BAB 2

TINJAUAN PUSTAKA

2.1 Profil Institusi

2.1.1 Sejarah RM. Roso Echo

Rumah makan Roso Echo merupakan rumah makan yang pertama dan satu-satunya rumah makan di Cirebon yang menyediakan aneka masakan jawa dengan kualitas rasa khas masakan jawa. Rumah makan Roso Echo ini pertama kali di dirikan di Jalan Pangeran Cakrabuana BTN Cempaka Arum A-3, Sumber-Cirebon, pada tanggal 1 September 2006 oleh Ir. Suwarno. Selain menyediakan masakan khas jawa, rumah makan Roso Echo ini menerapkan konsep prasmanan dengan nuansa khas jawa dalam pelayanannya, sehingga memberikan kemudahan bagi para konsumen untuk melihat dan memilih secara langsung menu makanan yang sesuai dengan keinginan dan kebutuhannya tanpa harus menunggu menu makanan disajikan. Selain itu, konsumen juga dapat menikmati nuansa jawa yang belum pernah konsumen dapatkan di rumah makan lain di Cirebon. Hal inilah yang menjadikan rumah makan ini sangat diminati oleh para konsumen dari berbagai daerah yang ada di wilayah Cirebon.

Seiring dengan minat konsumen terhadap rumah makan Roso Echo yang semakin meningkat setiap tahunnya, sehingga pada tahun 2010 tepatnya tanggal 1 Juli 2010, rumah makan Roso Echo ini resmi membuka cabang yang berlokasi tak jauh dari pusatnya, yaitu di Jalan Sultan Ageng Tirtayasa No. 4 (Selatan Brimob), Sumber-Cirebon. Kini, cabang rumah makan Roso Echo yang didirikan pada tahun 2010 pun telah mengalami perkembangan yang semakin pesat. Dahulu rumah makan ini hanya memiliki satu ruangan bagi konsumen untuk menikmati makananannya, dan kini rumah makan ini telah memiliki beberapa ruangan yang dapat konsumen gunakan.

2.1.2 Logo

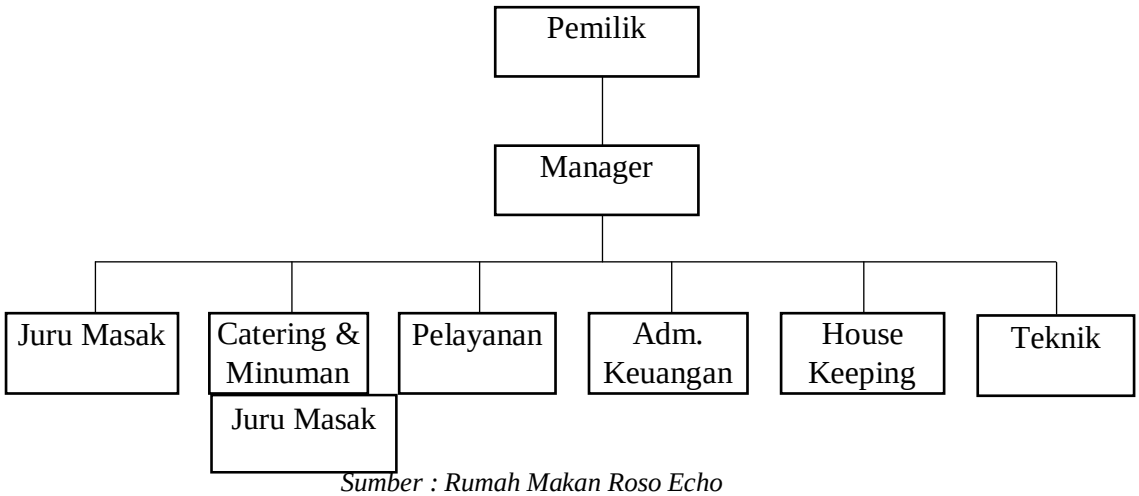
Logo Rumah Makan Roso Echo



Gambar 2.1 Logo RM. Roso Echo

Sumber : Rumah Makan Roso Echo

2.1.3 Struktur Organisasi



Sumber : Rumah Makan Roso Echo

Gambar 2.2 Struktur Organisasi RM. Roso Echo

Job description :

