proses *print* yang dideskripsikan pada **Tabel 58**.

Tabel 69 *Use case scenario transforming*

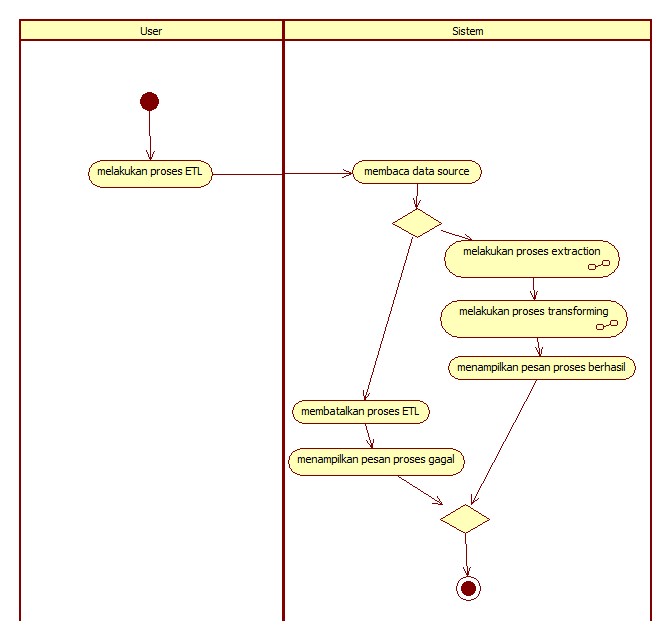
|  |  |  |
| --- | --- | --- |
| ***Use Case name*** | *Transforming* | |
| ***Goal In Context*** | Sistem dapat melakukan *transforming data* | |
| ***Preconditions*** | Sistem telah melakukan proses *extraction* | |
| ***Successful End Condition*** | Sistem berhasil melakukan *transforming* data | |
| ***Failed End Condition*** | Sistem gagal melakukan *transforming* data | |
| ***Primary Actors*** | User | |
| ***Secondary Actors*** | - | |
| ***Trigger*** | Sistem melakukan proses *extraction* | |
| ***Main Flow*** | ***Step*** | ***Action*** |
| 1 | Sistem membaca data hasil ekstraksi |
| 2 | Sistem melakukan proses *cleaning* dan *conditioning* data |
| 3 | Sistem berhasil melakukan proses *transforming*  data. |
| ***Extensions*** | ***Step*** | ***Branching Action*** |
| 1.1 | Sistem tidak dapat membaca hasil ekstraksi |
| 2.2 | Sistem berhasil melakukan proses transforming data. |

# 3.3.2.3 *Activity Diagram*

*Activity diagram* digunakan untuk memodelkan alur aktivitas dari suatu *use case.* Berikut merupakan *activity* *diagram*  yang merepresentasikan alur aktivitas *use case* pada sistem yang dibangun.

1. *Activity Diagram ETL*

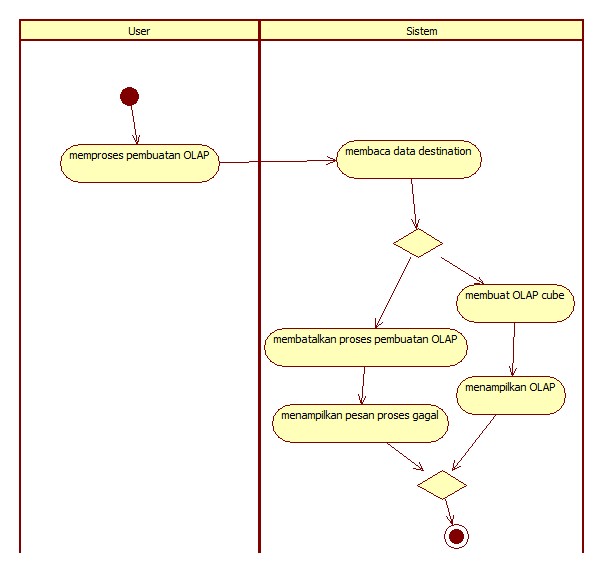
**Gambar 16** merupakan diagram alur aktivitas yang di jelaskan dalam *use case scenario ETL*.



