BAB II
TINJAUAN PUSTAKA

II.1 Profil Instansi

PT. Celebes Lima Bersaudara yang berlokasi di Jalan Raya Cibeber Kilometer Tujuh Kampung Kubang Cilaku Cianjur, bergerak dalam bidang penjualan barang. Barang-barang yang mereka jual terbagi kedalam tiga kategori, yaitu sembako, *food*, *non-food*.

II.1.1 Sejarah Instansi

Celebes pertama kali berdiri pada tanggal 30 September 2001. Celebes ini pada awalnya hanya menjual sembako saja mulai dari beras, minyak, gula dan tepung. Pada April 2005, Celebes untuk pertama kalinya menjual barang-barang jenis *food* dan *non-food*. Barang yang pertama kali dijual dalam kategori tersebut adalah Susu Bendera dan Minuman Panther untuk kategori *food* serta Sabun Cusson dan Purbasari untuk kategori *non-food*. Agustus 2007 Celebes resmi menjadi badan usaha yang terdaftar dengan nama PT. Celebes Lima Bersaudara. Dan dari situlah Celebes mulai berkembang dan menjadi perusahaan distributor barang seperti saat ini.

II.1.2 Visi dan Misi Instansi

Visi yang dijalankan perusahaan adalah menjadi perusahaan distributor lokal yang berkedudukan dan berkepanjangan. Misi perusahaan ada 2 jenis yaitu :

1. Produktif

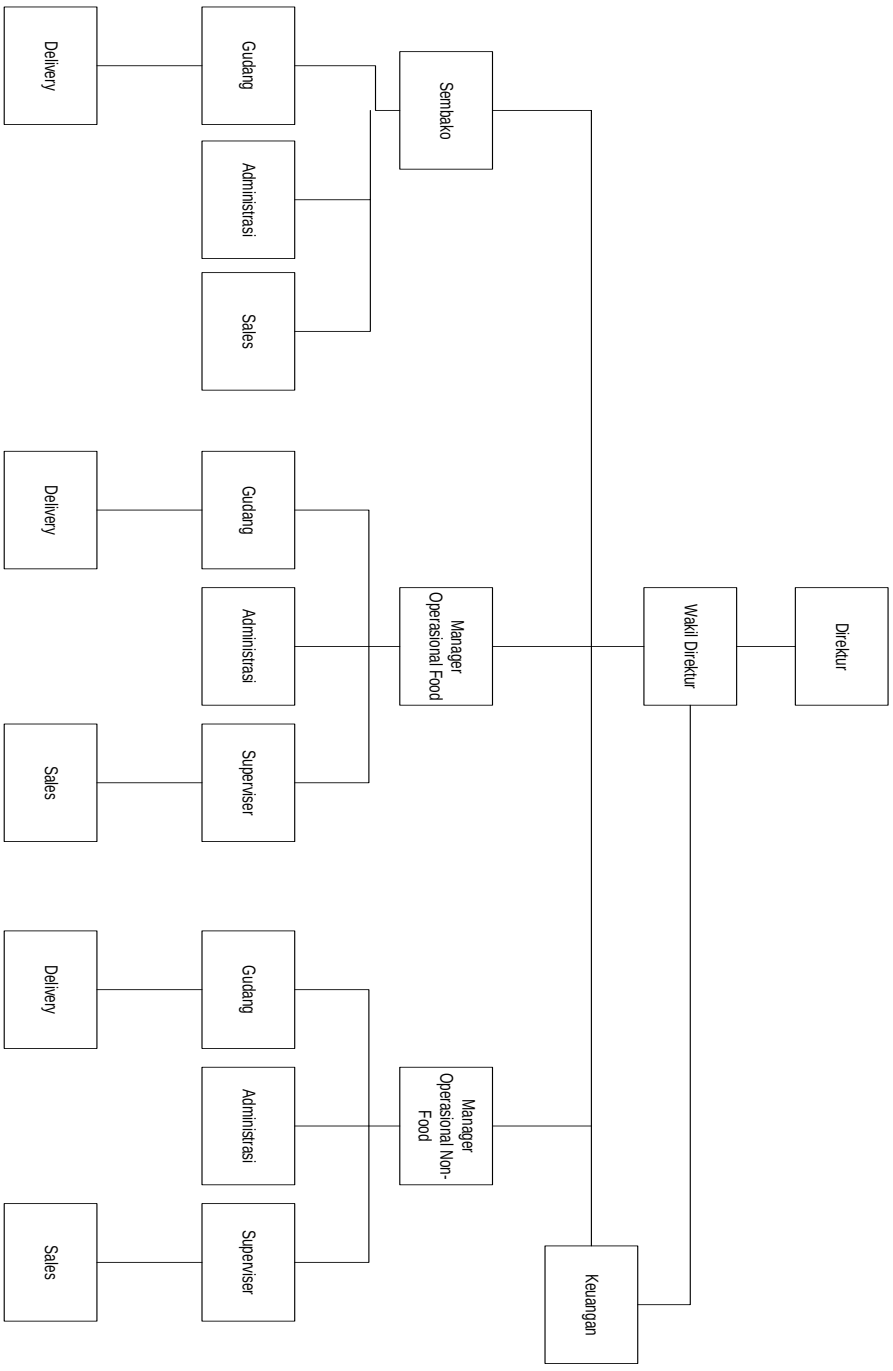
   Meningkatkan produktifitas perusahaan dan sumber daya manusianya agar perusahaan dapat terus berkembang kearah yang lebih baik.

