BAB II

TINJAUAN PUSTAKA

2.1 Ruang Lingkup Objek Penelitian

Objek penelitian yang sedang diteliti adalah PT. Indomarco Prismatama cabang Bandung.

2.1.1 Profil PT. Indomarco Prismatama

Indomaret merupakan jaringan minimarket yang menyediakan kebutuhan pokok dan kebutuhan sehari-hari dengan luas jangkauan kurang dari 200 m^2. Dikelola oleh PT. Indomarco Prismatama, cikal bakal pembukaan Indomaret di Kalimantan dan toko pertama dibuka di Ancol, Jakarta Utara. Toko pertama Indomaret dibuka pada bulan Agustus 1987 . **IND[9]**

Tahun 1997 perusahaan mengembangkan bisnis gerai waralaba pertama di Indonesia, setelah Indomaret teruji dengan lebih dari 230 gerai. Pada Mei 2003 Indomaret meraih penghargaan "Perusahaan waralaba 2003" dari presiden Megawati Soekarno Putri. Hingga Januari 2010 Indomaret mencapai 3967 gerai. Dari total itu 220, gerai adalah milik sendiri dan sisanya 1764 gerai waralaba milik masyarakat, yang tersebar Jabotabek, Jawa Barat, Jawa Tengah, Jawa Timur, DIY, Bali dan Lampung. Di DKI Jakarta terdapat sekitar 488 gerai.

Indomaret mudah ditemukan di daerah perumahan, gedung perkantoran dan fasilitas umum karena penempatan lokasi gerai didasarkan pada motto "mudah dan hemat". Lebih dari 3500 jenis produk makanan dan non makanan tersedia dengan harga bersaing memenuhi hampir semua kebutuhan konsumen sehari-hari. Didukung oleh 12 pusat distribusi (Ancol Jakarta, Cimanggis Depok, Tanggerang, Bekasi, Parung, Bandung, Semarang, Jogyakarta, Jember, Surabaya, Lampung, Medan) dengan menjalin lebih dari 500 pemasok, Indomaret merupakan salah satu aset bisnis yang sangat menjanjikan. Keberadaan Indomaret diperkuat oleh anak perusahaan dibawah bendera grup INTRACO, yaitu Indogrosir, BSD Plaza dan Charmant.

2.1.2 Visi dan Misi

Visi Indomaret adalah menjadi aset nasional dalam bentuk jaringan ritel waralaba yang unggul dalam persaingan global. Misi Indomaret adalah menyediakan kebutuhan pokok sehari-hari dengan mudah dan hemat [9].

2.1.3 Logo dan Maskot Indomaret

Logo dan maskot Indomaret diuraikan sebagai berikut :

a) Logo Indomaret

Logo resmi Indomaret dapat dilihat pada gambar 2.1.



Gambar 2.1 Logo Indomaret

b) Maskot Indomaret

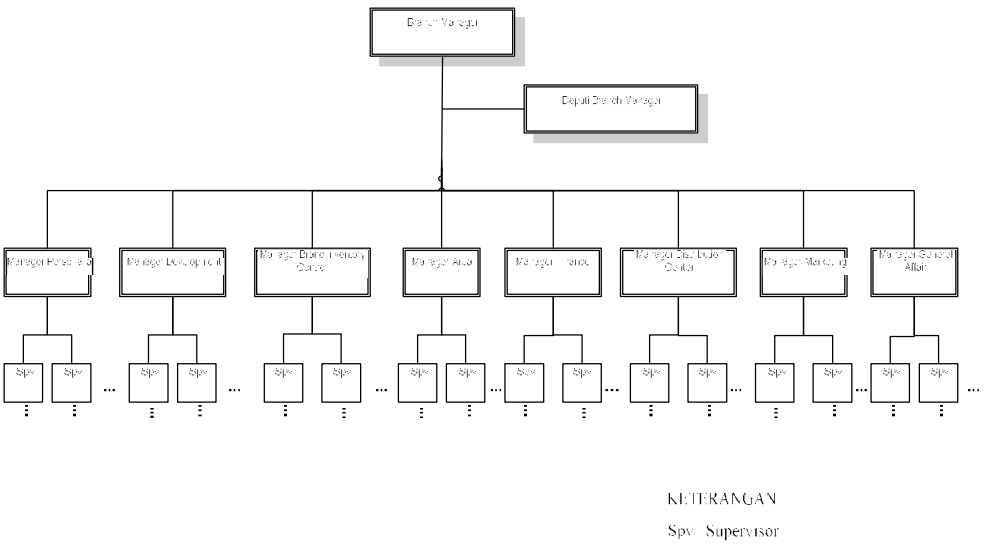
Maskot Indomaret dibuat pada tahun 2005 melalui ajang kompetisi yang diikuti oleh kalangan umum. Desain dari pemenang kompetisi secara resmi digunakan oleh Indomaret sebagai maskot perusahaan. Maskot Indomaret diberi julukan " Si Domar ". Si Domar dapat dilihat pada gambar 2.2.



Gambar 2.2 Maskot Indomaret

2.1.4 Struktur Organisasi di PT. Indomarco Prismatama Cabang Bandung

Secara garis besar, PT. Indomarco Prismatama Cabang Bandung dipimpin oleh seorang kepala cabang sebagai pimpinan tertinggi perusahaan yang dibantu oleh seseorang yang menempati posisi sebagai wakil pemimpin cabang pada struktur inti dari perusahaan. Kepala cabang dalam menjalankan kegiatan bisnis dibantu oleh para manajer yang dibagi atas beberapa departemen. Seperti halnya manajer secara umum, manajer juga dibantu oleh para supervisor lainnya sebagai bawahan yang masing-masing bertanggung jawab atas bagian/departemen yang mereka tempati. Untuk lebih jelas mengenai struktur organisasi PT. Indomarco Prismatama Cabang Bandung dapat dilihat pada gambar 2.3.