Gambar 41 *Activity diagram ETL*

2*. Activity Diagram Creating OLAP*

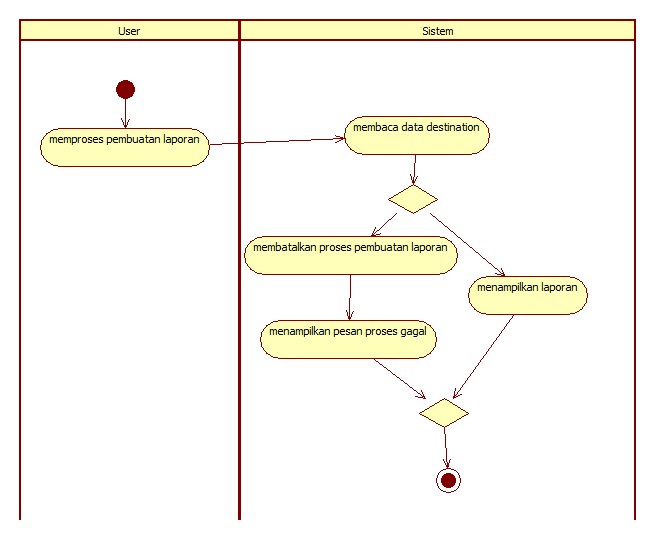
**Gambar 17** merupakan diagram alur aktivitas yang di jelaskan dalam *use case scenario creating OLAP*.

**

Gambar 42 *Activity diagram creating OLAP*

*3. Activity Diagram Creating Report*

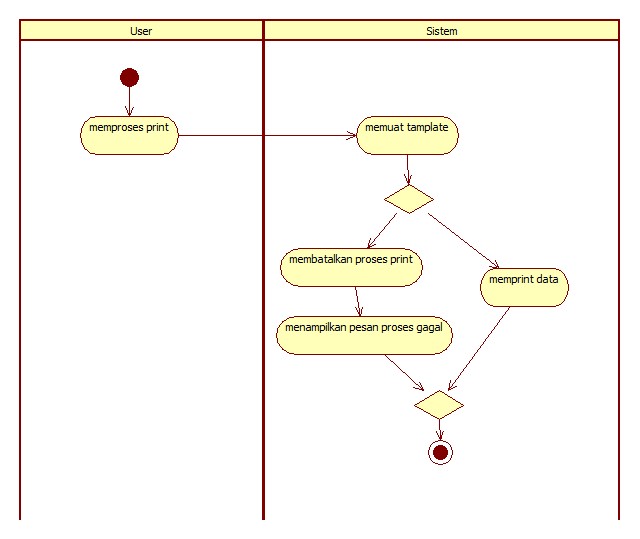
**Gambar 18** merupakan diagram alur aktivitas yang di jelaskan dalam *use case scenario creating report.*

**

Gambar 43 *Activity diagram creating report*

*4. Activity Diagram Print*

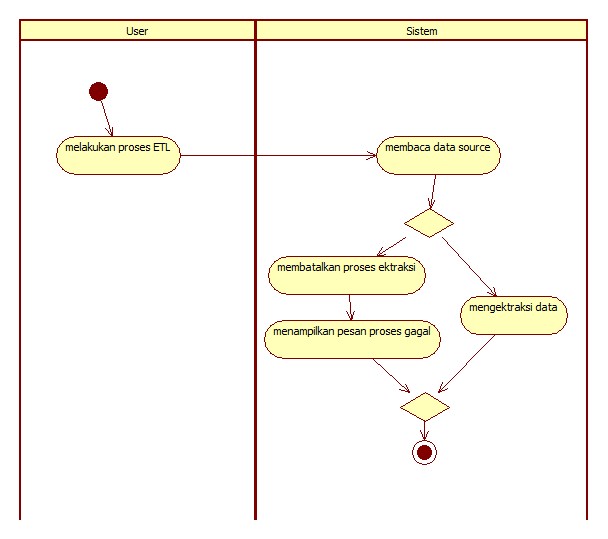
**Gambar 19** merupakan diagram alur aktivitas yang di jelaskan dalam *use case scenario print.*

