

## BAB II

### LANDASAN TEORI

#### 2.1 Guru

Guru merupakan pendidik profesional dengan tugas utama mendidik, mengajar, membimbing, mengarahkan, melatih, menilai dan mengevaluasi peserta didik pada jalur pendidikan formal. tugas utama itu akan efektif jika guru memiliki derajat profesionalitas tertentu yang tercermin dan kompetensi, kemahiran, kecakapan atau ketrampilan yang memenuhi standart mutu atau norma etik tertentu. Secara formal untuk menjadi guru profesional, guru disyaratkan memenuhi kualifikasi akademik minimum dan bersertifikat mendidik. (Ali Mudlofir : 2014)

#### 2.2 Database

*Database* merupakan kumpulan informasi yang disimpan di dalam komputer secara sistematis untuk memperoleh informasi dari basis data tersebut. *Database* adalah kumpulan data-data dalam bentuk tabel yang saling berhubungan yang disimpan dalam media perangkat keras (contohnya: harddisk) yang dapat diambil lagi sebagai informasi. Eko Koswara (2011:95)

#### 2.3 Microsoft Visual Basic .Net

*Visual Basic .Net* adalah visual basic yang direkayasa kembali untuk digunakan pada *platform.Net* sehingga aplikasi yang dibuat menggunakan *Visual Basic .Net* dapat berjalan pada sistem komputer apapun, dan dapat mengambil data dari server dengan tipe apapun asalkan terinstal *.Net framework*. (Priyanto: 2014)

*Visual Studio 2008* sebuah perangkat lunak buatan *Microsoft* yang berfungsi untuk membuat program berbasis *Desktop* maupun berbasis *Web*. Pada pemrograman *visual*, pengembangan aplikasi dimulai dengan pembentukan *user interface*, kemudian mengatur properti dari objek-objek yang digunakan dalam *user interface*, dan baru dilakukan penulisan kode program untuk menangani kejadian-kejadian(*event*). Tahap pengembangan aplikasi demikian dikenal dengan istilah pengembangan aplikasi dengan pendekatan *Botton Up*.

## 2.4 MySQL

*MySQL* adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi berbasis *web* yang menggunakan *database* sebagai sumber dan pengolahan datanya. (Arief, 2011).

*MySQL* berfungsi untuk mengolah database dengan menggunakan bahasa *SQL*. *MySQL* bersifat *open source* sehingga bisa digunakan secara gratis. *SQLyog* adalah salah satu *tool* administrasi untuk *database MySQL*. Jika kita biasanya menggunakan *PhpMyAdmin* yang *include* di dalam aplikasi *Xampp* untuk melakukan administrasi *database*, *SQLyog* merupakan aplikasi alternatif untuk melakukan proses administrasi *database MySQL*

## 2.5 Sistem Pendukung Keputusan

*Decision Support System* atau Sistem Pendukung Keputusan adalah sebuah sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah dengan

kondisi semi terstruktur dan tak terstruktur. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat. (Turban, 2011)

Sistem Pendukung Keputusan sebagai sebuah sistem berbasis komputer yang membantu dalam proses pengambilan keputusan. SPK sebagai sistem informasi berbasis komputer yang adaptif, interaktif dan fleksibel yang secara khusus dikembangkan untuk mendukung solusi dari permasalahan manajemen yang tidak terstruktur untuk meningkatkan kualitas pengambilan keputusan.

a. Konsep SPK

SPK adalah sistem yang dapat dikembangkan, mampu mendukung analisis data dan pemodelan keputusan, berorientasi pada perencanaan serta tidak bisa direncanakan interval (periode) waktu pemakaiannya. Sistem Pendukung Keputusan terdiri dari 3 komponen yang berinteraksi satu dengan yang lainnya, yaitu :

- 1) *Language System*, adalah suatu mekanisme untuk menjembatani pemakai dan komponen lainnya.
- 2) *Knowledge System*, adalah repository pengetahuan yang berhubungan dengan masalah tertentu baik berupa data maupun prosedur.
- 3) *Problem Processing System*, adalah sebagai penghubung kedua komponen lainnya, berisi satu atau

beberapa kemampuan manipulasi atau menyediakan masalah secara umum yang diperlukan dalam pengambilan keputusan.

b. Karakteristik SPK

Karakteristik dari Sistem Pendukung Keputusan yang membedakan dari sistem lainnya adalah:

- 1) SPK dirancang untuk membantu pengambilan keputusan dalam memecahkan masalah yang sifatnya semi terstruktur maupun tidak terstruktur.
- 2) Dalam proses pengolahannya, SPK mengkombinasikan menggunakan model-model analisis dengan teknik pemasukan data konvensional serta fungsi-fungsi pencari.
- 3) SPK dirancang sedemikian rupa sehingga dapat digunakan dengan mudah oleh orang-orang yang tidak memiliki dasar kemampuan pengoperasian komputer yang tinggi.
- 4) SPK dirancang dengan menekankan pada aspek fleksibilitas serta kemampuan adaptasi yang tinggi. Sehingga mudah disesuaikan dengan berbagai perubahan lingkungan yang terjadi dan kebutuhan pemakai.

### c. Keuntungan Sistem Pendukung Keputusan

Keuntungan dari Sistem Pendukung Keputusan antara lain:

- 1) Memperluas kemampuan pengambilan keputusan dalam memproses data/informasi bagi pemakainya.
- 2) Membantu pengambilan keputusan dalam hal penghematan waktu yang dibutuhkan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
- 3) Dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.