Gambar 2.3 Struktur Organisasi Perusahaan

2.2 Landasan Teori

Subbab landasan teori ini akan menjelaskan landasan teori mengenai sistem yang dirancang dan perangkat lunak yang akan dibangun, penjelasan meliputi teori-teori yang digunakan, perangkat lunak, perangkat keras dan bahasa pemrograman yang dibutuhkan dalam membangun aplikasi sistem pendukung keputusan pemberian beasiswa di PT. Indomarco Prismatama Cabang Bandung.

2.2.1 Pengertian Beasiswa

Pada dasarnya, beasiswa adalah penghasilan bagi yang menerimanya. Beasiswa ini sesuai dengan ketentuan pasal 4 ayat (1) UU PPh/2000. Disebutkan pengertian penghasilan adalah tambahan kemampuan ekonomis dengan nama dan dalam bentuk apa pun yang diterima atau diperoleh dari sumber Indonesia atau luar Indonesia yang dapat digunakan untuk konsumsi atau menambah kekayaan Wajib Pajak (WP), karena beasiswa bisa diartikan menambah kemampuan ekonomis bagi penerimanya, berarti beasiswa merupakan penghasilan (Jawa Pos, 2009). **JOU[10]**.

2.2.2 Metode Penelitian

Metode penelitian pada dasarnya merupakan cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu **SUG[7]**. Berdasarkan hal tersebut terdapat empat kata kunci yang perlu diperhatikan yaitu, cara ilmiah, data, tujuan, dan kegunaan. Cara ilmiah berarti kegiatan penelitian itu didasarkan pada cirri-ciri keilmuan, yaitu rasional, empris, dan sistematis. Rasional berarti kegiatan penelitian itu dilakukan dengan cara-cara yang masuk akal, sehingga

terjangkau oleh penalaran manusia. Empiris berarti ca ra-cara yang dilakukan itu dapat diamati oleh indera manusia, sehingga orang lain dapat mengamati dan mengetahui cara-cara yang digunakan. Sistematis artinya, proses yang digunakan dalam penelitian itu menggunakan langkah-langkah tertentu yang bersifat log is.

Melalui penelitian manusia dapat menggunakan hasilnya. Secara umum data yang telah diperoleh dari penelitian dapat digunakan untuk memahami, memecahkan dan mengantisipasi masalah. Memahami berarti memperjelas suatu masalah atau informasi yang tidak dik etahui dan selanjutnya menjadi tahu, memecahkan berarti meminimalkan masalah, dan mengantisipasi berarti mengupayakan agar masalah tidak terjadi, walaupun masalah terjadi kita mempunyai cara untuk menyelesaikan dan menuntaskan masalah tersebut.

2.2.2.1 Penerapan Metode Kano

Metode Kano yang dikembangkan oleh Norioko Kano dari *Tokyo Riko University* pada tahun 1984, merupakan salah satu metode yang dapat digunakan untuk pemetaan preferensi pelanggan dalam memperoleh kepuasan atas produk atau jasa tertentu **BLO[9]**. Dengan metode ini dapat dipahami atribut-atribut produk atau jasa yang dapat diterima dan akan mempengaruhi konsumen. Beberapa strategi dapat dikembangkan untuk memperoleh posisi yang lebih baik. Ketidaktahuan terhadap kategori atribut layanan dapat menimbulkan *negative image* bagi korporasi karena gagal memberikan jasa sesuai ketegori atribut jasa yang memberikan nilai lebih bagi pelanggan.

Beberapa dasar yang digunakan dalam pengembangan Metode Kano adalah atribut kualitas yang sulit untuk dilihat *(intangible*) dapat dipahami dengan

mengategorikan atribut-atribut secara jelas, dalam pelayanan kebutuhan konsumen, kepuasan pelanggan sebanding dengan fungsi pelayanan yang maksimal, atribut-atribut kebutuhan konsumen tidak selalu masuk didalam kategori *one dimensional*, tetapi bisa masuk *must be* atau *attractive*, atribut - atribut kebutuhan konsumen dapat dipetakan/ diklasifikasikan melalui kuesionair yang ditujukan kepada konsumen. Atribut-atribut kebutuhan pelanggan dapat dibedakan menjadi enam kategori yaitu:

a. *Attractive (excitement needs)* merupakan kategori yang tingkat kepuasan pelanggan akan meningkat sangat tinggi dengan meningkatnya kinerja atribut, sedangkan penurunan kinerja atribut tidak akan menyebabkan penurunan tingkat kepuasan.

b. *One dimensional (performance needs)* adalah kategori yang apabila terpenuhi maka dapat meningkatkan kepuasan pelanggan dan akan menyebabkan ketidakpuasan bila tidak terpenuhi, atau dengan kata lain tingkat kepuasan pelanggan berhubungan *linier* dengan kinerja atribut, sehingga menurunnya kinerja atribut akan menurunkan pula tingkat kepuasan pelanggan.

c. *Must be* (*basic needs*) merupakan kategori dimana pelanggan menjadi tidak puas apabila kinerja dari atribut yang bersangkutan rendah tetapi kepuasan pelanggan tidak akan meningkat jauh diatas netral meskipun kinerja dari atribut tersebut tinggi. Kategori ini merupakan kebutuhan dasar yang harus dipenuhi oleh penyedia jasa kepada konsumennya, apabila perusahaan ingin meningkatkan performansi pada kategori ini maka tidak akan meningkatkan

kepuasan tetapi atribut-atribut ini wajib dipenuhi oleh perusahaan dengan kata lain atribut tersebut harus ada pada setiap produk atau jasa agar dapat diterima karena bila tidak maka konsumen tidak akan memakai jasa yang kita sediakan.

