BAB II
TINJAUAN PUSTAKA

II.1 Profil Instansi

PT. Focus Gaya Graha yang berlokasi di Jalan Leuwigajah No. 106 Cimahi, bergerak dalam bidang industi *furniture*. Pertama kali didirikan pada tanggal 21 Mei 1995 oleh Sigit Sasmitapura, PT. Focus Gaya Graha terus berkembang sampai saat ini. Jenis produksi yang ada di PT. Focus Gaya Graha dibagi menjadi 3 yaitu *wood furniture*, *panel furniture* dan *photo frame*. Ada beberapa jenis bahan baku yang digunakan dalam proses produksi, diantaranya adalah kayu mahoni, kayu albasia, kayu pinus, kayu meranti, kayu jemitri, kayu bayur, MDF dan partikel *board*.

Produk-produk yang dihasilkan oleh PT. Focus Gaya Graha banyak diminati oleh konsumen dalam negeri maupun luar negeri. Beberapa konsumen dari negara-negara yang berada di benua Eropa, Amerika dan Asia, telah menjadi pelanggan tetap perusahaan ini. Saat ini, dengan 450 orang pekerja dan luas pabrik 12.000m^2, PT. Focus Gaya Graha memiliki kapasitas produksi hingga 25 kontainer per bulan.

II.1.1 Logo Instansi

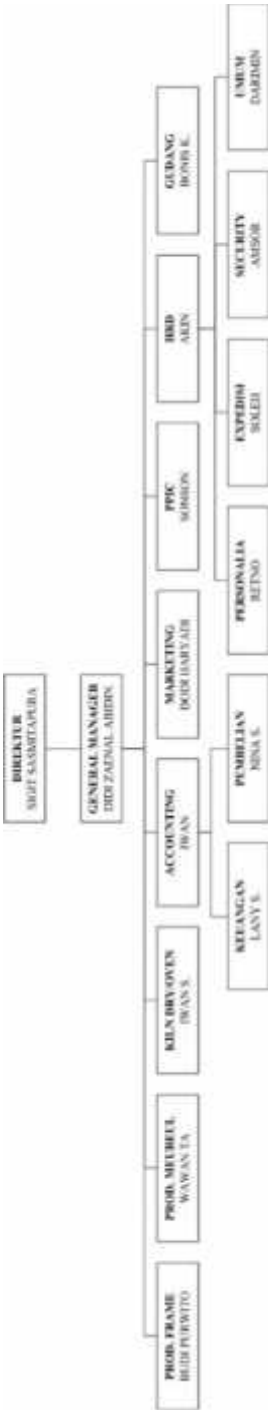
Berikut ini adalah logo dari PT. Focus Gaya Graha yang dapat dilihat pada Gambar II.1 :



Gambar II.1 Logo Instansi

II.1.2 Struktur Organisasi

Berikut ini adalah struktur organisasi PT. Focus Gaya Graha yang dapat dilihat pada Gambar II.2 :



Gambar II.2 Struktur Organisasi

II.2 Landasan Teori

Landasan teori yang berkaitan dengan materi atau teori yang digunakan sebai acuan dalam melakukan penelitian. Landasan teori yang diuraikan merupakan hasil studi literatur, buku-buku, maupun situs internet.

II.2.1 Data

Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya [6]. Dalam pendekatan basis data tidak hanya berisi basis data itu sendiri tetapi juga termasuk definisi atau deskripsi dari data yang disimpan. Definisi data disimpan dalam sistem katalog, yang berisi informasi tentang struktur tiap berkas, tipe dan format penyimpanan tiap item data, dan berbagai konstrin dari data. Semua informasi yang disimpan dalam katalog ini biasa disebut meta-data [7].

II.2.2 Basis Data

Basis data adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah sesuatu yang kita gunakan sehari-hari untuk berbagai alasan. Dengan basis data, pengguna dapat menyimpan data secara terorganisasi. Setelah data disimpan, informasi hasur mudah diambil. Kriterisa dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basis data menentukan seberapa mudak mencari informasi berdasarkan banyak kriteria. Data pun harus mudah ditambahkan ke dalam basis data, dimodifikasi dan dihapus [8].

Menurut Fathansyah [6], basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.

3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronis.

II.2.3 *Database Management System* (DBMS)

"Managemen Sistem Basis Data (*Database Management System* /DBMS) adalah perangkat lunak yang di desain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar" [6].