**

Gambar 44 *Activity diagram print*

*5. Activity Diagram Extraction*

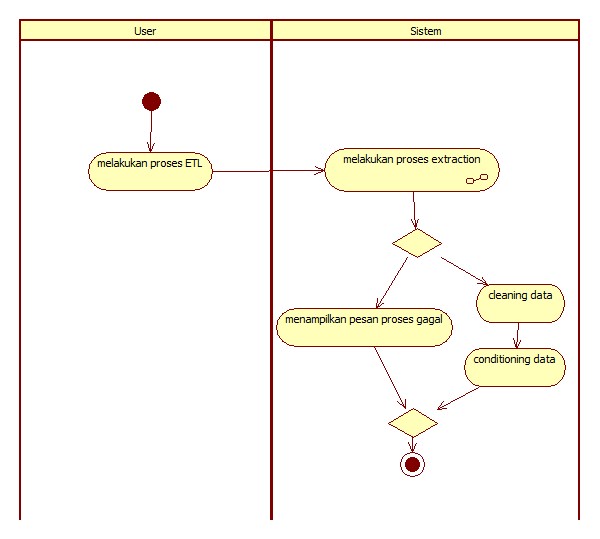
**Gambar 20** merupakan diagram alur aktivitas yang di jelaskan dalam *use case scenario print.*

**

Gambar 45 *Acrivity diagram extraction*

*6. Activity Diagram Transforming*

**Gambar 21** merupakan diagram alur aktivitas yang di jelaskan dalam *use case scenario print.*

**

Gambar 46 *Activity diagram transforming*

# 3.4 Analisis Model

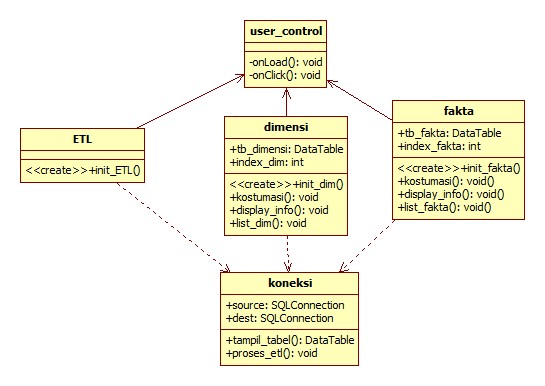
Analisis Model digunakan untuk merealisasikan *use case* yang ada pada **Gambar 6** kedalam kelas yang dibutuhkan untuk guna menjalankan proses yang ada didalam *use case*. Analisis ini juga menggambarkan keseluruhan kelas yang akan digunakan didalam sistem.

## 3.4.1 Realisasi *Use Case* Tahap Analisis

realisasi *use case* tahap analisis digunakan untuk menjelaskan keterkaitan kegunaan kelas didalam proses *use case*. Berikut merupakan analisis realisasi *use case* terhadap semua proses.

1. *ETL*

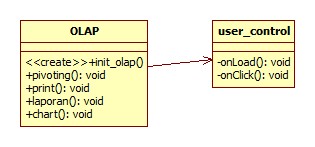
Fungsional *ETL* melibatkan lima kelas yaitu koneksi, ETL, dimensi, fakta, dan user\_control. Fungsional dijalankan dengan menunggu masukan dari *user*. **Gambar 22** merupakan relasi kelas yang akan digunakan dalam proses *ETL*.



Gambar 47 Diagram kelas pada fungsional ETL

2. *Creating OLAP*

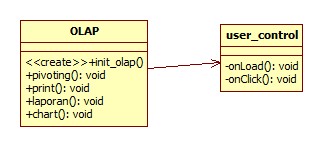
Fungsional *creating OLAP* melibatkan tiga kelas yaitu olap, fakta dan user\_control. Fungsional dijalankan dengan menunggu masukan dari *user.* **Gambar 23** merupakan relasi kelas yang akan digunakan dalam proses *Creating OLAP*.



