

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Sistem**

Menurut ( Kusrini ; 2007 : 11) Sistem merupakan elemen kumpulan elemen yang saling berkaitan yang bertanggung jawab masukan( input) sehingga menghasilkan keluaran (output).

Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian,mendukung operasi, bersifat manajerial dan merupakan kegiatan strategi dari suatu organisasi,serta menyediakan laporan –laporan yang diperlukan oleh pihak luar.

Berdasarkan dukungan kepada pemakainya,sistem informasi dibagi menjadi 7 yaitu sebagai berikut:

1. Sistem Pemrosesan Transaksi ( *Transaction Processing System* ).
2. Sistem Informasi Manajemen ( *Management Information System* ).
3. Sistem Otomasi Perkantoran ( *Office Automation System* ).
4. Sistem Pendukung Keputusan ( *Desicion Support System* ).
5. Sistem Informasi Eksekutif ( *Excecutive Support System* ).
6. Sistem Informasi Kelompok ( *Group Support System* ).
7. Sistem Pendukung Cerdas ( *Intellegent Support System* ).

## **II.2. Pengambilan Keputusan**

Menurut ( Kusrini ; 2007 : 9 ) Pengambilan keputusan merupakan tahap-tahap yang harus dilalui atau digunakan untuk membuat keputusan. Tahap-tahap ini merupakan kerangka dasar, sehingga setiap tahap dapat dikembangkan lagi menjadi beberapa sub tahap (disebut langkah) yang lebih khusus/spesifik dan lebih operasional. Secara umum, proses pengambilan keputusan terdiri atas beberapa tahap yaitu sebagai berikut :

1. Identifikasi alternatif-alternatif keputusan untuk memecahkan masalah
2. Perhitungan mengenai faktor-faktor yang tidak dapat diketahui sebelumnya atau di luar jangkauan manusia, identifikasi peristiwa-peristiwa di masa datang (*state of nature*).
3. Pengumpulan data yang di butuhkan untuk melaksanakan model keputusan.
4. Pemilihan dan penggunaan model pengambilan keputusan.
5. Melaksanakan solusi terpilih.

### **II.2.1. Masalah**

Menurut ( Kusrini ; 2007 : 7) Masalah merupakan suatu kondisi yang berpotensi menimbulkan kerugian luar biasa atau menghasilkan keuntungan luar biasa. Tindakan memberi respons terhadap masalah untuk menekan akibat buruknya atau memanfaatkan peluang keuntungannya disebut pemecah masalah. Pentingnya pemecah masalah bukan didasarkan pada jumlah waktu yang dihabiskan, tetapi konsekuensinya yaitu apakah

pemecahan masalah tersebut bisa menekan sebanyak mungkin kemungkinan kerugian atau memperoleh sebesar mungkin keuntungan.

### **II.2.2. Keputusan**

Menurut (Kusrini ; 2007 : 7) Keputusan merupakan kegiatan memilih suatu strategi atau tindakan dalam pemecahan masalah tersebut. Tindakan memilih strategi tersebut memberikan solusi terbaik atas sesuatu itu disebut pengambilan keputusan. Tujuan dari keputusan adalah untuk mencapai target atau aksi tertentu yang harus dilakukan. Ciri-ciri dari keputusan adalah sebagai berikut:

1. Banyak pilihan/alternatif
2. Ada kendala atau syarat
3. Mengikuti suatu pola/model tingkah laku, baik yang terstruktur maupun yang tidak terstruktur.
4. Banyak input.
5. Ada faktor resiko.
6. Dibutuhkan kecepatan, ketepatan dan keakuratan.

### **II.3. Sistem Pendukung Keputusan**

Menurut (Kusrini ; 15 : 2007) Sistem Pendukung Keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan dan situasi

yang tidak terstruktur, di mana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya di buat.

Keputusan yang diambil untuk menyelesaikan suatu masalah di lihat dari keterstrukturannya yang bisa di bagi menjadi :

1. Keputusan terstruktur

Keputusan terstruktur adalah keputusan yang di lakukan secara berulang-ulang dan bersifat rutin. misalnya: keputusan pemesanan barang.

2. Keputusan semi terstruktur

Keputusan semiterstruktur adalah keputusan yang memiliki dua sifat, sebagian keputusan bisa di tangani oleh komputer dan yang lain tetap harus di lakukan oleh pengambil keputusan. misalnya : pengevaluasian kredit dan penjadwalan produk.