Sistem Manajemen Basis data (*Database Management System*) merupakan sistem pengoperasian dan sejumlah data pada komputer. Dengan sistem ini dapat merubah data, memperbaiki data yang salah dan menghapus data yang tidak dapat dipakai. Sistem manajemen *database* merupakan suatu perluasan *software* sebelumnya mengenai *software* pada generasi komputer yang pertama. Dalam hal ini data dan informasi merupakan kesatuan yang saling berhubungan dan berkerja sama yang terdiri dari: peralatan, tenaga pelaksana dan prosedur data. Sehingga pengolahan data ini membentuk sistem pengolahan data. Peralatan dalam hal ini berupa perangakat keras (*hardware*) yang digunakan, dan prosedur data yaitu berupa perangakat lunak (*software*) yang digunakan dan dipakai untuk mengalokasikan dalam pembuatan sistem informasi pengolahan *database*.

Manipulasi basis data meliputi pembuatan pernyataan (*query*) untuk mendapatkan informasi tertentu, melakukan pembaharuan atau penggantian (*update*) data, serta pembuatan report dari data. Tujuan utama DBMS adalah untuk menyediakan tinjauan abstrak dari data bagi *user*. Jadi sistem menyembunyikan informasi mengenai bagaimana data disimpan dan dirawat, tetapi data tetap dapat diambil dengan efisien.

Pertimbangan efisiensi yang digunakan adalah bagaimana merancang struktur data yang kompleks, tetapi tetap dapat digunakan oleh pengguna yang masih awam, tanpa mengetahui kompleksitas stuktur data. Sistem manajemen *database* atau *database* management system (DBMS) adalah merupakan suatu sistem *software* yang memungkinkan seorang *user* dapat mendefinisikan, membuat, dan memelihara serta menyediakan akses terkontrol terhadap data. *Database* sendiri adalah sekumpulan data yang berhubungan dengan secara logika

dan memiliki beberapa arti yang saling berpautan. DBMS yang utuh biasanya terdiri dari :

1. *Hardware*

   Hardware merupakan sistem komputer aktual yang digunakan untuk menyimpan dan mengakses *database*. Dalam sebuah organisasi berskala besar, *hardware* terdiri : jaringan dengan sebuah server pusat dan beberapa program *client* yang berjalan di komputer desktop.

2. *Software*

   Software beserta utility *Software* adalah DBMS yang aktual. DBMS memungkinkan para *user* untuk berkomunikasi dengan *database*. Dengan kata lain DBMS merupakan mediator antara *database* dengan *user*. Sebuah *database* harus memuat seluruh data yang diperlukan oleh sebuah organisasi.

3. Prosedur

   Bagian integral dari setiap sistem adalah sekumpulan prosedur yang mengontrol jalannya sistem, yaitu praktik-praktik nyata yang harus diikuti *user* untuk mendapatkan, memasukkan, menjaga, dan mengambil data.

4. Data

   Data adalah jantung dari DBMS. Ada dua jenis data. Pertama, adalah kumpulan informasi yang diperlukan oleh suatu organisasi. Jenis data kedua adalah metadata, yaitu informasi mengenai *database*.

5. *User* (Pengguna)

   Ada sejumlah *user* yang dapat mengakses atau mengambil data sesuai dengan kebutuhan penggunaan aplikasi-aplikasi dan *interface* yang disediakan oleh DBMS, antara lain adalah :

   a. *Database* administrator adalah orang atau group yang bertanggungjawab mengimplementasikan sistem *database* di dalam suatu organisasi.
   b. *End user* adalah orang yang berada di depan workstation dan berinteraksi secara langsung dengan sistem.
   c. Programmer aplikasi, orang yang berinteraksi dengan *database* melalui cara yang berbeda.

II.2.4 *Data Mining*

Data mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam database [9].

Menurut Pramudiono [1], "*Data mining* adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual."

Menurut Larose [4] kemajuan luar biasa yang terus berlanjut dalam bidang data mining didorong oleh beberapa faktor, antara lain :