2. Efisien

   Perusahaan harus selalu efisian dalam setiap mengerjakan kegiatannya agar konsumen dapat terlayani sepuasnya.

II.1.3 Struktur Organisasi

Struktur organisasi instansi adalah gambar bagan yang menerangkan posisi dan hierarki struktur kerja pegawai di dalam suatu perusahaan. Struktur organisasi PT. Celebes Lima Bersaudara dapat dilihat pada



Gambar II-1 Struktur Organisasi PT. Celebes Lima Bersaudara

II.2 Landasan Teori

Landasan teori membahas mengenai materi atau teori apa saja yang digunakan sebagai acuan dalam membuat tugas akhir ini. Landasan teori yang di uraikan merupakan hasil studi literatur, baik dari 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 [4]. 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* [5].

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 harus mudah diambil. Kriteria dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basis data menentukan seberapa mudah mencari informasi berdasarkan banyak kriteria. Data pun harus mudah ditambahkan ke dalam basis data, dimodifikasi, dan dihapus [6].

Menurut Fathansyah [4], 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

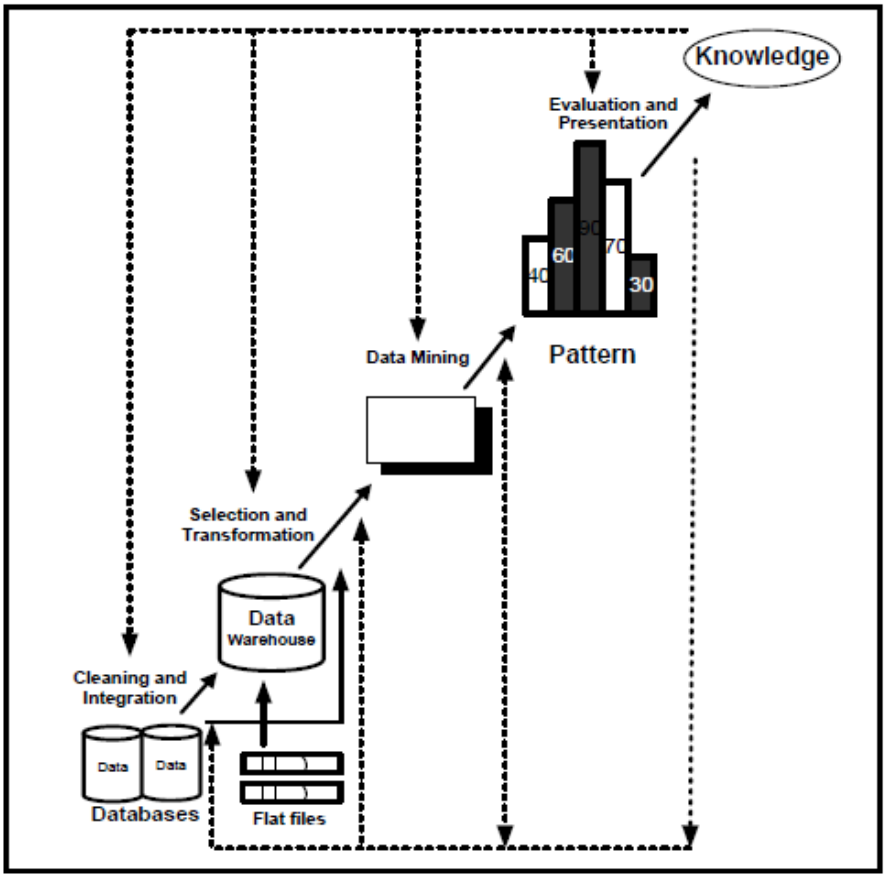
Kumpulan atau gabungan database dengan perangkat lunak aplikasi yang berbasis database tersebut dinamakan *Database Management System* (DBMS). DBMS merupakan koleksi terpadu dari database dan program–program komputer (utilitas) yang digunakan untuk mengakses dan memelihara database. Program-program tersebut menyediakan berbagai fasilitas operasi untuk memasukan, melacak, dan memodifikasi data kedalam database, mendefinisikan data baru, serta mengolah data menjadi informasi yang dibutuhkan (DBMS = *Database* + *Program Utilitas*) [4].