d. Kategori *Indefferent* terjadi jika ada maupun tidaknya layanan tidak akan memberikan pengaruh kepada kepuasan konsumen.

e. *Reverse* merupakan kategori kebalikan dari kategori *one dimensional* yaitu derajat kepuasan konsumen lebih tinggi jika layanan berlangsung tidak semestinya, dibandingkan layanan yang berjalan lebih baik.

f. Kategori *Questionable* merupakan kondisi yang kadangkala konsumen puas atau tidak puas jika layanan itu diberikan atau tidak diberikan.

Ada beberapa keuntungan yang diperoleh ketika perusahaan mengklasifikasikan kebutuhan pelanggan dengan Motode Kano. Dengan menggunakan metode ini, perusahaan dapat melakukan prioritas pengembangan produk setelah mengetahui dengan lebih dalam atribut-atribut yang ada pada produk atau jasa yang diberikan. Sebagai contoh, pada atribut-atribut yang masuk dalam kategori *must be* akan lebih baik bila tidak dikembangkan lebih lanjut dengan investasi yang tinggi. Hal ini karena atribut tersebut memang merupakan atribut yang harus ada. Lebih baik pengembangan atribut dilakukan pada yang masuk kategori *one dimensional*. Ketika perusahaan dapat menemukan dan memenuhi kategori *attractive* akan menciptakan kemungkinan besar untuk perbedaan, yaitu membedakan produk jasa perusahaan dengan pihak pesaing.

Ada tiga tahapan untuk pengukuran kualitas layanan dengan Metode Kano. Petama adalah identifikasi atribut, yang dilakukan dengan melakukan pengelompokan berdasarkan dimensi kepuasan pelanggan dari atribut-atribut tersebut, disusun prakuesioner yang telah dikelompokkan. Kedua ada pah tahapan test prakuesioner untuk mengidentifikasi pertanyaan mana yang dianggap tidak perlu oleh konsumen. Dan ketiga adalah penyebaran kuesioner yang digunakan untuk menentukan preferensi konsumen.

Ada dua jenis kuesioner yang digunakan untuk mengeta hui preferensi pelanggan yaitu *Functional questions*, berisi pertanyaan apa yang pelanggan rasakan jika atribut-atribut layanan tersebut ada atau tersedia dan *Dysfunctional questions*, berisikan apa yang dirasakan oleh pelanggan jika atribut-atribut layanan tersebut tidak terpenuhi. Pemberian nilai didalam kuesioner dengan klasifikasikan dalam lima jenis, yaitu :

a. Suka, artinya ketika layanan tersebut sangat berguna bagi pelanggan, atau menikmati akan adanya pelayanan.
b. Mengharapkan yang artinya layanan atau f asilitas tersebut merupakan suatu kaharusan bagi pelanggan atau merupakan layanan yang harus dipenuhi oleh pihak bank.
c. Netral ketika ada tidaknya layanan atau fasilitas tersebut tidak akan berpengaruh terhadap pelanggan.
d. Memberi toleransi ketika pelanggan tidak suka akan layanan tersebut tetapi dapat menerima kondisi tersebut.
e. Tidak suka, ketika pelanggan tidak dapat menerima kondisi tersebut.

Untuk mengklasifikasikan atribut berdasarkan Kategori Kano, data yang diperoleh digunakan sebagai dasar menentukan kategori atribut tiap responden berdasarkan kategori kano. Tabel 2.1 digunakan sebagai referensi.

Tabel 2.1 Referensi Metode Kano

		Dysfunctional				
		1	2	3	4	5
Functional	1	Q	A	A	A	O
	2	R	I	I	I	M
	3	R	I	I	I	M
	4	R	I	I	I	M
	5	R	R	R	R	Q

Keterangan:
Q = *Questionable*, R = *Reserve*, A = *Attractive*, I = *Indifferent*, dan O = *One dimensional*, M = *Must be*.
1 = Suka, 2 = Mengharapkan, 3 = Netral, 4 = Toleransi, 5 = Tidak suka.

Untuk menentukan kategori Kano tiap atribut dilakukan dengan menggunakan *Blauth's formula*, sebagai berikut:

a. Jika (*one dimensional+attractive+must be*) > (*indifferent+reverses +questionable*) maka nilai diperoleh dari yang paling maksimum dari (*one dimensional, attractive, must be*).

b. Jika (*one dimensional+attractive+must be*) < (*indifferent+reverses+ questionable*) maka nilai diperoleh dari yang paling maksimum dari (*indifferent, reverses, questionable*).

Hasil dari analisis ini akan digunakan sebagai langkah untuk tindakan perbaikan. Secara umum perbaikan yang dilakukan adalah bagaimana memenuhi semua atribut *must be*, mempunyai kinerja lebih baik daripada pesaing pada atribut *one dimensional*, dan memasukkan atribut *attractive* yang beda dengan pesaing.