Gambar 48 Diagram Kelas pada fungsional *Creating OLAP*

3. *Creating Report*

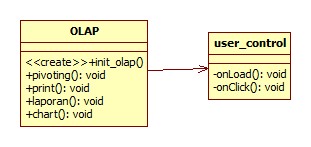
Fungsional *creating report* melibatkan tiga kelas yaitu OLAP, fakta dan user\_control. Fungsional dijalankan dengan menunggu masukan dari *user*  dimana proses yang dilakukan berasal dari subrutin dari kelas OLAP*.* **Gambar 24** merupakan relasi kelas yang akan digunakan dalam proses *Creating Report*.



Gambar 49 Kelas diagram pada fungsional *Creating Report*

4. *Print*

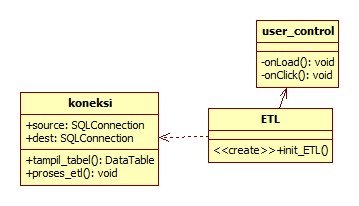
Fungsional *print* melibatkan tiga kelas yaitu OLAP, fakta dan user\_control. Fungsional dijalankan dengan menunggu masukan dari *user*  dimana proses yang dilakukan berasal dari subrutin dari kelas OLAP*.* **Gambar 25** merupakan relasi kelas yang akan digunakan dalam proses *print*.



Gambar 50 Kelas digram pada fungsional *print*

5. *Extraction*

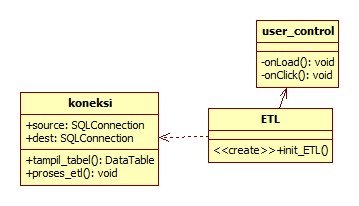
Fungsional *extraction* melibatkan tiga kelas yaitu koneksi, ETL dan user\_control. Fungsional dijalankan secara otomatis dimana proses yang dilakukan berasal dari subrutin dari kelas OLAP*.* **Gambar 26** merupakan relasi kelas yang akan digunakan dalam proses *extraction*.



Gambar 51 Kelas diagram pada fungsional *extraction*

6. *Transforming*

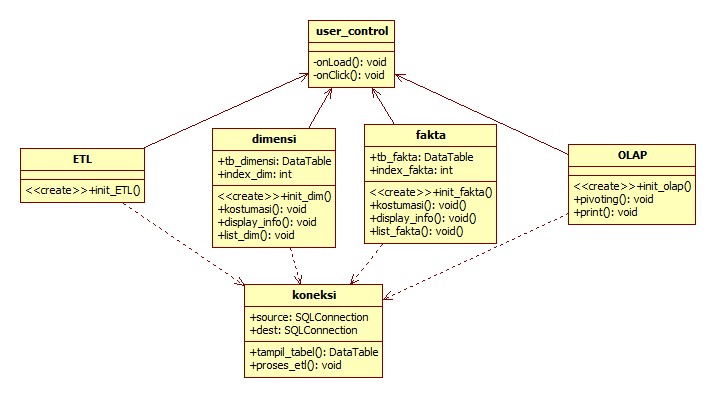
Fungsional *transforming* melibatkan tiga kelas yaitu koneksi, ETL dan user\_control. Fungsional dijalankan secara otomatis dimana proses yang dilakukan berasal dari subrutin dari kelas OLAP*.* **Gambar 27** merupakan relasi kelas yang akan digunakan dalam proses *transforming*.



Gambar 52 Kelas diagram pada fungsional *transforming*

## 3.4.2 Kelas Diagram Keseluruhan

**Gambar 28** merupakan diagram kelas keseluruhan yang digunakan di dalam sistem yang telah terealisasi oleh fungsional yang ada di dalam sistem.



Gambar 53 Diagram kelas keseluruhan

## 3.4.3 Analisis Kelas

Analisis kelas digunakan untuk menjelaskan jenis-jenis kelas yang digunakan dan memaparkan tanggung jawab kelas terhadap fungsionalitasnya. **Tabel 59** menjelaskan jenis-jenis kelas yang digunakan di dalam sistem.