II.2.4 Data Mining

Data mining, sering juga disebut *knowledge discovery in database* (KDD), adalah kegiatan yang meliputi pengumpulan, pemakaian data historis untuk menemukan keteraturan, pola atau hubungan dalam set data berukuran besar. Keluaran dari data mining ini bisa dipakai untuk memperbaiki pengambilan keputusan di masa depan. Sehingga istilah *pattern recognition* sekarang jarang digunakan karena ia termasuk bagian dari *data mining* [1].

II.2.4.1 Tahapan – Tahapan Data Mining

Data mining adalah sebagian dari rangkaian proses, tahapan-tahapan proses data mining tersebut dapat dilihat pada Gambar II-2.



Gambar II-2 Tahapan proses data mining

Tahapan-tahapan data mining :

1. Pembersihan data (*data cleaning*)

   Pembersihan data merupakan proses menghilangkan noise dan data yang tidak konsisten. Pembersihan data ini juga akan mempengaruhi performasi dari teknik *data mining* karena data yang ditangani akan berkurang jumlah dan kompleksitasnya.

2. Integrasi data (*data integration*)

   Integrasi data merupakan penggabungan data dari berbagai *database* ke dalam satu *database* baru. Tidak jarang data yang diperlukan untuk *data mining* tidak hanya berasal dari satu *database* saja, tetapi bisa juga berasal dari beberapa *database* atau *file* teks. Integrasi data perlu dilakukan secara teliti dan cermat karena kesalahan pada integrasi data bisa menghasilkan hasil yang menyimpang dan dapat menyesatkan proses pengambilan aksi nantinya.

3. Seleski data (*data selection*)

   Data yang ada pada suatu *database* acap kali tidak semua terpakai, oleh karena itu hanya data yang sesuai kebutuhan saja yang akan dipakai. Sebagai contoh, dalam mencari nilai asosiasi suatu barang, data nama pelanggan dalam data transaksi tidak diperlukan, karena data nama pelanggan tersebut tidak diperlukan, data tersebut dapat dihilangkan.

4. Transformasi data (*data transformation*)

   Data diubah atau digabung ke dalam format yang sesuai untuk diproses dalam data mining. Beberapa metode data mining membutuhkan suatu format data khusus sebelum bisa diaplikasikan.

5. Proses mining

   Proses utama dalam tahapan data mining, disaat inilah metode-metode yang digunakan diterapkan untuk menemukan pengetahuan berharga dan tersembunyi dari data.

6. Evaluasi pola (*pattern evaluation*)

   Untuk mengidentifikasi pola-pola menarik kedalam knowledge based yang ditemukan. Dalam tahap ini hasil dari teknik data mining berupa pola-pola yang khas maupun model prediksi dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai. Bila ternyata hasil yang didapat tidak sesuai, maka ada beberapa *alternative* yang dapat diambil, seperti menjadikannya umpan balik untuk memperbaiki proses data mining atau mencoba metode *data mining* lainnya yang lebih sesuai.

7. Presentasi pengetahuan (*knowledge presentation*)

   Tahap terakhir dari proses data mining ini adalah bagaimana memformulasikan keputusan atau aksi dari hasil data mining yang didapat sehingga dapat dipahami oleh semua pihak.

II.2.4.2 Metode - Metode Data Mining

Secara garis besar, Han dalam bukunya menjelaskan bahwa metode *data mining* dapat dilihat dari dua sudut pandang pendekatan yang berbeda, yaitu pendekatang deskriptif dan pendekatan prediktif [7].

Pendekatan deskriptif adalah pendekatan dengan cara mendeskripsikan data inputan. Metode yang termasuk kedalam pendekatan ini adalah :

1. Metode deskripsi konsep/kelas, yaitu data dapat diasosiasikan dengan kelas atau konsep. Ada tiga macam pendeskripsian yaitu (1) karakteristik data, dengan membuat summary karakter umum atau fitur data suatu kelas target, (2) diskriminasi data, dengan membandingkan class target dengan satu atau sekelompok kelas pembanding, (3) gabungkan antara karakterisasi dan diskriminasi.
2. Metode *association rule*, yaitu menemukan aturan asosiatif atau pola kombinasi dari suatu item yang sering terjadi dalam sebuah data.