2.2.2.2 Pengolahan Data Kuesioner Menggunakan Model Kano

Data kuesioner diolah menggunakan Model Kano dengan langkah sebagai berikut **DIA[10]**:

a. Mengombinasikan jawaban fungsional dan disfungsional tiap atribut untuk tiap responden berdasarkan hasil tabel penentuan Katagori Kano, sehingga diperoleh Katagori Kano tiap atribut produk untuk tiap responden, seperti yang ditunjukkan pada tabel 2.2, misalnya, jika responden menjawab dengan skor 1 pada pertanyaan fungsional serta menjawab dengan skor 2 pada pernyataan disfungsional maka atribut tersebut masuk katagori A *(attractive).*

Tabel 2.2 Penentuan kategori Kano

NO	**Pilihan alternatif jawaban**				
	1	**2**	**3**	**4**	**5**
1	Q	A	A	A	O
2	R	I	I	I	A
3	R	I	I	I	A
4	R	I	I	I	A
5	R	R	R	R	Q

Sumber ; Walden, 1993

Keterangan :

Q = *questionable*	skor 1 = setuju
R = *reserve*	skor 2 = cukup setuju
A = *attractive*	skor 3 = biasa
O = *one dimensional*	skor 4 = kurang setuju
M = *must be*	skor 5 = tidak setuju
I = *indifferent*	

b. Menghitung frekuensi Katagori Kano untuk tiap-tiap atribut produk dari semua responden sesuai hasil kuesioner kemudian menentukan katagori kano dengan menggunakan Blauth's formula sebagai berikut :

1. Jika frekuensi katagori (O + A + M) > katagori (I + R + Q), maka grade yang diperoleh merupakan nilai yang paling maksimum dari (O,A,M).
2. Jika frekuensi katagori (O + A + M) < katagori (I + R + Q), maka grade yang diperoleh merupakan nilai yang paling maksimum dari (I,R,Q).

2.2.3 Definisi Sistem

Terdapat dua kelompok pendekatan didalam pendefinisian sistem, yaitu yang menekankan pada prosedur dan yang menekankan pada komponen atau elemen–elemen. Pendekatan sistem yang lebih menekankan pada prosedur mendefinikan sistem sebagai berikut :

"Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu".**JOG[2]**

Pendekatan sistem yang lebih menekankan pada komponen atau elemen – elemennya mendefinisikan sistem sebagai berikut :

"Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu". **JOG[2]**

2.2.3.1 Ciri-Ciri Utama Sistem

Ciri-ciri utama dari suatu sistem adalah **JOG[2]** :

1. Mengarah pada suatu tujuan tertentu, yaitu setiap elemen sistem melakukan kegiatan masing-masing sesuai dengan fungsinya, tetapi mengarah kepada tujuan sistem yang sama dan secara keseluruhan.
2. Bersifat terbuka, artinya sistem berinteraksi dengan lingkungannya .
3. Mempunyai mekanisme kontrol, yaitu suatu sistem mempunyai kemampuan untuk mengatur dirinya sendiri ke suatu posisi yang lebih baik dengan menjalankan kedinamisan dan mempersatukan keharmonisan dalam sistem itu sendiri.
4. Merupakan status keseluruhan, yaitu sistem merupakan suatu keseluruhan yang bulat dan utuh, dimana tujuan masing-masing bagian yang membentuk sistem tersebut saling menunjang untuk mencapai tujuan sistem secara keseluruhan.
5. Adanya proses transformasi, yaitu didalam sistem selalu ada proses transformasi, yang mengubah *input* menjadi *output*, atau mengubah suatu masukan menjadi suatu keluaran untuk mencapai tujuan yang diinginkan .
6. Adanya keterkaitan atau hubungan timbal-balik, yaitu adanya hubungan timbal-balik antara elemen-elemen sistem yang satu dengan yang lainnya ataupun dengan lingkungannya, sehingga apabila terjadi perubahan elemen dari suatu sistem, maka akan berpengaruh terhadap sistem secara keseluruhan.

2.2.3.2 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu antara lain sebagai berikut **JOG[2]** :

1. Komponen-komponen sistem (*Components*), merupakan salah satu karakteristik sistem yang berupa sub-sistem atau gagasan sistem.
2. Batas sistem (*Boundary*), merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya atau dengan lingkungan lainnya .
3. Lingkungan luar sistem *(Environment)*, segala sesuatu yang berada di luar batas dari sistem yang mempengaruhi operasi baik itu menguntungkan operasi sistem yang merupakan energi dari sistem yang harus dipelihara atau ditahan sehingga tidak menggangu atau merusak sistem.
4. Penghubung sistem (*interface*), merupakan media penghubung antara satu sub-sistem dengan sub-sistem lainnya.
5. Masukan sistem (*input)*, masukan sistem adalah data yang dimasukkan ke dalam sistem yang diproses untuk mendapat satu kesatuan atau informasi .
6. Keluaran sistem (*output*), keluaran dari sistem adalah data yang diolah serta diklasifikasikan menjadi keluaran yang berguna atau menjadi informasi yang dibutuhkan.
7. Pengolah sistem *(process)*, suatu sistem yang mempunyai bagian pengolahan data yang akan merubah masukan menjadi keluaran yang berguna.
8. Sasaran sistem *(goal)*, merupakan maksud tertentu atau tujuan atas sasaran, dimana yang menentukan masukan serta keluaran sistem yang dibutuhkan pada sasaran atau tujuan yang maksud.

2.2.4 Definisi Informasi

Berikut ini akan dikemukakan definisi dari informasi **JOG[2]** :

"Informasi adalah data yang diolah menjadi bentuk yang lebih berarti bagi yang menerimanya".

2.2.4.1 Kualitas Informasi

Sumber dari informasi adalah data, kualitas dari suatu informasi tergantung dari empat hal antara lain **JOG[2]** :

1. Akurat, berarti suatu informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan.
2. Tepat, berarti suatu informasi harus diterima oleh pemakai informasi tepat pada waktunya.
3. Lengkap, berarti suatu informasi diberikan dengan selengkap-lengkapnya agar dapat dimanfaatkan secara maksimal oleh pemakai informasi.
4. Relevan, berarti suatu informasi tersebut mempunyai manfaat yang sesuai dengan pemakainya.