Tabel 70 Jenis kelas yang digunakan di dalam sistem

| **No.** | **Nama Kelas** | **Jenis** |
| --- | --- | --- |
| 1 | User\_control | Boundary |
| 2 | ETL | Model |
| 3 | Koneksi | Control |
| 4 | OLAP | Model |
| 5 | Dimensi | Model |
| 6 | Fakta | Model |

**Tabel 60** menjelaskan tanggung jawab pada masing-masing kelas yang digunakan didalam sistem.

Tabel 71 Tanggung jawab kelas

| **Nama Kelas** | **Tanggung Jawab** | **Attribut** |
| --- | --- | --- |
| User\_control | 1. Mengontrol masukan yang dilakukan oleh user. 2. Mengimplementasikan relasi kelas terhadap kelas sendiri. |  |
| ETL | 1. Memproses data *source* meliputi ekstraksi dan transformasi sampai bisa di muat ke data *destination*. |  |
| Koneksi | 1. Menghubungkan sistem ke data *source*. 2. Menghubungkan sistem ke data *destination*. 3. Mengolah proses ETL. 4. Memanggil data yang diperlukan dengan balikan berupa tabel. | 1. Source 2. Destination |
| OLAP | 1. Membuat *cube* multidimensi yang dapat dilihat oleh user. 2. Membuat laporan. 3. Melakukan pencetakan. |  |
| Dimensi | 1. Menampung data dimensi yang digunakan. 2. Mengkostumasi tampilan dimensi. 3. Membuat list dimensi yang digunakan, 4. Menampilkan dimensi yang ada. | 1. Tb\_dimensi 2. Index\_dim |
| Fakta | 1. Menampung fakta yang digunakan didalam sistem. 2. Mengkostumasi fakta. 3. Menampilkan informasi fakta yang ada di dalam sistem. 4. Membuat list fakta yang digunakan di dalam sistem. | 1. Tb\_fakta 2. Index\_fakta |

# 3.5 Model Perancangan

Model perancangan digunakan untuk menjelaskan model analisis yang telah dipaparkan pada point **3.4.** Dalam model penelitian kelas yang dibuat akan di rancang sesuai alur proses ke dalam *sequence diagram*.

## 3.5.1 Realisasi *Use Case* Tahap Perancangan

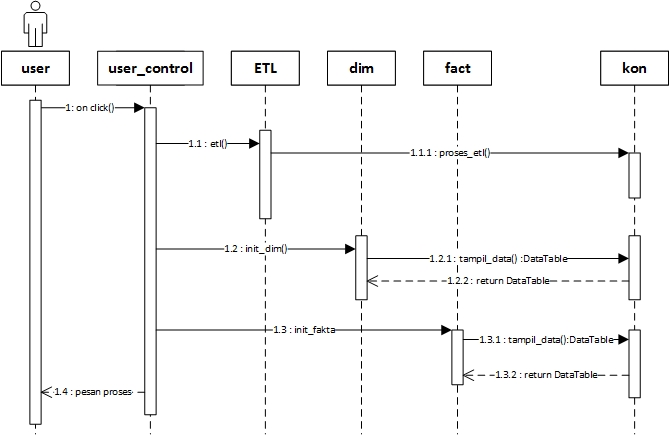
Realisasi *use case* tahap perancangan merupakan tahap merancang apa yang dijelaskan pada point **3.4.1**. Tahapan ini melibatkan perancangan kelas yan digunakan dan memodelkan alur proses dari *use case* terhadap kelas yang digunakan.

1. *ETL*

**Tabel 61** menjelaskan keterkaitan kelas yang dirancangan dengan kelas yang dianalisis dan pada **Gambar 29** merupakan *sequence diagram* pada *use case ETL.*

Tabel 72 Identifikasi perancangan kelas pada *use case ETL*

| **No.** | **Perancangan Kelas** | **Analisis Kelas Terkait** |
| --- | --- | --- |
| 1 | koneksi | Koneksi |
| 2 | ETL | ETL |
| 3 | User\_control | User\_control |
| 4 | Dimensi | Dimensi |
| 5 | Fakta | Fakta |