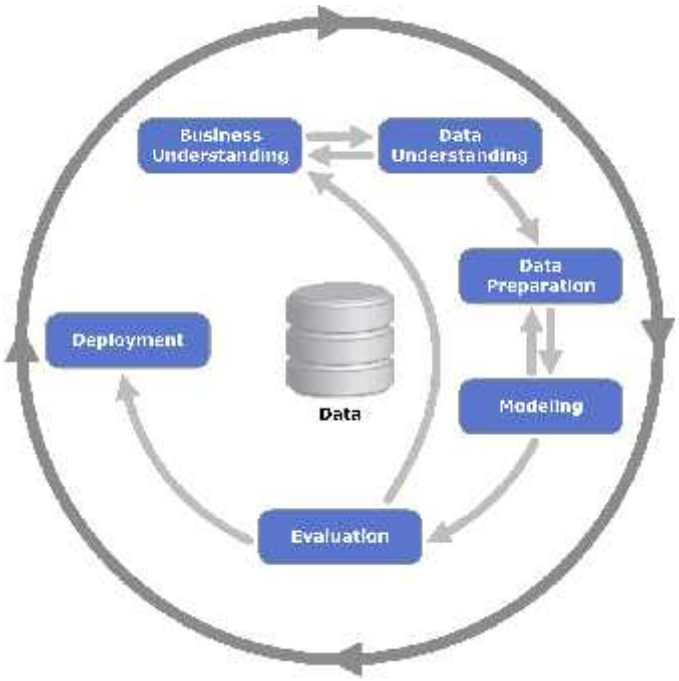
1. Pertumbuhan yang cepat dalam kumpulan data.
2. Penyimpanan data dalam *data warehouse*, sehingga seluruh perusahaan memiliki akses ke dalam *database* yang andal.
3. Adanya peningkatan akses data melalui navigasi *web* dan intranet.
4. Tekanan kompetensi bisnis untuk meningkatkan penguasaan pasar dalam globalisasi ekonomi.
5. Perkembangan teknologi perangkat lunak untuk *data mining* (ketersediaan teknologi).
6. Perkembangan yang hebat dalam kemampuan komputasi dan pengembangan kapasitas media penyimpanan.

II.2.4.1 *Cross-Industry Standard Process for Data Mining* (CRISP-DM)

CRISP-DM yang dikembangkan tahun 1996 oleh analis dari beberapa industri seperti Daimler Chrysler, SPSS dan NCR. CRISP-DM menyediakan standar proses *data mining* sebagai strategi pemecahan masalah secara umum dari bisnis atau unit penelitian.

Dalam CRISP-DM, sebuah proyek *data mining* memiliki siklus hidup yang terbagi dalam enam fase. Keseluruhan fase berurutan yang ada tersebut bersifat adaptif. Fase berikutnya dalam urutan bergantung kepada keluaran dari fase sebelumnya. Hubungan antarfase digambarkan dengan panah. Sebagai contoh, jika proses berada pada fase *modelling*. Berdasar pada perilaku dan karakteristik model, proses mungkin harus kembali kepada fase *data preparation* untuk perbaikan lebih lanjut terhadap data atau berpindah maju kepada fase

evaluation. Enam fase yang ada dalam metode CRISP-DM tersebut dapat dilihat pada Gambar II.3 :



Gambar II.3 CRISP-DM

Fase-fase CRISP-DM [4] :

1. *Business Understanding*

   Business Understanding atau pemahaman domain (penelitian). Pada fase ini dibutuhkan pemahaman tentang substansi dari kegiatan *data mining* yang akan dilakukan, kebutuhan dari perspektif bisnis. Kegiatannya antara lain: menentukan sasaran atau tujuan bisnis, memahami situasi bisnis, menentukan tujuan *data mining* dan membuat perencanaan strategi serta jadwal penelitian.

2. *Data Understanding*

   Data Understanding atau pemahaman data adalah fase mengumpulkan data awal, mempelajari data untuk bisa mengenal data yang akan dipakai. Fase ini mencoba mengidentifikasikan masalah yang berkaitan dengan kualitas data, mendeteksi subset yang menarik dari data untuk membuat hipotesa awal.

3. *Data Preparation*

   Data preparation atau persiapan data. Fase ini sering disebut sebagai fase yang padat karya. Aktivitas yang dilakukan antara lain memilih *table* dan *field* yang akan ditransformasikan ke dalam database baru untuk bahan *data mining* (set data mentah).

4. *Modelling*

   Modeling adalah fase menentukan tehnik *data mining* yang digunakan, menentukan *tools data mining*, teknik *data mining*, algoritma *data mining*, menentukan parameter dengan nilai yang optimal.

5. *Evaluation*

   Evaluation adalah fase interpretasi terhadap hasil *data mining* yang ditunjukan dalam proses pemodelan pada fase sebelumnya. Evaluasi dilakukan secara mendalam dengan tujuan menyesuaikan model yang didapat agar sesuai dengan sasaran yang ingin dicapai dalam fase pertama.