2.2.4.2 Nilai Informasi

Nilai dari informasi *(Value of Information)* ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya **JOG[2]**. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Akan tetapi perlu diperhatikan bahwa informasi yang digunakan didalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan. Sebagian informasi tidak dapat ditaksir keuntungannya dengan suatu nilai tetapi dapat ditaksir dari nilai keefektifannya.

2.2.5 Definisi Sistem Informasi

Sistem informasi merupakan hal yang sangat penting bagi manajemen didalam mengambil keputusan. Berikut ini akan dikemukakan definisi dari sistem informasi :

"Sistem informasi adalah suatu cara tertentu untuk menyediakan informasi yang dibutuhkan oleh organisasi untuk beroperasi dengan cara yang sukses untuk mengorganisasi bisnis dengan cara yang menguntungkan". **JOG[2]**

2.2.5.1 Kegiatan Sistem Informasi

Dalam sebuah sistem informasi terdapat beberapa kegiatan yang dilakukan antara lain **JOG[2]**:

1. Masukan *(input)*, merupakan suatu kegiatan yang dilakukan untuk menyediakan data yang akan di proses.
2. Proses merupakan suatu kegiatan yang dibutuhkan untuk memproses suatu data yang akan menghasilkan suatu informasi yang memiliki nilai tambah.
3. Keluaran *(output)*, merupakan suatu kegiatan yang dilakukan untuk menghasilkan laporan dari proses yang diatas.

2.2.6 SPK (Sistem Pendukung Keputusan)

SPK sebagai sebuah sistem berbasis komputer yang membantu dalam proses pengambilan keputusan **JOU[12]**. SPK sebagai sistem informasi berbasis komputer yang adaptif, interaktif, fleksibel, yang secara khusus dikembangkan untuk mendukung solusi dari pemasalahan manajemen yang tidak terstruktur untuk meningkatkan kualitas pengambilan keputusan. Sistem pendukung

keputusan dapat ditarik satu definisi tentang SPK yaitu sebuah sistem berbasis komputer yang adaptif, fleksibel, dan interaktif yang digunakan untuk memecahkan masalah-masalah tidak terstruktur sehingga meningkatkan nilai keputusan yang diambil.

2.2.7 Himpunan Klasik *(Crisp)*

Pada himpunan klasik (*crisp*) **KUS[5]**, keberadaan suatu elemen pada suatu himpunan (A), hanya akan mempunyai dua kemungkinan keanggotaan, yaitu elemen tersebut menjadi anggota A atau tidak menjadi anggota A. Suatu nilai yang menunjukkan seberapa besar tingkat keanggotaan suatu elemen (x) dalam suatu himpunan (A), sering dikenal dengan nama nilai keanggotaan atau derajat keanggotaan, dinotasikan dengan $\mu_A(x)$. Pada himpunan klasik, hanya ada nilai keanggotaan, yaitu $\mu_A(x) = 1$ untuk x menjadi anggota A, dan $\mu_A(x) = 0$ untuk x bukan anggota A.

2.2.8 Metode FMADM (*Fuzzy Multiple Attribute Decission Making*)

Fuzzy Multiple Attribute Decision Making (FMADM) adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu **KUS[5]**. Inti dari FMADM adalah menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan. Pada dasarnya, ada 3 (tiga) pendekatan untuk mencari nilai bobot atribut, yaitu pendekatan subjektif, pendekatan objektif dan pendekatan integrasi antara subjektif dan objektif. Masing-masing pendekatan memiliki kelebihan dan kelemahan. Pada pendekatan subjektif, nilai bobot ditentukan berdasarkan subjektivitas dari para pengambil

keputusan, sehingga beberapa faktor dalam proses perankingan alternatif bisa ditentukan secara bebas. Pada pendekatan objektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektivitas dari pengambil keputusan.

Ada beberapa metode yang dapat digunakan untuk menyelesaikan masalah FMADM yaitu:

a. *Simple Additive Weighting Method* (SAW).

b. *Weighted Product* (WP).

c. ELECTRE.

d. TOPSIS (*Technique for Order Preference by Similarity to Ideal Solution).*

e. *Analytic Hierarchy Process* (AHP).

2.2.9 SAW (*Simple Additive Weighting*)

Metode SAW sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari *rating* kinerja pada setiap alternatif pada semua atribut. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada **KUS[5]**.

$$r_{ij} - \begin{cases} \dfrac{X_{ij}}{\underset{i}{Max}\, X_{ij}} & \text{Jika j adalah atribut keuntungan } (benefit) \\ \dfrac{\underset{i}{Min}\, X_{ij}}{X_{ij}} & \text{Jika j adalah atribut biaya} (cost) \end{cases} \qquad (1)$$

Keterangan :

r_{ij} = nilai rating kinerja ternormalisasi

x_{ij} = nilai atribut yang dimiliki dari setiap kriteria

$\text{Max}_i\, x_{ij}$ = nilai terbesar dari setiap kriteria

$\text{Min}_i\, x_{ij}$ = nilai terkecil dari setiap kriteria

benefit = jika nilai terbesar adalah terbaik

cost = jika nilai terkecil adalah terbaik

dimana r_{ij} adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j; i=1,2,...,m dan j=1,2,...,n. Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai:

$$V_i = \sum_{j=1}^{n} w_j\, r_{ij} \qquad (2)$$

Keterangan :

V_i = rangking untuk setiap alternatif

w_j = nilai bobot dari setiap kriteria

r_{ij} = nilai rating kinerja ternormalisasi

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih.