## 2.6 *Simple Addictive Weighting*

Metode *Simple Addictive Weighting* (SAW) sering juga dikenal dengan istilah metode penjumlahan berbobot. Konsep dasar metode *Simple Addictive Weighting* adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. (Pahlevy 2010)

Metode *Simple Addictive Weighting* membutuhkan proses normalisasi matriks keputusan ( $x$ ) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Metode *Simple Addictive Weighting* merupakan metode yang paling terkenal dan paling banyak digunakan dalam menghadapi situasi *Multiple Attribute Decission Making*.

Metode *Simple Addictive Weighting* ini mengharuskan pembuat keputusan menentukan bobot bagi setiap atribut. Skor total untuk alternatif diperoleh dengan menjumlahkan seluruh hasil perkalian antara rating

(yang dapat dibandingkan lintas atribut) dan bobot tiap atribut. Rating tiap atribut haruslah bebas dimensi dalam arti telah melewati proses normalisasi matriks sebelumnya.

Formula untuk melakukan normalisasi adalah:

$$R_{ij} = \begin{cases} \frac{x_{ij}}{\max_i x_{ij}} & \text{Jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\min_i x_{ij}}{x_{ij}} & \text{Jika } j \text{ adalah atribut biaya (cost)} \end{cases} \quad (1)$$

Keterangan:

$R_{ij}$  = Nilai rating kinerja ternormalisasi

$\max_{ij}$  = Nilai maksimum dari setiap kriteria

$\min_{ij}$  = Nilai minimum dari setiap kriteria

$x_{ij}$  = Nilai atribut yang dimiliki oleh setiap kriteria

Benefit = Jika nilai terbesar adalah yang terbaik

Cost = Jika nilai terkecil adalah yang terbaik

Dengan  $r_{ij}$  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:

$$V_i = \sum_{j=1}^n W_j R_{ij} \quad (2)$$

Keterangan:

$V_i$  = Nilai akhir dari alternatif

$W_j$  = Bobot yang telah ditentukan

$R_{ij}$  = Normalisasi matriks

Nilai  $V_i$  yang lebih besar mengidentifikasi bahwa alternatif  $A_i$  telah terpilih.

Langkah penyelesaian *Simple Addictive Weighting* adalah sebagai berikut:





- a. Menentukan kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu  $C_i$
- b. Menentukan rating kecocokan setiap alternatif pada setiap kriteria
- c. Membuat matriks keputusan berdasarkan kriteria ( $C_i$ ), kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan ataupun atribut biaya) sehingga diperoleh matriks ternormalisasi  $R$ .
- d. Hasil akhir diperoleh dari proses perangkangan yaitu penjumlahan dari perkalian matriks ternormalisasi  $R$  dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik ( $A_i$ ) sebagai solusi.

## **2.7 Entity Relationship Diagram**

*Entity Relationship Diagram* (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh *System Analysts* dalam tahap analisis persyaratan proyek pengembangan sistem. Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain database relasional yang mendasari sistem informasi yang dikembangkan. ERD bersama-sama dengan detail

pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk database. (Brady dan Loonam: 2011)

Tabel 2.1 Simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Keterangan
1		Entitas
2		Atribut
3		Hubungan
4		Garis

### 1. Entitas

Objek dalam dunia nyata yang dapat dibedakan dengan objek lain. Entitas terdiri atas beberapa atribut mengidentifikasi atau membedakan yang satu dengan yang lainnya. Pada setiap entitas baru harus memiliki 1 atribut unik atau yang disebut dengan *primary key*.

### 2. Atribut

Isi dari atribut mempunyai elemen yang dapat mengidentifikasi isi elemen satu dengan yang lain. Ada dua jenis atribut, yaitu:

- a. *Identifier (key)* digunakan untuk menentukan suatu *entity* secara unik (*primary key*).
- b. *Descriptor (nonkey attribute)* digunakan untuk menspesifikasi karakteristik dari suatu *entity* yang tidak unik.



### 3. Kardinalitas

Menyatakan jumlah himpunan relasi antar entitas.

Pemetaan kardinalitas terdiri dari:

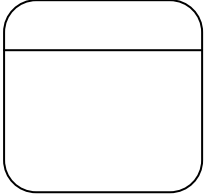
- a. *One-to-one*, sebuah entitas pada A berhubungan dengan entitas B paling banyak.
- b. *One-to-many*, sebuah entitas pada A berhubungan dengan entitas B lebih dari satu.
- c. *Many-to-many*, sebuah entitas pada A berhubungan dengan entitas B lebih dari satu dan entitas B berhubungan dengan entitas A lebih dari satu juga.




### 2.8 Data Flow Diagram

*Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Data Alir Diagram (DAD) adalah representasi grafis yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (input) dan keluaran (output). (S, Rosa A & M, Shalahuddin, 2011)

Adapun simbol-simbol yang digunakan untuk menggambarkan *Data Flow Diagram* (DFD) adalah sebagai berikut:

Tabel 2.2 Simbol Data Flow Diagram (DFD)

Simbol	Keterangan
	<p><b>Proses</b>, kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang dapat berupa masukan untuk sistem.</p>

	<p><b>External Entity</b>, menyatakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem yang lainnya, yang berada di lingkungan luarnya yang akan memberikan input atau menerima output dari sistem.</p>
	<p><b>Simpanan Data</b>, simpanan dari data yang dapat berupa suatu file atau data base di sistem komputer, suatu arsip atau catatan maual dan suatu agenda atau buku.</p>
	<p><b>Arus Data</b>, menyatakan arus data yang mengalir diantara data proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.</p>

Adapun beberapa tahapan dalam pembuatan *Data Flow Diagram* adalah:

1. Diagram level konteks, digunakan untuk menggambarkan sistem secara global.
2. Diagram level nol, digunakan untuk menggambarkan tahapan-tahapan proses data dalam diagram konteks.
3. Diagram detail (level), digunakan untuk menggambarkan arus data yang lebih mendetail dalam proses diagram level nol.