1. Pemilik bertanggung jawab penuh terhadap RM. Makan Roso Echo. Tugas dan tanggung jawab :
   a. Membuat perencanaan, strategi dan kebijakan yang menyangkut operasi rumah makan.

b. Menyusun anggaran kebutuhan persediaan bahan baku.
c. Melakukan kontrol secara keseluruhan atas operasi rumah makan.
d. Memegang kendali atas keputusan penting yang bersifat umum berkaitan dengan keuangan.

2. Manager berfungsi untuk membantu pemilik. Tugas dan tanggung jawab :
   a. Membantu pemilik dalam mengawasi rumah makan.
   b. Mengatur setiap bagian yang ada di rumah makan agar menjalankan tugasnya dengan baik.
3. Bagian juru masak memiliki tugas dan tanggung jawab :
   a. Memasakan semua menu yang harus di sediakan di meja prasmanan.
   b. Bertanggung jawab atas pesanan pelanggan.
   c. Bertanggung jawab atas hasil masakan.
   d. Mengontrol persediaan bahan baku.
4. Bagian catering&minuman memiliki tugas dan tanggung jawab :
   a. Menyiapkan catering untuk pesanan pelanggan.
   b. Mengantarkan catering kepada pelanggan.
   c. Menyiapkan minuman yang dipesan setiap pelanggan.
   d. Mengontrol persediaan bahan baku minuman.
5. Bagian pelayananan berfungsi untuk melayani pelanggan. Tugas dan tanggung jawab:
   a. Membantu dalam melayani pesanan-pesanan pelanggan.
   b. Bertanggung jawab atas kebutuhan piring, sendok, gelas, menu makanan, dan sebagainya.
   c. Melayani semua pesanan pelanggan.
6. Bagian adm keuangan bertugas terhadap transaksi pelanggan. Tugas dan tanggung jawab :
   a. Bertanggung jawab terhadap hal-hal yang menyangkut keuangan.
   b. Menghitung pemasukan dan pengeluaran setiap bulannya.
   c. Melayani pelanggan dalam pembayaran transaksi.
7. Bagian *house keeping* bertugas terhadap kebersihan dan kenyamanan pelanggan. Tugas dan tanggung jawab :

a. Bertanggung jawab terhadap kebersihan, baik di luar ataupun didalam lingkungan rumah makan.
b. Membereskan dan membersihkan peralatan makan yang telah pelanggan gunakan.

8. Bagian teknik bertugas dan bertanggung jawab terhadap semua keperluan teknisi yang ada di rumah makan.

2.1.4 Visi dan Misi

2.1.4.1 Visi Roso Echo

Visi RM. Roso Echo adalah :

1. Mendirikan rumah makan dengan suasana konsep alam.
2. Mewujudkan makanan berkualitas, dengan cita rasa makanan tradisional.

2.1.4.2 Misi Roso Echo

Misi RM. Roso Echo adalah :

1. Mengutamakan layanan serta kualitas makanan yang akan disajikan.
2. Berinovasi secara berkala, baik dalam produk maupun sistem penjualan yang diterapkan.
3. Mampu bersaing dengan rumah makan lain yang ada di dalam ataupun di luar wilayah Cirebon.
4. Mengembangan usaha rumah makan tidak hanya di wilayah Cirebon, tapi juga ke seluruh Indonesia.

2.2 Landasan Teori

2.2.1 Data

Data adalah fakta atau apapun yang dapat digunakan sebagai *input* dan menghasilkan informasi. Dalam data terdapat himpunan data (*data-set*) yang merupakan kumpulan dari objek dan atributnya. Atribut merupakan sifat atau

karakteristik dari suatu objek yang biasanya dikenal sebagai *variabel*, *field*, karakteristik atau fitur.