- 3 Keputusan tak terstruktur

Keputusan tak struktur adalah keputusan yang penangannya rumit karena tidak terjadi berulang-ulang atau tidak selalu terjadi. misalnya: keputusan untuk pengembangan teknologi baru.

#### **II.4. Anggota DPRD**

Dewan Perwakilan Rakyat Daerah atau disingkat DPRD adalah lembaga perwakilan rakyat daerah yang melaksanakan fungsi-fungsi pemerintah daerah sebagai mitra sejajar pemerintah daerah. Dalam Struktur pemerintahan daerah, DPRD berada di dua jenjang yaitu di tingkat

propinsi disebut DPRD Propinsi serta di tingkat kabupaten/kota disebut DPRD kabupaten/kota. Adapun tugas dan wewenang DPRD adalah sebagai berikut :

- a. Membentuk peraturan daerah kabupaten bersama Kepala Daerah.
- b. Membahas dan memberikan persetujuan rancangan peraturan daerah mengenai anggaran pendapatan dan belanja daerah kabupaten yang diajukan oleh Kepala Daerah.
- c. Melaksanakan pengawasan terhadap pelaksanaan peraturan daerah dan anggaran pendapatan dan belanja daerah kabupaten.
- d. Mengusulkan pengangkatan dan pemberhentian Kepala Daerah dan/atau wakil Kepala Daerah kepada Menteri Dalam Negeri melalui gubernur untuk mendapatkan pengesahan pengangkatan atau pemberhentian.
- e. Meminta laporan keterangan pertanggungjawaban Kepala Daerah dalam penyelenggaraan pemerintahan daerah kabupaten.
- f. Mengupayakan terlaksananya kewajiban daerah sesuai dengan ketentuan peraturan perundang-undangan.

## **II.5. Metode SAW**

SAW (*Simple Additive Weight*) sering juga dikenal dengan istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari kinerja setiap alternatif pada semua atribut.

Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat dibandingkan dengan semua rating alternatif yang ada. Metode ini merupakan metode yang paling terkenal dan paling banyak digunakan dalam menghadapi situasi *Multiple Attribute Decision Making* (MADM). MADM itu sendiri merupakan suatu metode yang digunakan untuk mencari alternatif dengan kriteria tertentu.

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max x_{ij}} & \text{jika } j \text{ adalah atribut keberuntungan (benefit)} \\ \frac{\min x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases}$$

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

Dimana :

$r_{ij}$  = rating kerja ternormalisasi.

$\max_i$  = nilai maksimum dari setiap baris dan kolom.

$\min_i$  = nilai minimum dari setiap baris dan kolom.

$X_{ij}$  = baris dan kolom dari matriks.

$(r_{ij})$  adalah rating kinerja ternormalisasi dari alternatif  $(A_i)$  pada atribut  $(C_j)$   $i= 1,2,\dots,$  dan  $j= 1,2,\dots,n$ .

$$V_i = \sum_{j=1}^n w_j r_{ij}$$

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

Dimana :

$V_i$  = nilai akhir dari alternatif.

$W_j$  = bobot yang telah ditentukan.

$r_{ij}$  = normalisasi matriks nilai yang lebih besar mengindikasikan bahwa alternatif lebih terpilih.

Ada beberapa langkah dalam penyelesaian metode *Simple Additive Weighting (SAW)*. Yang diterapkan sebagai berikut :

1. Menentukan kriteria-kriteria yang dijadikan acuan dalam pendukung keputusan yaitu  $C_i$ .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria ( $C_i$ ).
4. Kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan ataupun atribut biaya) sehingga diperoleh matriks ternormalisasi  $R$ .

5. Hasil akhir diperoleh dari proses perangkingan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vector bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik ( $A_i$ ) sebagai solusi.

## **II.6. PHP**

Menurut ( Rulianto Kurniawan ; 2010 : 2 ) PHP adalah script yang digunakan untuk membuat halaman web yang dinamis. dinamis berarti halaman yang akan di tampilkan dibuat saat halaman itu diminta oleh client. mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru. semua script PHP dieksekusi pada server dimana script tersebut di jalankan.