6. *Deployment*

   Deployment atau penyebaran adalah fase penyusunan laporan atau presentasi dari pengetahuan yang didapat dari evaluasi pada proses data mining.

II.2.4.2 Metode-Metode Data Mining

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu [10] :

1. Deskripsi

   Terkadang peneliti dan analis secara sederhana ingin mencoba mencari cara untuk menggambarkan pola dan kecenderungan yang terdapat dalam data. Sebagai contoh, petugas pengumpulan suara mungkin tidak dapat mengumpulkan keterangan atau fakta bahwa siapa yang tidak cukup profesional akan sedikit didukung dalam pemilihan presiden. deskripsi dari pola dan kecenderungan sering memberikan kemungkinan penjelesan untuk suatu pola atau kecenderungan.

2. Estimasi

   Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih kearah numerik daripada ke arah kategori. Model dibangun dengan record lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi. Sebagai contoh, akan dilakukan estimasi tekanan darah sistolik pada pasien rumah sakit berdasarkan umur pasien, jenis kelamin, indeks berat badan, dan level sodium darah. Hubungan antara tekanan darah sistolik dan nilai variabel prediksi dalam proses pembelajaran akan menghasilkan model estimasi. Model estimasi yang dihasilkan dapat digunakan untuk kasus baru lainnya.

3. Prediksi

   Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada dimasa mendatang. Contoh prediksi dalam bisnis dan penelitian adalah :

   1. Prediksi harga beras dalam tiga bulan yang akan datang.
   2. Prediksi persentase kenaikan kecelakaan lalu lintas tahun depan jika batas bawah dinaikan.

4. Klasifikasi

   Dalam klasifikasi, terdapat terget variabel kategori. sebagai contoh, penggolongan pendapatan dapat dipisahkan dalam tiga kategori yaitu: pendapatan tinggi, pendapatan sedang, dan pendapatan rendah. Contoh lain klasifikasi dalam bisnis dan penelitian adalah :

   1. Menentukan apakah suatu transaksi kartu kredit merupakan transaksi yang curang atau bukan.
   2. Mendiagnosis penyakit seorang pasien untuk mendapatkan termasuk kategori penyakit apa.

5. Pengklusteran

   Pengklusteran merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan. Kluster adalah kumpulan record yang memiliki kemiripan satu dengan yang lainnya dan tidak memiliki kemiripan dengan record-record dalam kluster lain.

Pengklusteran berbeda dengan klasifikasi yaitu tidak adanya variabel target dalam pengklusteran. pengklusteran tidak mencoba untuk melakukan klasifikasi, mengestimasi, atau memprediksi nilai dari variabel target. Akan tetapi, algoritma pengklusteran mencoba untuk melakukan pembagian terhadap keselurahan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), yang mana kemiripan record dalam suatu kelompok akan bernilai maksimal, sedangkan kemiripan dengan record dalam kelompok lain akan bernilai minimal. Contoh pengklusteran dalam bisnis dan penelitian adalah :

1. Mendapatkan kelompok-kelompok konsumen untuk target pemasaran dari suatu produk sebuah perusahaan yang tidak memiliki dana pemasaran yang besar.
2. Untuk tujuan audit akuntansi, yaitu melakukan pemisahan terhadap perilaku finansial dalam baik maupun mencurigakan.

6. Asosiasi

Tugas asosiasi dalam data mining adalah menemukan atribut yang muncul dalam satu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang pasar. Contoh asosiasi dalam bisnis dan penelitian adalah :

1. Meneliti jumlah pelanggan dari perusahaan telekomunikasi seluler yang diharapkan untuk memberikan respon positif terhadap penawaran upgrade layanan yang diberikan.
2. Menemukan barang dalam supermarket yang dibeli secara bersamaan dan barang yang tidak pernah dibeli secara bersamaan.

II.2.5 Metode *Association Rule*

Association rule adalah teknik *data mining* untuk mencari hubungan antar-*item* dalam suatu dataset yang ditentukan. Contoh aturan asosiatif bisa kita ambil dari suatu transaksi penjualan di sebuah toko, kita dapat mengetahui berapa besar kemungkinan seorang konsumen membeli suatu *item* bersamaan dengan *item* lainnya. Metode ini sering juga dinamakan dengan *market basket analysis* karena awal mulanya yang berasal dari studi tentang database transaksi penjualan.

Bila kita ambil contoh dalam sebuah transaksi pembelian barang di sebuah toko didapat bentuk *association rule* roti → selai. Yang berarti bahwa konsumen yang membeli roti ada kemungkinan konsumen tersebut juga akan membeli selai.