Salah satu himpunan data (*data-set*) adalah *record* data, yaitu data yang terdiri dari sekumpulan *record*, yang masing-masing terdiri dari satu set atribut yang tetap. Salah satu yang termasuk dalam tipe data *record* yaitu data transaksi. Data transaksi merupakan sebuah tipe khusus dari *record* data, dimana tiap *record* (transaksi) meliputi satu *set item* [2].

Tabel 2.1 Contoh data transaksi

No	NoTable	NoBill	Date	Time	Server	Cetak	NamaMenu	Kategori	Qty	Harga	SubTotal	Pajak	GrandTotal	Cash	ChangeDue
1	1	006093	01/10/2013	09:49:34	Didin	Oka	Nasi	Jenis Lain-lain	1	Rp 3.000					
1	1	006093	01/10/2013	09:49:34	Didin	Oka	Sate Jamur	Jenis Makanan	2	Rp 3.000					
1	1	006093	01/10/2013	09:49:34	Didin	Oka	Daging Gepuk	Jenis Makanan	1	Rp 10.000	Rp 28.000	Rp 2.800	Rp 30.800	Rp 50.000	Rp 19.200
1	1	006093	01/10/2013	09:49:34	Didin	Oka	Gorengan / Bacem	Jenis Makanan	2	Rp 2.000					
1	1	006093	01/10/2013	09:49:34	Didin	Oka	Bakmi	Jenis Makanan	1	Rp 10.000					

2.2.2 Database

2.2.2.1 Pengertian Database

Database adalah susunan *record* data operasional lengkap dari suatu organisasi atau perusahaan yang diorganisir dan disimpan secara ter*integrasi* dengan menggunakan metode tertentu dalam komputer sehingga mampu memenuhi informasi yang optimal yang dibutuhkan oleh para pengguna [3].

2.2.2.2 Data pada Database dan Hubungannya

Ada 3 jenis data pada sistem *database*, yaitu [3]:

1. Data operasional dari suatu organisasi, berupa data yang disimpan dalam *database*.
2. Data masukan (*input* data), data dari luar sistem yang dimasukkan melalui peralatan *input* (*keyboard*) yang dapat merubah data operasional.
3. Data keluaran (*output* data), berupa laporan melalui peralatan *output* sebagai hasil dari dalam sistem yang mengakses data operasional.

2.2.2.3 Keuntungan dan Kerugian Pemakaian Sistem Database

Keuntungan [3] :

1. Terkontrolnya kerangkapan data dan inkonsistensi.
2. Terpeliharanya keselarasan data.
3. Data dapat dipakai secara bersama-sama.
4. Memudahkan penerapan standarisasi dan batas-batas pengamanan.
5. Terpeliharanya keseimbangan atas perbedaan kebutuhan data dari setiap aplikasi.
6. Program/data *independent*.

Kerugian :

1. Mahal dalam implementasinya.
2. Rumit/kompleks.
3. Penanganan proses *recovery backup* sulit.
4. Kerusakan pada sistem basis data dapat mempengaruhi.

2.2.3 *Data Mining*

2.2.3.1 Pengertian *Data Mining*

Data mining adalah serangkaian proses untuk menggali nilai tambah berupa informasi yang selama ini tidak diketahui secara manual dari suatu basis data dengan melakukan penggalian pola-pola dari data dengan tujuan untuk memanipulasi data menjadi informasi yang lebih berharga, yang diperoleh dengan cara mengekstraksi dan mengenali pola yang penting atau menarik dari data yang terdapat dalam basis data.