### **II.6.1. Kelebihan PHP**

Banyak sekali kelebihan-kelebihan yang dimiliki PHP dibandingkan dengan bahasa pemrograman lainnya di antaranya:

1. Bisa membuat web menjadi dinamis.
2. PHP bersifat open Source sehingga dapat digunakan oleh semua orang secara gratis.
3. PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
4. Aplikasi dengan PHP cukup cepat dibandingkan dengan aplikasi CGI bahkan lebih cepat dengan ASP maupun JAVA dalam berbagai aplikasi web.

5. Web server yang mendukung PHP dapat dapat ditemukan dimana-mana mulai dari *Apache,IIS* hingga *xitami* dengan konfigurasi yang relatif mudah.

### II.6.2. Kekurangan PHP

Selain kelebihan *PHP*,*PHP* juga mempunyai beberapa kekurangan.

Adapun beberapa kekurangan diantaranya adalah sebagai berikut:

1. Permasalahannya yang sering terjadi adalah padad register\_globals.
2. Tidak mengenal packages.
3. Jika tidak ada diencoding,maka kode PHP dapat dibaca semua orang.Untuk encodingnya anda membutuhkan tool dan zend dan biaya encoding yang sangat mahal.
4. Tidak memiliki suatu sistem pemrograman berorientasi objek yang sesungguhnya.
5. PHP memiliki kelemahan *security*( keamanan ).

### II.7. MY SQL

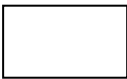
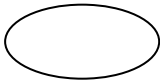
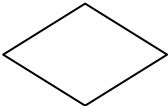

Menurut ( Rulianto kurniawan ; 2010 : 148 ) MySQL adalah salah satu jenis database server yang terkenal,MySQL termasuk jenis salah satu RDBMS (*Relational Database Managemen System* ).MySQL adalah sebuah perangkat lunak database (basis data) sistem terbuka yang sangat terkenal dikalangan pengembang sistem *database* dunia yang di gunakan untuk berbagai aplikasi terutama untuk aplikasi berbasis web. *MySQL* umumnya

digunakan bersamaan dengan *PHP* untuk membuat aplikasi yang dinamis dan *powerfull*.

## II.8. ERD

ERD (*Entity Relationship Diagram*) adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol. Pada dasarnya ada empat notasi yang digunakan dapat dilihat pada tabel II.1 berikut ini :

**Tabel II.1 Notasi dalam Diagram ER**

Notasi	Keterangan
	Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	Relasi menunjukkan adanya hubungan sejumlah entitas yang berbeda.
	Atribut berfungsi mendeskripsikan karakter entitas (atribut yang berfungsi sebagai key diberi garis bawah)
	Garis adalah penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut

*Sumber : Adi nugroho ; 2010 : 9*

a) Entiti

Entiti merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari entiti ini biasanya digambarkan dengan persegi panjang.

b) Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol elips.

c) Hubungan/Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Relasi dapat digambarkan sebagai berikut:

1. Satu ke satu (One to one)

Hubungan relasi satu ke satu yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B.

2. Satu ke banyak (One to many)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A.

### 3. Banyak ke banyak (Many to many)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B.

## II.9. UML

UML adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan mereka dengan yang lain.



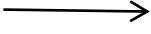
Dalam konteks UML, tahap konseptualisasi dilakukan dengan pembuatan usecase diagram yang sesungguhnya merupakan deskripsi peringkat tinggi bagaimana perangkat lunak akan digunakan oleh penggunanya, selanjutnya use case diagram tidak hanya sangat penting pada tahap analisis, tetapi juga sangat penting untuk perancangan (*design*), untuk mencari kelas-kelas yang terlibat dalam aplikasi dan untuk melakukan pengujian (*testing*).