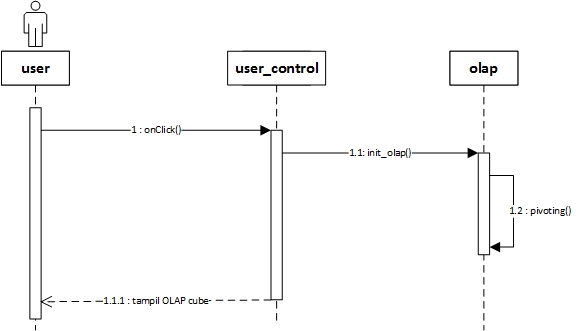
Gambar 54 *Sequence diagram* pada *use case ETL*

2. *Creating OLAP*

**Tabel 62** menjelaskan keterkaitan kelas yang dirancang dengan kelas yang dianalisis dan pada **Gambar 30** merupakan *sequence diagram* pada *use case creating OLAP.*

Tabel 73 Identifikasi perancangan kelas pada *use case creating OLAP*

| **No.** | **Perancangan Kelas** | **Analisis Kelas Terkait** |
| --- | --- | --- |
| 3 | User\_control | User\_control |
| 4 | Olap | Olap |



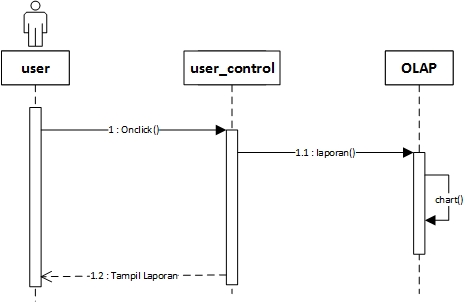
Gambar 55 *Sequence diagram* pada *use case creating OLAP*

3. *Creating Report*

**Tabel 63** menjelaskan keterkaitan kelas yang dirancang dengan kelas yang dianalisis dan pada **Gambar 31** merupakan *sequence diagram* pada *use case creating OLAP.*

Tabel 74 Identifikasi perancangan kelas pada *use case creating report*

| **No.** | **Perancangan Kelas** | **Analisis Kelas Terkait** |
| --- | --- | --- |
| 3 | User\_control | User\_control |
| 4 | Olap | Olap |



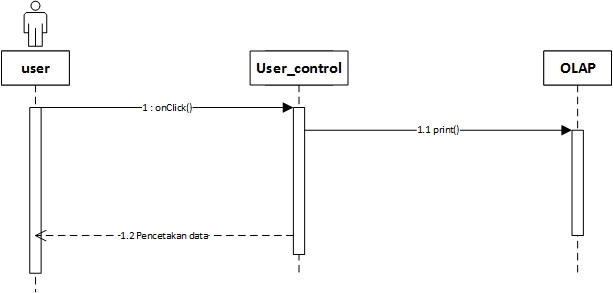
Gambar 56 *Sequence diagram* pada *use case creating report*

4. *Print*

**Tabel 64** menjelaskan keterkaitan kelas yang dirancang dengan kelas yang dianalisis dan pada **Gambar 32** merupakan *sequence diagram* pada *use case creating OLAP.*

Tabel 75 Identifikasi perancangan kelas pada *use case print*

| **No.** | **Perancangan Kelas** | **Analisis Kelas Terkait** |
| --- | --- | --- |
| 3 | User\_control | User\_control |
| 4 | Olap | Olap |



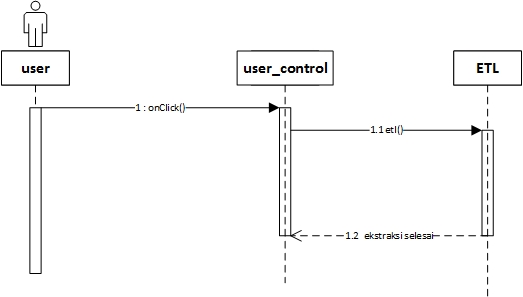
Gambar 57 *Sequence diagram* pada *use case print*

5. *Extraction*

**Tabel 65** menjelaskan keterkaitan kelas yang dirancang dengan kelas yang dianalisis dan pada **Gambar 33** merupakan *sequence diagram* pada *use case creating OLAP.*