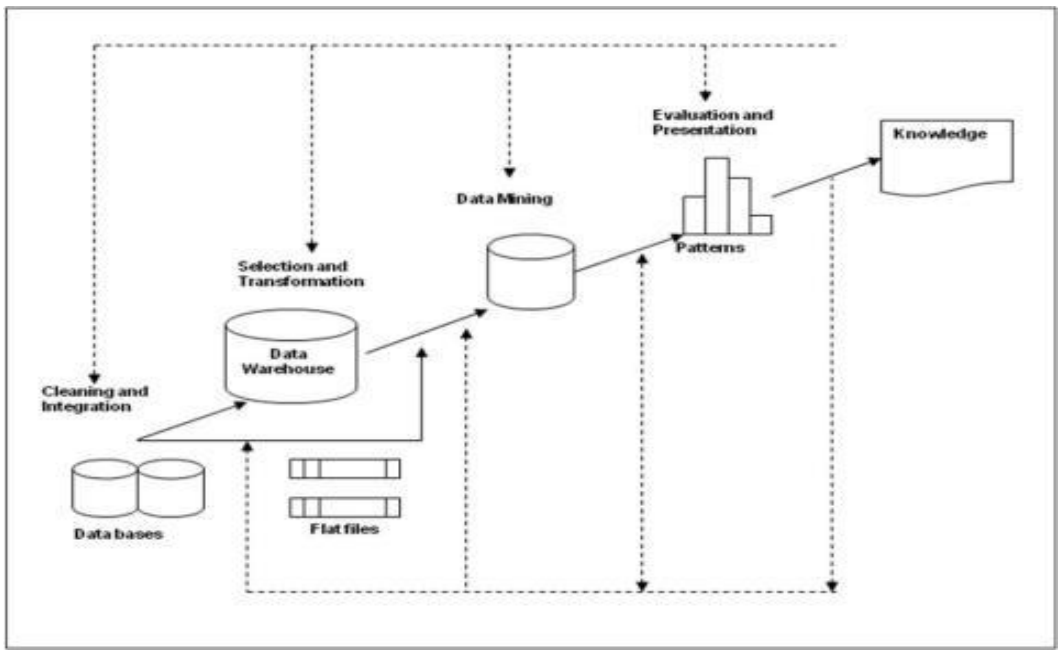
Data mining juga dikenal dengan nama *Knowledge Discovery* (*mining*) *in Database* (KDD), ektraksi pengetahuan (*knowledge extraction*). Secara umum, definisi *data mining* dapat diartikan sebagai berikut [4] :

1. Proses penemuan pola yang menarik dari data yang tersimpan dalam jumlah besar.

2. Ekstraksi dari suatu informasi yang berguna atau menarik (*non-trivial*, implisit, sebelumnya belum diketahui potensian kegunaannya) pola atau pengetahuan dari data yang di simpan dalam jumlah besar.
3. Eksplorasi dari analisa secara otomatis atau semiotomatis terhadap data-data dalam jumlah besar untuk mencari pola dan aturan yang berarti.

2.2.3.2 Konsep *Data Mining*

Data mining sangat diperlukan terutama dalam mengelola data yang sangat besar untuk memudahkan aktifitas *recording* suatu transaksi dan untuk proses *data warehousing* agar dapat memberikan informasi yang akurat bagi pengguna *data mining.* Alasan utama *data mining* sangat dibutuhkan dalam industri informasi karena tersedianya data dalam jumlah yang besar dan semakin besarnya kebutuhan untuk mengubah data tersebut menjadi informasi dan pengetahuan yang berguna karena sesuai fokus bidang ilmu ini yaitu melakukan kegiatan mengekstraksi atau menambang pengetahuan dari data yang berukuran atau berjumlah besar. Informasi inilah yang nantinya sangat berguna untuk pengembangan. Berikut adalah langkah-langkah dalam data mining [4]:



Gambar 2.3 Konsep Data *Mining*

1. *Data cleaning* yaitu untuk menghilangkah *noise* data yang tidak konsisten.
2. *Data integration* yaitu menggabungkan beberapa *file* atau *database*.
3. *Data selection* yaitu data yang relevan dengan tugas analisis dikembalikan ke dalam *database* untuk proses data *mining*.
4. *Data transformation* yaitu data berubah atau bersatu menjadi bentuk yang tepat untuk menambang dengan ringkasan performa atau operasi agresi.
5. *Data mining* yaitu proses esensial dimana metode yang intelejen digunakan untuk mengekstrak pola data.
6. *Knowledge Discovery* yaitu proses esential dimana metode yang intelejen digunakan untuk mengekstrak pola data.
7. *Pattern evolution* yaitu untuk mengidentifikasi pola yang benar-benar menarik yang mewakili pengetahuan berdasarkan atas beberapa tindakan yang menarik.
8. *Knowledge presentation* yaitu gambaran teknik visualisasi dan pengetahuan digunakan untuk memberikan pengetahuan yang telah ditambah kepada *user*.