Adapun beberapa alat bantu lain yang sangat sering di pakai oleh sistem analis atau perancang yaitu sebagai berikut:

### II.9.1. Use Case Diagram

Use case adalah abstraksi dari interaksi antara system dan actor. Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. Usecase merupakan konstruksi untuk mendeskripsikan bagaimana system akan terlihat di mata user. Sedangkan use case diagram memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan client. Adapun simbol Usecase diagram dapat dilihat pada table II.2 berikut ini :

**Tabel II.2 Simbol Use Case**

Simbol	Nama	Keterangan
	<i>Actor</i>	Menggambarkan pengguna software aplikasi ( user )
	<i>Usecase</i>	Menggambarkan fungsionalitas dari suatu sistem, sehingga customer atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun
	<i>Association</i>	Menggambarkan hubungan antara actor dengan use case

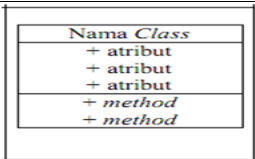




*Sumber : Munawar ; 2007 : 64*

### II.9.2. Class Diagram

*Class* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti)

suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). Adapun symbol class diagram dapat dilihat pada table II.3 berikut ini:

**Tabel II.3 Simbol Class Diagram**



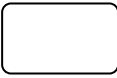

Symbol	Nama	Keterangan
	<b>Class</b>	blok-blok pembangun pada pemrograman berorientasi obyek
	<b>Dependency</b>	untuk menunjukkan operasi pada suatu class yang menggunakan class yang lain
	<b>Composition</b>	Jika sebuah class tidak bisa berdiri sendiri dan harus merupakan bagian dari class yang lain, maka class tersebut memiliki relasi Composition terhadap class tempat dia bergantung tersebut.
	<b>Aggregation</b>	mengindikasikan keseluruhan bagian relationship dan biasanya disebut sebagai relasi.
	<b>Association</b>	Sebuah asosiasi merupakan sebuah relationship paling umum antara 2 class dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 class

*Sumber : Munawar ; 2007 : 20*

### II.9.3. Activity Diagram

Activity Diagram adalah teknik untuk mendeskripsikan logika prosedural ,proses bisnis dan aliran kerja.Dalam banyak kasus activity diagram mempunyai peran seperti halnya flowchart,akan tetapi perbedaannya dengan flowchart adalah activity diagram bisa mendukung perilaku paralel sedangkan flowchart tidak bisa.Adapun simbol activity diagram dapat di lihat pada table II.4 berikut ini :

**Tabel II.4 Simbol Activity Diagram**

<b>Simbol</b>	<b>Nama</b>	<b>Keterangan</b>
	<i>Start State</i>	Menggambarkan awal dari suatu aktivitas yang berjalan pada sistem
	<i>End State</i>	Menggambarkan akhir dari suatu aktivitas yang berjalan pada sistem
	<i>Activity</i>	Menggambarkan aktivitas yang dilakukan pada sistem
	<i>Transition State</i>	Menggambarkan hubungan antara dua state, dua activity ataupun antara state dan activity.

*Sumber : Munawar ; 2007 : 110*





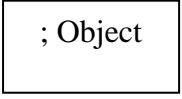
### II.9.4. Sequence Diagram

Sequence diagram digunakan untuk menggunakan perilaku pada sebuah skenario.Diagram ini menunjukkan sejumlah contoh objek dan pesan yang diletakan di antara objek-objek ini kedalam use case. Komponen utama sequence terdiri atas objek yang dituliskan dengan kotak

segi empat bernama message .Message diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan progress vertical.

Sequence diagram menambahkan dimensi waktu lain pada interaksi diantara objek.Pada Diagram ini *lifeline* diurutkan dari setiap participant.Kotak kecil pada *lifeline* menyatakan activation yaitu menjalankan salah satu operation dari participant .Adapun simbol Sequence diagram dapat di lihat pada table II.5 berikut ini :

**Tabel II.5 Simbol Sequence Diagram**

Simbol	Nama	Keterangan
	<i>Actor</i>	Menggambarkan seseorang atau sesuatu (seperti perangkat,sistem lain) yang berinteraksi dengan sistem
	<i>Lifeline</i>	Mengindikasikan keberadaan sebuah object dalam basis waktu. Notasi untuk Lifeline adalah garis putus-putus vertikal yang ditarik dari sebuah obyek.
	<i>Activation</i>	Sebuah kotak segi empat yang digambar pada sebuah lifeline. Activation mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.
	<i>Message</i>	Digambarkan dengan anak panah horizontal antara Activation. Message mengindikasikan komunikasi antara object-object.
	<i>Object</i>	Merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal.Digambarkan sebagai sebuah class (kotak) dengan nama obyek didalamnya yang diawali dengan sebuah titik koma.

Sumber : Munawar ; 2007 : 87