Association rule memiliki dua tahap pengerjaan, yaitu [11]:

1. Mencari kombinasi yang paling sering terjadi dari suatu itemset.
2. Mendefinisikan Condition dan Result (untuk conditional association rule).

Dalam menentukan suatu *association rule*, terdapat suatu *interestingness measure* (ukuran kepercayaan) yang didapat dari hasil pengolahan data dengan perhitungan tertentu. Umumnya ada dua ukuran, yaitu :

1. *Support* : suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu *item*/*itemset* dari keseluruhan transaksi. Ukuran ini menentukan apakah suatu *item*/*itemset* layak untuk dicari *confidence*-nya (misal, dari keseluruhan transaksi yang ada, seberapa besar tingkat dominasi suatu item yang menunjukkan bahwa *item* A dan *item* B dibeli bersamaan).
2. *Confidence* : suatu ukuran yang menunjukkan hubungan antara 2 *item* secara *conditional* (misal, menghitung kemungkinan seberapa sering *item* B dibeli oleh pelanggan jika pelanggan tersebut membeli sebuah *item* A).

Kedua ukuran ini nantinya berguna dalam menentukan kekuatan suatu pola dengan membandingkan pola tersebut dengan nilai minimum kedua parameter tersebut yang ditentukan oleh pengguna. Bila suatu pola memenuhi kedua nilai minimum parameter yang sudah ditentukan sebelumnya, maka pola tersebut dapat disebut sebagai *interesting rule* atau *strong rule*.

Metodologi dasar analisis asosiasi terbagi menjadi dua tahap [9] :

1. Analisis pola frekuensi tertinggi

   Tahap ini mencari kombinasi *item* yang memenuhi syarat minimum dari nilai *support* dalam *database*. Nilai *support* sebuah *item* diperoleh dengan rumus berikut.

$$Support(A) = \frac{jumlah\ transaksi\ yang\ mengandung\ A}{jumlah\ total transaksi} \quad \text{Persamaan (II-1)}$$

Sementara itu, nilai *support* dari 2 *item* diperoleh dari rumus berikut.

$$Support(A,B) = P(A \cap B)$$ Persamaan (II-2)

$$Support(A,B) = \frac{Jumlah\ transaksi\ yang\ mengandung\ A\ dan\ B}{Jumlah\ total\ transaksi}$$ Persamaan (II-3)

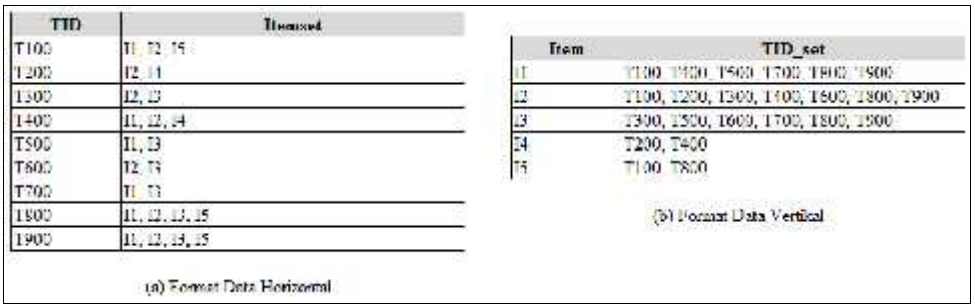
2. Pembentukan aturan asosiasi

Setelah semua pola frekuensi tinggi ditemukan, kemudian mencari aturan asosiatif yang memenuhi syarat minimum untuk *confidence* dengan menghitung *confidence* aturan assosiatif A -> B dari *support* pola frekuensi tinggi A dan B, menggunakan rumus :

$$Confidence = P(B \mid A) = \frac{Jumlah\ transaksi\ yang\ mengandung\ A\ dan\ B}{Jumlah\ transaksi\ yang\ mengandung\ A} x\ 100\%$$ Persamaan (II-4)

II.2.6 Algoritma *Equivalence Class Transformation* (Eclat) [12]

Baik Apriori dan FP-growth keduanya merupakan metode *frequent mining* dari kumpulan transaksi dengan menggunakan bentuk TID-*itemset*, dimana TID merupakan ID transaksi dan *itemset* merupakan kumpulan dari barang yang dibeli pada transaksi TID. Ini dikenal dengan format data horizontal. Selain itu, data dapat direpresentasikan dalam bentuk item-TID_set, dimana item merupakan nama item, dan TID_set adalah kumpulan ID transaksi yang mengandung item. Ini dikenal dengan format data vertikal. Algoritma Eclat adalah algoritma yang menggunakan format data vertikal untuk merepresentasikan datanya.