2.2.3.3 Tahapan Data Mining

Tahapan dalam melakukan data *mining* salah satunya adalah *preprocessing* data. Tahapan ini biasanya diperlukan karena data yang akan digunakan belum baik, yang disebabkan oleh beberapa faktor berikut ini [5]:

1. ***Incomplete*** : tidak lengkapnya nilai suatu atribut, tidak lengkapnya atribut-atribut yang penting, atau hanya mempunyai data yang merupakan rekapitulasi.

   Contoh : pekerjaan = “ ”

   Hal tersebut dapat disebabkan oleh perbedaan kebijakan ketika dapat tersebut dianalisa, bisa juga disebabkan oleh permasalahan yang ditimbulkan oleh manusia, *hardware*¸atau *software*.
2. ***Noisy*** : mengandung *error* atau merupakan *value* yang tidak wajar.

   Contoh : gaji “-100”

   Timbul karena kesalahan entry oleh manusia atau komputer *error*, atau karena terdapat kesalahan ketika proses pengiriman data.

3. ***Inconsisten* :** mengandung nilai yang saling bertentangan.

   Contoh : umur = “42” dan ulang tahun = “02/10/1981”

   Masalah ini muncul karena perbedaan sumber data, karena pada data *mining* data didapatkan dari banyak sumber dan sangat mungkin terdapat perbedaan persepsi pengolahan data. Selain itu, perbedaan ini muncul karena pelanggaran terhadap *fungsional dependency*, misalnya melakukan perubahan pada data yang terhubung dengan data lain.

Padahal data yang bisa diterima untuk bisa diproses menjadi informasi atau *knowledge* adalah data yang mempunyai kualitas diantaranya :

1. Akurat
2. Lengkap
3. Konsisten
4. Relevan
5. Bisa dipervaya
6. Mempunyai nilai tambah
7. Kemudahan untuk dimengerti

Jika data tidak dalam kualitas seperti yang telah diuraikan diatas, maka kualitas analisis data menjadi kurang sehingga hasilnya pun kurang bermakna. Hal tersebut harus dihindarkan karena hasil analisa yang salah dapat berujung pada solusi yang salah. Untuk itu, perlu dilakukan *preprocessing* data yang bertujuan agar membuat data menjadi lebih berkualitas. Adapun tahapan-tahapannya adalah sebagai berikut :

1. ***Data Cleaning*** : mengisi/mengganti nilai-nilai yang hilang, menghaluskan data yang *noisy*, mengidentifikasi dan menghilangkan data yang tidak wajar, dan menyelesaikan masalah *inconsistensi* data.
2. ***Data Integration*** : menggabungkan beberapa *database* dan *file* menjadi satu sehingga didapatkan sumber data yang besar.
3. ***Data Transformation*** : normalisasi dan agregasi data.
4. ***Data Reduction*** : mengurangi *volume* data namun tetap mempertahankan arti dalam hal hasil analisis data.

5. ***Data Discretization*** : merupakan bagian dari data *reduction* dengan memperhitungkan data yang signifikan, khususnya pada data numerik.

2.2.4 *Assosiation Rule*

Assocition rule adalah salah satu teknik utama atau prosedur dalam *Market Basket Analysis* untuk mencari hubungan antar-*item* dalam suatu *dataset* dan menampilkan bentuk *association rule* [10]. *Association rule* (aturan asosiasi) akan menemukan pola tertentu untuk mengasosiasikan data yang satu dengan data yang lain. Untuk mencari *association rule* dari suatu kumpulan data, tahap pertama yang harus dilakukan adalah mencari *frequent itemset* terlebih dahulu. *Frequent itemset* adalah sekumpulan *item* yang sering muncul secara bersamaan. Setelah semua pola *frequent itemset* ditemukan, barulah mencari aturan assosiatif atau aturan keterkaitan yang memenuhi syarat yang telah ditentukan.