Pendekatan kedua adalah pendekatan prediktif, yaitu pendekatan yang dapat digunakan untuk memprediksi, dengan hasil berupa kelas atau *cluster*. Metode yang termasuk dalam pendekatan ini adalah :

1. Metode klasifikasi dan prediksi, yaitu metode analisis data yang digunakan untuk membentuk model yang mendeskripsikan kelas data yang penting, atau model yang memprediksikan trend data. Klasifikasi digunakan untuk memprediksi kelas data yang bersifat kategorial, sedangkan prediksi untuk memodelkan fungsi yang mempunyai nilai kontinu.
2. Metode *clustering*, mengelompokan data untuk membentuk kelas-kelas baru atau sering disebut *cluster*. Metode *clustering* bertujuan untuk memaksimalkan persamaan dalam satu *cluster* dan meminimalkan perbedaan antar *cluster*.

Dalam metode-metode *data mining* diketahui bahwa metode *association rule* digunakan untuk menemukan aturan asosiatif atau pola kombinasi dari suatu *item*. Bila metode *association rule* ini diterapkan dalam sebuah data transaksi penjualan, metode ini dapat menghasilkan informasi berupa pola kebiasaan

membeli pelanggan. Sebagai contoh, dari suatu data transaksi penjualan didapat bahwa seorang pelanggan bila membeli roti, maka pelanggan tersebut akan membeli juga selai kacang dalam satu transaksi yang sama. Dari contoh tersebut bisa diambil kesimpulan bahwa metode *association rule* cocok untuk diterapkan dalam penelitian ini, karena pola kebiasaan membeli pelanggan bisa menjadi acuan dalam menentukan barang apa saja yang akan dijual sebagai paket penjualan barang.

II.2.5 Metode Association Rule

Association Rule atau Aturan Asosiasi adalah teknik *Data Mining* untuk menemukan aturan asosiatif atau pola kombinasi dari suatu item. Bila kita mengambil contoh aturan asosiatif dalam suatu transaksi pembelian barang di suatu *minimarket* adalah kita dapat mengetahui berapa besar kemungkinan seorang konsumen membeli suatu *item* bersamaan dengan *item* lainnya (membeli roti bersama dengan selai). Karena awalnya berasal dari studi tentang *database* transaksi pelanggan untuk menentukan kebiasaan suatu produk dibeli bersama apa, maka *association rule* sering juga dinamakan *market basket analysis* [1].

Association Rule adalah bentuk jika "kejadian sebelumnya" kemudian "konsekuensinya" (*If antecedent, then consequent*), yang diikuti dengan perhitungan aturan *support* dan *confidence*. Bentuk umum dari *association rule* adalah *Antecedent -> Consequent*.

Bila kita ambil contoh dalam sebuah transaksi pembelian barang di sebuah *minimarket* didapat bentuk *association rule* roti -> selai. Yang artinya bahwa pelanggan yang membeli roti ada kemungkinan pelanggan tersebut juga akan membeli selai, dimana tidak ada batasan dalam jumlah *item-item* pada bagian *antecedent* ataupun *consequent* dalam sebuah rule.

Association rule memiliki dua tahap pengerjaan, yaitu [8]:

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*.

II.2.5.1 Metodologi Dasar Analisis Asosiasi

Metodologi dasar *Association Rule* terbagi menjadi dua tahap, yaitu [9]:

a. Analisa pola frekuensi tinggi

   Tahap ini mencari pola *item* yang memenuhi syarat minimum dari nilai *support* dalam *database*. Menurut Larose, kita bebas menentukan nilai *minimum support* (minsup) dan *minimum confidence* (mincof) sesuai kebutuhan [10]. Sebagai contoh, bila ingin menemukan data-data yang memiliki hubungan asosiasi yang kuat, minsup dan mincof-nya bisa diberi nilai yang tinggi. Sebaliknya, bila ingin melihat banyaknya variasi data tanpa terlalu mempedulikan kuat atau tidaknya hubungan asosiasi antara *item*-nya, nilai minsup dan mincofnya dapat diisi rendah. Nilai *support* sebuah *item* diperoleh dengan rumus :

$$\text{Support}(A) = \frac{\text{Jumlah Transaksi Yang Mengandung A}}{\text{Jumlah Total Transaksi}} \quad \text{(II-1)}$$