2.2.10 Algoritma Penyelesaian

Dalam penelitian ini menggunakan metode FMADM dengan metode SAW. Adapun langkah-langkahnya adalah:

1. Memberikan nilai setiap alternatif (A_i) pada setiap kriteria (C_j) yang sudah ditentukan, dimana nilai i=1,2,…m dan j=1,2,…n.

2. Memberikan nilai bobot (W) yang juga didapatkan berdasarkan nil ai *crisp*.
3. Melakukan normalisasi matriks dengan cara menghitung nilai *rating* kinerja ternormalisasi (r_{ij}) dari alternatif A_i pada atribut C_j berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan/*benefit*=MAKSIMUM atau atribut biaya/*cost*=MINIMUM). Apabila berupa artibut keuntungan maka nilai *crisp* (X_{ij}) dari setiap kolom atribut dibagi dengan nilai *crisp* MAX (MAX X_{ij}) dari tiap kolom, sedangkan untuk atribut biaya, nilai *crisp* MIN (MIN X_{ij}) dari tiap kolom atribut dibagi dengan nilai *crisp* (X_{ij}) setiap kolom.
4. Melakukan proses perankingan untuk setiap alternatif (V_i) dengan cara mengalikan nilai bobot (w_i) dengan nilai *rating* kinerja ternormalisasi (r_{ij}).

2.2.11 Basis Data

Database adalah kumplan data yang saling berkaitan, berhubungan yang disimpan secara bersama-sama sedemikian rupa tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan **FAT[1]** . Data-data ini harus mengandung semua informasi untuk mendukung semua kebutuhan sistem. Proses dasar yang dimiliki oleh *database* ada empat, yaitu:

1. Pembuatan data-data baru *(create database)*
2. Penambahan data *(insert)*
3. Mengubah data *(update)*
4. Menghapus data *(delete)*.

Database merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi pada para pengguna. Basis data (*database*) menjadi penting karena munculnya beberapa masalah bila tidak menggunakan data yang terpusat, seperti adanya duplikasi data, hubungan antar data tidak jelas, organisasi data dan *update* menjadi rumit. Jadi tujuan dari pengaturan data dengan menggunakan basis data adalah :

a. Menyediakan penyimpanan data untuk dapat digunakan oleh organisasi saat sekarang dan masa yang akan datang.
b. Cara pemasukan data sehingga memudahkan tugas operator dan menyangkut pula waktu yang diperlukan oleh pemakai untuk mendapatkan data serta hak-hak yang dimiliki terhadap data yang ditangani.
c. Pengendalian data untuk setiap siklus agar data selalu *up-to-date* dan dapat mencerminakan perubahan spesifik yang terjadi di setiap sistem.
d. Pengamanan data terhadap kemungkinan penambahan, modifikasi, pencurian dan gangguan-gangguan lain.

Dalam basis data sistem informasi digambarkan dalam model *entity relationship* (E-R). Bahasa yang digunakan dalam basis data (*database*) yaitu :

a. DDL (*Data Definition Language*)

   Merupakan bahasa definisi data yang digunakan untuk membuat dan mengelola objek *database* seperti *database*, tabel dan *view*
b. DML (*Data Manipulation Language*)

Merupakan bahasa manipulasi data yang digunakan untuk memanipulasi data pada objek *database* seperti tabel

c. DCL (*Data Control Language*)

Merupakan bahasa yang digunakan untuk mengendalikan pengaksesan data.

Penyusunan basis data meliputi proses memasukkan data kedalam media penyimpanan data, dan diatur dengan menggunakan perangkat Sistem Manajemen Basis Data (*Database Management System / DBMS*).

2.2.12 DBMS (*Database Management System*)

Sistem Manajemen Basis data *(Database Management System)* merupakan sistem pengoperasian dan sejumlah data pada computer **FAT[1]**. 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 da ta 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 *databse*. Dalam sebuah organisasi berskala besar, *hardware* terdiri : jaringan dengan sebuah *server* pusat dan beberapa program *client* yang berjalan di komputer desktop.

2. *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 pros edur 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 u*ser* 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. *Enduser* adalah orang yang berada di depan *workstatio*n dan berinteraksi secara langsung dengan sistem.
c. *Programmer aplikasi*, orang yang berinteraksi dengan *database* melalui cara yang berbeda.

2.2.13 ERD (*Entity Relationship Diagram*)

Setelah semua memori esensial ditransformasikan menjadi *object data store*, maka dapat disusun relasi antara satu *object data store* dengan yang lain **FAT[1]**. *Enttity Relationship* (E-R) *Diagram* dibuat untuk memperlihatkan relasi itu. *Enttity Relationship Diagram* adalah ilustrasi grafis objek-objek (atau *entities*)

dan atribut, serta relasi antara keduanya. Diagram E-R diperkenalkan oleh Peter Chen dan sifatnya *independent* terhadap teknologi *database* yang digunakan.

Salah satu metode pembuatan diagram E-R, yaitu dengan menggambarkan himpunan entitas sebagai persegi panjang, himpunan relasi sebagai belah ketupat, atribut sebagai lingkaran (atribut yang berfungsi sebagai *key* digarisbawahi), dan notasi yang lain adalah berupa gasis lurus yang dinyatakan dengan banyaknya garis cabang atau dengan pemakaian angka menerangkan tentang kardinalitas relasi.