Jika diasumsikan bahwa barang yang dijual di swalayan adalah semesta, maka setiap barang akan memiliki *boolean* variabel yang akan menunjukkan keberadaannya atau tidak barang tersebut dalam satu transaksi atau satu keranjang belanja. Pola *boolean* yang didapat digunakan untuk menganalisa barang yang dibeli secara bersamaan. Pola tersebut dirumuskan dalam sebuah *association rule*. Sebagai contoh konsumen biasanya akan membeli kopi dan susu yang ditujukkan sebagai berikut :

Kopi → susu [*support* = 2%, *confidence* = 60%]

Association rule diperlukan suatu variabel ukuran yang ditentukan sendiri oleh *user* untuk menentukan batasan sejauh mana atau sebanyak apa *output* yang diinginkan *user*. *Support* dan *Confidence* adalah sebuah ukuran kepercayaan dan kegunaan suatu pola yang telah ditemukan. Nilai *support* 2% menunjukkan bahwa keseluruhan dari total transaksi konsumen membeli kopi dan susu secara bersamaan yaitu sebanyak 2%. Sedangkan *confidence* 60% yaitu menunjukkan bila konsumen membeli kopi dan pasti membeli susu sebesar 60%.

2.2.5 Algoritma A priori

2.2.5.1 Pengertian Algortima A priori

Algoritma apriori adalah salah satu algoritma yang melakukan pencarian *frequent itemset* dengan menggunakan teknik *association rule*. Algortima apriori menggunakan pengetahuan *frekuensi* atribut yang telah diketahui sebelumnya untuk memproses informasi selanjutnya. Pada algoritma apriori, menentukan kandidat yang mungkin muncul dengan cara memperhatikan minimum *support* dan minimum *confidence*. *Support* adalah nilai pengunjung atau persentase kombinasi sebuah *item* dalam *database*.

Nilai *support* sebuah *item* diperoleh dengan rumus sebagai berikut [11] :

$$Support\ (A) = \left(\frac{\text{Jumlah transaksi mengandung A}}{\text{Total Transaksi}}\right) \text{x } 100\% \quad \text{.....(2.1)}$$

Sedangkan nilai dari *support* 2 *item* diperoleh dari rumus berikut :

$$Support\ (A,B) = \left(\frac{\text{Jumlah transaksi mengandung A dan B}}{\text{Total Transaksi}}\right) \text{x } 100\% \quad \text{.....(2.2)}$$

Sedangkan *confidence* adalah nilai kepastian yaitu kuatnya hubungan antar item dalam sebuah apriori. *Confidence* dapat dicari setelah pola frekuensi menculnya sebuah *item* ditemukan. Rumus untuk menghitung *confidence* adalah sebagai berikut :

$$Confidence = \frac{\text{Jumlah transaksi yang mengandung A dan B}}{\text{Jumlah transaksi yang mengandung A}} \quad \text{.....(2.3)}$$

Proses utama yang dilakukan dalam algoritma apriori untuk mendapat *frequent itemset* yaitu :

1. *Join* (Penggabungan)
   Proses ini dilakukan dengan cara pengkombinasian *item* dengan *item* yang lainnya hingga tidak dapat terbentuk kombinasi lagi.
2. *Prune* (Pemangkasan)
   Proses pemangkasan yaitu hasil dari *item* yang telah dikombinasikan kemudian dipangkas dengan menggunakan minimum *support* yang telah

ditentukan oleh *user*. Pada iterasi ke-k akan ditemukan semua *itemset* yang memiliki k *item*, disebut dengan k-*itemset*. Tiap iterasi terdiri dari dua tahap, yaitu :

a. Gunakan *frequent* (k-1) *itemset* untuk membangun kandidat *frequent* k-*itemset.*
b. Gunakan *scan database* dan pencocokan pola untuk mengumpulkan hitungan untuk kandidat *itemset*.