Persamaan 1 menjelaskan bahwa nilai support didapat dengan cara membagi jumlah transaksi yang mengandung item A (satu item) dengan jumlah total seluruh transaksi. Sedangkan untuk mencari nilai support dari 2 item menggunakan rumus berikut :

$$\text{Support}(A,B) = P(A \cap B) = \frac{\text{Jumlah Transaksi Yang Mengandung Item A dan B}}{\text{Jumlah Total Transaksi}} \quad \text{(II-2)}$$

Persamaan 2 menjelaskan bahwa nilai support 2-*itemsets* didapat dengan cara membagi jumlah transaksi yang mengandung *item* A dan *item* B (*item* pertama bersamaan dengan *item* yang lain) dengan jumlah total seluruh transaksi.

b. Pembentukan Aturan Assosiatif

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 :

$$\text{Confidence} = (A \rightarrow B) = \frac{\text{Jumlah Transaksi Yang Mengandung A dan B}}{\text{Jumlah Transaksi Yang Mengandung A}} \quad \text{(II-3)}$$

Persamaan 3 menjelaskan bahwa nilai *confidenc*e diperoleh dengan cara membagi jumlah transaksi yang mengandung *item* A dan *item* B (*item* pertama bersamaan dengan *item* yang lain) dengan jumlah transaksi yang mengandung *item* A (*item* Pertama atau *item* yang ada disebelah kiri).

II.2.5.2 Lift/Improvement Ratio

Lift Ratio adalah parameter penting selain *support* dan *confidence* dalam *association rule*. *Lift ratio* mengukur seberapa penting rule yang telah terbentuk berdasarkan nilai *support* dan *confidence*. Lift Ratio merupakan nilai yang menunjukkan kevalidan proses transaksi dan memberikan informasi apakah benar item A dibeli bersamaan dengan item B [1].

Lift/Improvement Ratio dapat dihitung dengan rumus :

$$\frac{\text{Support}\,(A \cap B)}{\text{Support}(A) * \text{Support}(B)} \qquad \text{(II-4)}$$

Sebuah transaksi dikatakan valid jika mempunyai nilai *Lift/Improvement* lebih dari 1, yang berarti bahwa dalam transaksi tersebut item A dan item B benar-benar dibeli secara bersamaan.

II.2.5.3 Algoritma Apriori

Algoritma Apriori adalah suatu algoritma yang sudah sangat dikenal dalam melakukan pencarian *frequent itemset* dengan *association rule*. Sesuai dengan namanya, algoritma ini menggunakan pengetahuan mengenai *frequent itemset* yang telah diketahui sebelumnya untuk memproses informasi selanjutnya.

Algoritma inilah yang biasanya dipakai dalam proses data *mining* untuk *market basket analysis*. Algoritma apriori memakai pendekatan *iterative* (*level-wise search*), dimana k-*itemset* dipakai untuk menyelidiki (k+1)-*itemset*. Langkah-langkah dari algoritma ini adalah sebagai berikut [8]:

1. Set k=1 (menunjuk pada *itemset* ke-1).
2. Hitung semua k-*itemset* (*itemset* yang mempunyai k *item*).
3. Hitung *support* dari semua calon *itemset*. *Filter itemset* tersebut berdasarkan nilai *minimum support*-nya.
4. Gabungkan semua k-*sized itemset* untuk menghasilkan calon *itemset* k+1.
5. Set k=k+1 (menunjuk pada *itemset* ke-2 dan seterusnya)
6. Ulangi langkah 3-5 sampai tidak ada *itemset* yang lebih besar yang dapat dibentuk.
7. Buat final *set* dari *itemset* dengan menciptakan suatu *union* dari semua k-*itemset*.

II.2.6 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. Diagram Konteks
2. Data Flow Diagram (DFD)
3. Kamus Data

II.2.6.1 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 [11].

II.2.6.2 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 [11].

II.2.6.3 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 [11].

II.2.7 Alat-alat Pembangunan Perangkat Lunak

Alat-alat pembangunan perangkat lunak adalah alat-alat atau *tools* yang akan digunakan untuk mengimplementasikan perangkat lunak yang akan dibangun. Adapun *tools* yang akan digunakan adalah MySQL.

II.2.7.1 MySQL

MySQL merupakan salah satu contoh produk RDBMS yang sangat populer di lingkungan Linux, tetapi juga tersedia pada Windows. Banyak situs

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 [12].