Tabel 76 Identifikasi perancangan kelas pada *use case extraction*

| **No.** | **Perancangan Kelas** | **Analisis Kelas Terkait** |
| --- | --- | --- |
| 3 | User\_control | User\_control |
| 4 | ETL | ETL |



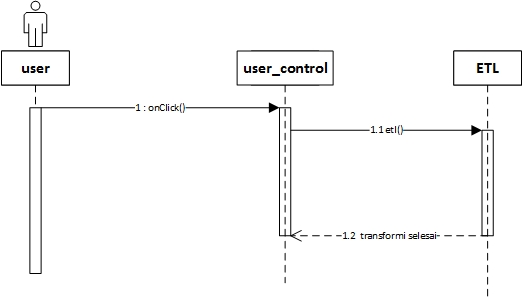
Gambar 58 *Sequence diagram* pada *use case extraction*

6. *Tarnsforming*

**Tabel 66** menjelaskan keterkaitan kelas yang dirancang dengan kelas yang dianalisis dan pada **Gambar 34** merupakan *sequence diagram* pada *use case creating OLAP.*

Tabel 77 Identifikasi perancangan kelas pada *use case transforming*

| **No.** | **Perancangan Kelas** | **Analisis Kelas Terkait** |
| --- | --- | --- |
| 3 | User\_control | User\_control |
| 4 | ETL | ETL |

**

Gambar 59 *Sequence diagram* pada *use case* *transforming*

## 3.5.2 Perancangan Kelas

Perancangan kelas digunakan untuk mendeskripsikan atribut dan operasi yang digunakan didalam sistem. Berikut perancangan kelas pada sistem ini:

1. *User Control*

Kelas *user control* digunakan sebagai fasilitas perantara sistem dengan interaksi user yang akan memicu kerja sistem. **Tabel 67** menjelaskan operasi pada kelas *user control.*

Tabel 78 Perancangan operasi pada kelas *user kontrol*

| **Nama Operasi** | **Visibility** | **Keterangan** |
| --- | --- | --- |
| onClick():void | *Private* | Mengontrol kejadian disaat user memasuikkan inputan berupa klik pada mouse. |
| onLoad() | *Private* | Mengontrol komponen yang akan disajikan pada sistem dibuka. |

2. *ETL*

Kelas *ETL*  digunakan untuk melakukan proses ekstraksi , transformasi, dan memuat data dari data *source*  ke data *destination*. **Tabel 68** menjelaskan operasi pada kelas *ETL*.

Tabel 79 perancangan operasi pada kelas *ETL*

| **Nama Operasi** | **Visibility** | **Keterangan** |
| --- | --- | --- |
| init\_etl() | *Public* | Membuat proses ekstraksi, transformasi, dan memuat data dari data *source* ke data *destination*. |

3. dimensi

Kelas dimensi digunakan untuk menyediakan tabel dimensi termasuk dengan operasi yang digunakan untuk menyajikan tabel dimensi. **Tabel 69** dan **Tabel 70** menjelaskan operasi dan atribut pada kelas dimensi.

Tabel 80 perancangan operasi kelas dimensi

| **Nama Operasi** | **Visibility** | **Keterangan** |
| --- | --- | --- |
| Init\_dim() | *Public* | Memuat data tabel dimensi. |
| Kostumasi() | *Public* | Mengontrol tampilan tabel dimensi. |
| Display\_info() | *Public* | Menyajikan tabel dimensi. |
| List\_dimensi() | *Public* | Menyusun urutan tabel dimensi. |

Tabel 81 Perancangan atribut kelas dimensi

| **Nama Atribut** | **Visibility** | **Tipe Data** |
| --- | --- | --- |
| Tb\_dimensi | *Public* | Data Table |
| Index\_dim | *Public* | Integer |

4. Fakta

Kelas fakta digunakan untuk menyediakan tabel fakta di dalam sistem yang terdiri dari atribut dan operasi didalamnya. **Tabel 71** dan **Tabel** **72** menjelaskan operasi dan atribut yang ada pada kelas fakta.