TID	Itemset
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3, I5

(a) Format Data Horizontal

Item	TID_set
I1	T100, T400, T500, T700, T800, T900
I2	T100, T200, T300, T400, T600, T800, T900
I3	T300, T500, T600, T700, T800, T900
I4	T200, T400
I5	T100, T800

(b) Format Data Vertical

Gambar II.4 Format Data Horizontal dan Vertikal

Berikut ini merupakan ilustrasi proses *mining frequent itemset* dengan menggunakan algoritma Eclat :

1. Representasikan data ke dalam bentuk format data vertikal.
2. Tentukan nilai *minimum support*.
3. Tentukan nilai *minimum confidence*.
4. Cari data kandidat k-*itemset* (C_k) lalu hitung masing-masing nilai *support*-nya.
5. Hilangkan *itemset* yang nilai *support*-nya kurang dari nilai *minimum support*.
6. Setelah mendapatkan data yang memenuhi nilai *minimum support* (*frequent*), lakukan kombinasi pada itemset berdasarkan TID_set yang sama pada data-data tersebut sehingga menciptakan k-*itemset*.
7. Ulangi langkah ke 4 sampai dengan langkah ke 6 sampai tidak menghasilkan *itemset* baru.
8. Cari aturan asosiatifnya dengan cara menentukan nilai *confidence*-nya,
9. Hilangkan data yang nilai *confidence*-nya kurang dari nilai *minimum confidence*.

Kelebihan yang dimiliki oleh algoritma Eclat yang digunakan pada penelitian ini adalah tidak memerlukan perhitungan nilai support, karena nilai support dari itemset telah direpresentasikan oleh TID_set dari itemset tersebut. Pekerjaan utama dari Eclat adalah mengkombinasikan item berdasarkan TID_set-nya, sehingga ukuran dari TID_set merupakan salah satu faktor utama yang mempengaruhi jumlah waktu dan memori yang diperlukan. Semakin besar TID_set maka semakin besar pula waktu dan memori yang diperlukan.

II.2.7 Alat-Alat Pemodelan Sistem

Alat-alat pemodelan sistem adalah alat dan metode yang akan digunakan untuk memodelkan perancangan perangkat lunak dalam skripsi ini. Adapun tools pemodelan yang digunakan adalah :

1. *Entity Relationship Diagram* (ERD)
2. Diagram Konteks
3. *Data Flow Diagram* (DFD)
4. Kamus Data

II.2.7.1 *Entity Relationship Diagram* (ERD)

ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak [13]. Basis data relasional adalah kumpulan dari relasi-relasi yang mengandung seluruh informasi berkenaan suatu entitas/ objek yang akan disimpan di dalam database. Tiap relasi disimpan sebagai sebuah *file* tersendiri. Perancangan basis data merupakan suatu kegiaatan yang setidaknya bertujuan sebagai berikut:

1. Menghilangkan redundansi data.
2. Meminimumkan jumlah relasi di dalam basis data.
3. Membuat relasi berada
4. Dalam bentuk normal, sehingga dapat meminimumkan permasalahan berkenaan dengan penambahan, pembaharuan dan penghapusan.

II.2.7.2 Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh *input* ke sistem dan *output* dari sistem. Ia akan membuat gambaran tentang keseluruhan sistem. Sistem dibatasi oleh *boundary* (dapat digambarkan dengan garis putus). Dalam diagram konteks hanya ada satu proses. Tidak boleh ada *store* dalam diagram konteks [13].

II.2.7.3 *Data Flow Diagram* (DFD)

Diagram aliran data merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan diagram aliran data adalah memudahkan pemakai atau *user* yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan [13].

II.2.7.4 Kamus Data

Kamus data berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan dalam sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai

dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses [13].

II.2.8 MySQL

MySQL merupakan salah satu contoh produk RDBMS yang sangat populer di lingkungan Linux, tetapi juga tersedia pada Windows. Banyak situs 19 *web* yang menggunakan MySQL sebagai *database server* (*server* yang melayani permintaan akses terhadap *database*). MySQL sebagai *database server* juga dapat diakses melalui program yang dibuat dengan menggunakan Borland Delphi. Dengan cara seperti ini *database* dapat diakses secara langsung melalui program *executable* yang dibuat sendiri [14].