E-R Diagram merupakan suatu bahasa pemodelan yang dimana posisinya dapat dianalogikan dengan *story board* dalam industri film, *blu print* arsitektur suatu bangunan, miniatur, dan lain-lain. Dalam praktiknya, membangun suatu sistem terlebih dahulu dilakukannya suatu perencaan. Pemodelan merupakan suatu sub bagian dari perencanaan secara keseluruhan sebagai salah satu upaya *feedback* evaluasi perampungan suatu perencanaan. E-R Diagram sebagai suatu pemodelan setidaknya memiliki beberapa karakteristik dan manfaat sebagai berikut:

a. Memudahkan untuk dilakukannya analisis dan perubahan sistem sejak dini, bersifat murah dan cepat.
b. Memberikan gambaran umum akan sistem yang akan di buat sehingga memudahkan developer.
c. Menghasilkan dokumentasi yang baik untuk *client* sebagai bahan diskusi dengan bentuk E-R Diagram itu sendiri, dan
d. Kamus data bagi bagi para pengembang *database*.

Struktur dari E-R Diagram secara umum ialah terdiri dari:

a. Entitas merupakan objek utama yang informasi akan disimpan, biasanya berupa kata benda. Contoh: mahasiswa, dosen, nasabah, mata kuliah, ruangan, dan lain-lain. Objek dapat berupa benda nyata maupun abstrak.
b. Atribut merupakan deskripsi dari objek yang bersangkutan.
c. *Relationship* merupakan suatu hubungan yang terjalin antara dua entitas yang ada.

Kardinalitas relasi ERD yang mempersentasikan suatu basisdata relasional senantiasa memiliki relasi-relasi dari sejumlah entitas yang dapat ditentukan banyaknya. Banyaknya suatu relasi yang dimiliki oleh suatu relasi entitas disebut derajat relasi. Derjat relasi maksimum disebut dengan kardinalitas sedangkan derajad minimum disebut dengan modalitas. Kardinalitas yang terjadi diantara dua himpunan entitas (misal A dan B) dapat berupa :

a. *One to One* (1-1), relasi yang terjadi jika sebuah *entry* dalam sebuah *object data sore* dihubungkan dengan hanya sebuah *entry* dalam *object data store* yang lain.
b. *One to Many* (1-M), relasi yang terjadi jika sebuah *entry* dalam sebuah *object data store* dihubungkan dengan satu atau lebih *entry* dalam *object data store* yang lain.
c. *Many to Many* (M-M), relasi yang terjadi jika satu atau lebih *entry* dalam sebuah *object data store* dihubungkan dengan satu atau lebih *entry* dalam *object data store.*

Dalam semua definisi relasi di atas, “1” menyatakan hanya satu, dan “M” menyatakan satu atau lebih. Jika suatu saat dibutuhkan pernyataan relasi paling sedikit sebuah dan juga lebih dari satu, maka relasi itu dapat dinyatakan dengan “M*” (* = *minimum of one*).

2.2.14 DFD *(Data Flow Diagram*)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut **FAT[1]**.

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik diman a data tersebut mengalir atau dimana data tersebut akan disimpan.

DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur. Kelebihan utama pendekatan aliran data, yaitu :

a. Kebebasan dari menjalankan implementasi teknis siste m.
b. Pemahaman lebih jauh mengenai keterkaitan satu sama lain dalam sistem dan subsistem.
c. Mengkomunikasikan pengetahuan sistem yang ada dengan pengguna melalui diagram aliran data.
d. Menganalisis sistem yang diajukan untuk menentukan apakah data -data dan proses yang diperlukan sudah ditetapkan.

Disamping itu terdapat kelebihan tambahan, yaitu :

1. Dapat digunakan sebagai latihan yang bermanfaat bagi penganalisis, sehingga bisa memahami dengan lebih baik keterkaitan satu sama lain dalam sistem dan subsistem.
2. Membedakan sistem dari lingkungannya dengan menempatkan batas - batasnya.
3. Dapat digunakan sebagai suatu perangkat untuk berinteraksi dengan pengguna.
4. Memungkinkan penganalisis menggambarkan setiap komponen yang digunakan dalam diagram.

DFD terdiri dari *context diagram* dan diagram rinci *(DFD Levelled)*. *Context diagram* berfungsi memetakan model lingkungan (menggambarkan hubungan antara entitas luar, masukan dan keluaran sistem), yang direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem. *DFD levelled* menggambarkan sistem sebagai jaringan kerja antara fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data, model ini hanya memodelkan sistem dari sudut pandang fungsi.

Dalam DFD *levelled* akan terjadi penurunan level dimana dalam penurunan level yang lebih rendah harus mampu merepresentasikan proses tersebut ke dalam spesifikasi proses yang jelas. Jadi dalam DFD levelled bisa dimulai dari DFD level 0 kemudian turun ke DFD level 1 dan seterusnya. Setiap penurunan hanya dilakukan bila perlu. Aliran data yang masuk dan keluar pada suatu proses di level x harus berhubungan dengan aliran data yang masuk dan

keluar pada level x+1 yang mendefinisikan proses pada level x tersebut. Proses yang tidak dapat diturunkan/dirinci lagi dikatakan primitif secara fungsional dan disebut sebagai proses primitif.