Tabel 82 perancangan operasi pada kelas fakta

| **Nama Operasi** | **Visibility** | **Keterangan** |
| --- | --- | --- |
| Init\_fakta() | *Public* | Memuat data tabel fakta. |
| Kostumasi() | *Public* | Mengontrol tampilan tabel fakta. |
| Display\_info() | *Public* | Menyajikan tabel fakta. |
| List\_fakta() | *Public* | Menyusun urutan tabel fakta. |

Tabel 83 perancangan atribut pada kelas fakta

| **Nama Atribut** | **Visibility** | **Tipe Data** |
| --- | --- | --- |
| Tb\_fakta | *Public* | Data Table |
| Index\_fakta | *Public* | Integer |

5. *OLAP*

Kelas OLAP digunakan untuk menyajikan *OLAP* *cube* yang akan digunakan user sebagai analsis yang bersifat multidimensional, pembuatan laporan, dan melakukan pencetakan. **Tabel 73** akan menjelaskan operasi yang ada pada kelas *OLAP*.

Tabel 84 Perancangan operasi pada kelas *OLAP*

| **Nama Operasi** | **Visibility** | **Keterangan** |
| --- | --- | --- |
| Init\_olap() | *Public* | Memuat *OLAP*. |
| pivoting() | *Public* | Membentuk cube *OLAP* |
| Print() | *Public* | Mencetak data |
| Laporan() | *Public* | Membuat laporan |
| Chart() | *Public* | Membuat data menjadi diagram berupa chart. |

6. Koneksi

Kelas koneksi digunakan untuk menghubungkan sistem ke data*source* dan menghubungkan ke data *destination*. Selain itu, operasi pada kelas koneksi juga melibatkan proses *ETL* dan penampilan data. **Tabel 74** dan **Tabel 75** mendiskripsikan operasi dan atribut yang ada pada kelas koneksi.

Tabel 85 Perancangan operasi pada kelas koneksi

| **Nama Operasi** | **Visibility** | **Keterangan** |
| --- | --- | --- |
| Tampil\_tabel() | *Public* | Membaca isi data dari data *source* maupun dari data *destination*. |
| Proses\_etl() | *Public* | Memproses ETL |

Tabel 86 Perancangan atribut pada kelas koneksi

| **Nama Atribut** | **Visibility** | **Tipe Data** |
| --- | --- | --- |
| Source | *Public* | SQLConnection |
| Dest | *Public* | SQLCOnnection |

# 3.4 Perancangan Antarmuka

# BAB IV

**IMPLEMENTASI DAN PENGUJIAN SISTEM**

# BAB V

**KESIMPULAN**

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | L. Sapir, A. Schimilovici dan L. Rokach, “A Methodolgy for The Design of a Fuzzy Data Warehouse,” *Intelligent System,* vol. 1, no. 1, pp. 2.14 - 2.21, 2008. |
| [2] | P. D. Sugiyono, Metode Penelitian Kuantitatif, Kualtitatif dan Kombinasi, Bandung: Alfabeta, 2014. |
| [3] | I. Sommerville, Software Engineering Ninth Edition, New York: Pearson Education, 2011. |
| [4] | R. Kimball dan M. Ross, The Data Warehouse Toolkit, Indianapolis: John WIlley & Sons, 2013. |
| [5] | D. Fasel, Fuzzy Data Warehousing for Performance Measurement : Concept and Implementation, Switzerland: Springer, 2014. |
| [6] | A. Silberschatz, F. K. Henry dan S. Sudarshan, Database System Concepts, Sixth edition, New York: McGraw Hill Companies, 2011. |
| [7] | P. Ponniah, Data Warehousing Fundamental, New York: John Willey & Sons, INC., 2001. |
| [8] | H. A. Jefrey, G. F. Joey dan J. S. Valacich, Modern Systems Analysis and Design Sixth Edition, New York: Pearson Education, 2011. |
| [9] | K. Hamilton dan R. Miles, Learning UML 2.0, Sebastopol: O'Reilly Media Inc, 2006. |