2.2.5.2 Langkah-Langkah Proses Perhitungan Association Rule Dengan Algoritma Apriori

Proses perhitungan *association rule* terdiri dari beberapa tahap adalah sebagai berikut [12] :

1. Sistem men-*scan database* untuk mendapat kandidat 1-*itemset* (himpunan *item* yang terdiri dari 1 *item*) dan menghitung nilai *support*-nya. Kemudian nilai *support*-nya tersebut dibandingkan dengan *minimum support* yang telah ditentukan. Jika nilainya lebih besar atau sama dengan *minimum support*, maka *itemset* tersebut termasuk *large itemset*.
2. *Itemset* yang tidak termasuk dalam *large itemset* tidak diikutkan dalam iterasi selanjutnya (di pangkas).
3. Pada iterasi kedua, sistem akan menggunakan hasil *large itemset* pada iterasi pertama (L_1) untuk membentuk kandidat *itemset* kedua (L_2). Pada iterasi selanjutnya sistem akan menggunakan hasil *large itemset* pada iterasi sebelumnya (L_{k-1}) untuk membentuk kandidat *itemset* berikut (L_k). Sistem akan menggabungkan (*join*) L_{k-1} dengan L_{k-1} untuk mendapatkan L_k. Seperti pada iterasi sebelumnya sistem akan menghapus (memangkas) kombinasi *itemset* yang tidak termasuk dalam *large itemset*.
4. Setelah dilakukan operasi *join*, maka pasangan *itemset* baru hasil proses *join* tersebut dihitung *support*-nya.

5. Proses pembentuk kandidat yang terdiri dari proses penggabungan dan pemangkasan akan terus dilakukan hingga himpunan kandidat *itemset*-nya *null*, atau sudah tidak ada lagi kandidat yang akan terbentuk.
6. Setelah itu, dari hasil *frequent itemset* tersebut dibentuk *association rule* yang memenuhi nilai *support* dan *confidence* yang telah ditentukan.
7. Pada pembentukan *associaton rule*, nilai yang sama dianggap sebagai satu nilai.
8. *Association rule* yang terbentuk harus memenuhi nilai *minimum* yang telah ditentukan.
9. Untuk setiap *large itemset* L, cari himpunan bagian L yang tidak kosong. Untuk setiap himpunan bagian tersebut, dihasilkan *rule* dengan bentuk $aB_{(L-a)}$ jika *support*-nya (a) lebih besar dari *minimum support*.

2.2.6 *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. *Lift ratio* dapat dihitung dengan rumus :

$$\frac{Support\ (A \cap B)}{Support\ (A) x\ Support (B)} \quad \text{.....(2.4)}$$

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.

2.2.7 Alat-Alat Pemodelan Sistem

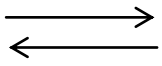
2.2.7.1 Diagram Konteks

Diagram konteks adalah diagram aliran data pada tingkat paling atas yang merupakan penggambaran yang berfungsi untuk memperlihatkan interaksi atau hubungan langsung antara sistem informasi dengan lingkungannya. Diagram konteks menggambarkan sebuah sistem berupa sebuah proses yang berhubungan dengan satu atau beberapa entitas/entity.

2.2.7.2 *Data Flow Diagram* (DFD)

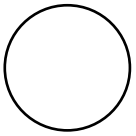
Data Flow Diagram (DFD) adalah suatu alat pemodelan yang digunakan untuk memodelkan fungsi dari sistem, menggambarkan secara rinci mengenai sistem sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan menunjukkan dari dan kemana data mengalir serta menyimpannya. Pada umumnya dimulai dari 0,1,2,dst. Level ke-0 disebut dengan diagram konteks yang menggambarkan sistem secara global.