2.2.15 Kamus Data

Kamus Data merupakan katalog (tempat penyimpanan) dari elemen-elemen yang berada dalam satu sistem **FAT[1]**. Kamus data mempunyai fungsi yang sama dalam pemodelan sistem dan juga berfungsi membantu pelaku sistem untuk mengerti aplikasi secara detail, dan me-reorganisasi semua elemen data yang digunakan dalam sistem sehingga pemakai dan penganalisa sistem punya dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses. Kamus data mendefinisikan elemen data dengan fungsi sebagai berikut :

1. Menjelaskan arti aliran data dan penyimpanan dalam DFD
2. Mendeskripsikan komposisi paket data yang bergerak melalui aliran misalnya alamat diuraikan menjadi kota, negara dan kode pos
3. Mendeskripsikan komposisi penyimpanan data
4. Menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran data.
5. Mendeskripsikan hubungan detail antar penyimpanan yang akan menjadi titik perhatian dalam Diagram Keterhubungan Entitas (E-R)

2.2.16 SQL Server 2000

Microsoft SQL Server 2000 merupakan program aplikasi *database* untuk *client-server* **ILM[13]**. Program ini dapat beroperasi pada berbagai sistem operasi, mulai dari Windows 9x, Windows 2000 dengan berbagai macam versinya dan juga Windows NT. Selama beroperasi, dimana SQL Server 2000 ini mengolah *database*, SQL Server dalam berhubungan dengan *user* akan dikelola oleh *Database Administrator.* Sistem SQL Server dapat diimplementasikan sebagai sistem *clent-server* ataupun *stand alone desktop system*.

Microsoft SQL Server 2000 merupakan sebuah program RDBMS (*Relational Database Management System*) yang menyediakan dukungan untuk mengorganisasi data dengan cara meyimpannya ke dalam tabel *database*. Hubungan antar data dapat dikelompokkan ke dalam tabel, dan keterhubungan dapat didefinisikan antar tabel-tabel yang disebut dengan *Relational Database*. SQL Server merupakan *database server*. *User* dapat berhubungan dengan *database* yang ada pada *server* melalui sebuah aplikasi, sedangkan *Administrator* dapat mengakses data secara langsung untuk mengelolanya.

TRANSACT-SQL adalah bahasa pemrograman yang dikembangkan dari SQL dengan alur program pemrograman melaui "*request*" dan "*response*". Melalui instruksi SQL, dapat dilakukan *query* atatu transaksi, yang kemudian akan menerima jawaban dari *Database Server* berupa hasil atau *ResultSet*. TRANSACT-SQL adalah sintaks SQL yang dijalankan di lingkungan Microsoft SQL Server.

TRANSACT-SQL dimulai dengan deklarasi variabel dan dilanjutkan dengan bagian blok program. Variabel harus dideklarasikan sebelum digunakan. Nama variabel selalu dimulai dengan karakter @. Variabel dapat diberikan nilai melalui instruksi SELECT. Penulisan nama variabel tidak membedakan huruf besar dan huruf kecil, dianggap sama keduanya. TRANSACT SQL menyediakan DDL dan DML yang sesuai dengan standar SQL, ditambah dengan *extended function*, sistem prosedur tersimpan, dan struktur pemrograman (seperti *if* dan *while*) untuk memberikan fleksibilitas dalam pemrograman. Penggunaan *statement-statement* dari TRANSACT-SQL dilakukan melalui *Query Analyzer*.

```
DECLARE
/* Deklarasi variabel */
@kode int
```

Beberapa contoh sintaks yang digunakan dalam Microsoft SQL Server 2000 adalah :

1. Membuat Tabel

Sintaks T-SQL :

```
USE MyData
GO
CREATE TABLE mahasiswa
(
        NIM         NIM_type      NOT NULL,
        nama        varchar (20),
        alamat      varchar (30),
        tmp_lahir   varchar (20),
```

```
    tgl_lahir       datetime,
    CONSTRAINT PK_NIM PRIMARY KEY (NIM)
                                    )
    GO
```

2. Memasukkan data

   Sintaks dasar : INSERT [INTO] *table_name* [(*column_list*)]
   VALUES
   (*value*)

   Sintaks T-SQL : USE MyData
   INSERT [INTO] *table_name* [(*column_list*)]
   VALUES
   (*value*)
   GO

3. Meng-*update* data

   Sintaks dasar : UPDATE *table_name* SET *column_name = value*
   WHERE *search_condition*

   Sintaks T-SQL : USE MyData
   UPDATE *table_name* SET *column_name = value*
   WHERE *search_condition*
   GO

4. Menghapus data

   Sintaks dasar : DELETE FROM *table_name* | *View_name*
   WHERE *search_condition*

   Sintaks T-SQL : USE MyData
   DELETE FROM *table_name* | *View_name*
   WHERE *search_condition*
   GO

5. Melihat data

Sintaks dasar : SELECT * FROM *table_name*

Sintaks T-SQL : USE MyData

SELECT * FROM *table_name*

GO

2.2.17 Borland Delphi

Borlan Delphi adalah suatu program berbasis bahasa Pascal yang berjalan dalam lingkungan *Windows* **KAD[3]**. Delphi telah memanfaatkan suatu teknik pemrograman yang disebut RAD yang telah membuat pemrograman menjadi lebih mudah. Delphi adalah suatu bahasa pemrograman yang telah memanfaatkan metode pemrograman *Object Oriented Programming* (OOP). Secara ringkas, *object* adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat (*visual*). *Object* biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Bahasa pemrograman secara singkat dapat disebut sebagai sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu.

Khusus untuk pemrograman *database*, Delphi menyediakan objek yang sangat kuat, canggih, dan lengkap, sehingga memudahkan pemrogram dalam merancang, membuat dan menyelesaikan aplikasi *database* yang diinginkan. Selain itu, Delphi juga dapat menangani data dalam berbagai format *database*, misalnya format MS-Access, SyBase, Oracle, FoxPro, Informix, DB2, MySQL dan lain-lain. Format *database* yang dianggap asli dari Delphi adalah Paradox dan dBase.