Komponen DFD [10]:



Data Flow/Aliran Data diwakili dengan tanda panah. Digunakan untuk menunjukkan pergerakan/aliran dari kumpulan data/informasi dari satu bagian sistem ke sistem lainnya.

Data Storage/penyimpanan data. Merupakan bagian dari DFD yang digunakan untuk menunjukkan suatu kumpulan dari paket data.



Proses, adalah bagian dari DFD yang mengubah satu atau lebih masukan menjadi keluaran-keluaran. Nama lainnya : *bubble, function.*



Entitas adalah seseorang atau sekelompok orang dalam suatu kelompok organisasi atau departemen lain didalam perusahaan.

Dapat terdiri dari orang, unit terkait yang berinteraksi. Nama lainnya : *terminator*

2.2.7.3 Kamus Data

Kamus data adalah suatu penjelasan tertulis mengenai data yang berada dalam *database*. Kamus data ikut berperan dalam perancangan dan pembangunan sistem informasi karena berfungsi untuk menjelaskan arti aliran data dan menyimpan dalam menggambarkan *data flow diagram*, menjelaskan spesifikasi nilai dan satuan yang relevan terhadap data yang mengalir dalam sistem tersebut, mendeskripsikan komposisi paket data yang bergerak melalui aliran.

Fungsi kamus data/data dictionary [14]:

1. Menjelaskan arti aliran data dan penyimpanan data dalam DFD.
2. Mendeskripsikan komposisi paket data (elemen data) yang bergerak melalui aliran data.
3. Mendeskripsikan nilai dan satuan (struktur data) yang relevan bagi penyimpanan dan aliran.
4. Mendeskripsikan hubungan *detail* antar penyimpanan.

Tabel 2.2 Struktur Kamus Data

Nama Data	
Deskripsi	
Bentuk Data	
Struktur data	

2.2.7.4 Spesifikasi Proses (*Process Specification/PSPEC*)

Digunakan untuk mendeskripsikan proses yang terjadi pada level paling rendah dari DFD. Hubungan DFD dengan PSPEC [14]:

1. Semua proses dalam DFD yang tidak dapat dipecah lagi dan harus didefinisikan dalam PSPEC.
2. Aliran data masuk (*input*) dan keluar (*output*) dalam DFD dan hubungan ke *data store* harus sesuai dan relevan dalam PSPEC.

3. Setiap elemen data dalam PSPEC harus :
   a. Nama dari aliran data atau *data store*.
   b. Atau komponen dalam kamus data atau *data store* yang berhubungan dalam DFD.

Tabel 2.3 Struktur Spesifikasi Proses

No. Urut	Proses	Keterangan
	No. Proses	
	Nama Proses	
	Source (Sumber)	
	Input	
	Output	
	Destination (tujuan)	
	Logika Proses	

2.2.8 Metode Pengujian Eksperimental

Metode pengujian eksperimental adalah suatu prosedur penelitian yang dilakukan untuk mengetahui hubungan sebab akibat dua variabel atau lebih dengan mengendalikan pengaruh dari *variabel* lain. Metode ini dilaksanakan dengan memberikan *variabel* bebas secara sengaja terhadap objek penelitian untuk mengetahui akibat dari *variabel* yang terkait. Adapun tujuan dari metode ini adalah untuk menguji efektifitas dan efisiensi dari suatu pendekatan, metode atau teknik, sehingga dapat diketahui apakah hasil yang didapatkan dapat diterapkan dalam objek yang sebenarnya atau tidak. Dalam metode eksperimen, terdapat tiga prinsip dasar yang terdapat dalam *desain* eksperimen, yaitu [1]:

1. Replikasi, merupakan pengulangan dari ekperimen dasar.
2. *Randomization*, digunakan pada uji signifikan valid. Uji siginifikan akan valid jika pengamatan didistribusikan secara bebas yang dilakukan dengan pengambilan *sampel* secara acak.
3. *Blocking*, merupakan prinsip yang digunakan untuk mengisolasi *treatment* dari pengaruh faktor lain agar hasil eksperimen menjadi lebih akurat.