

BAB II

LANDASAN TEORI

2.1 Sistem

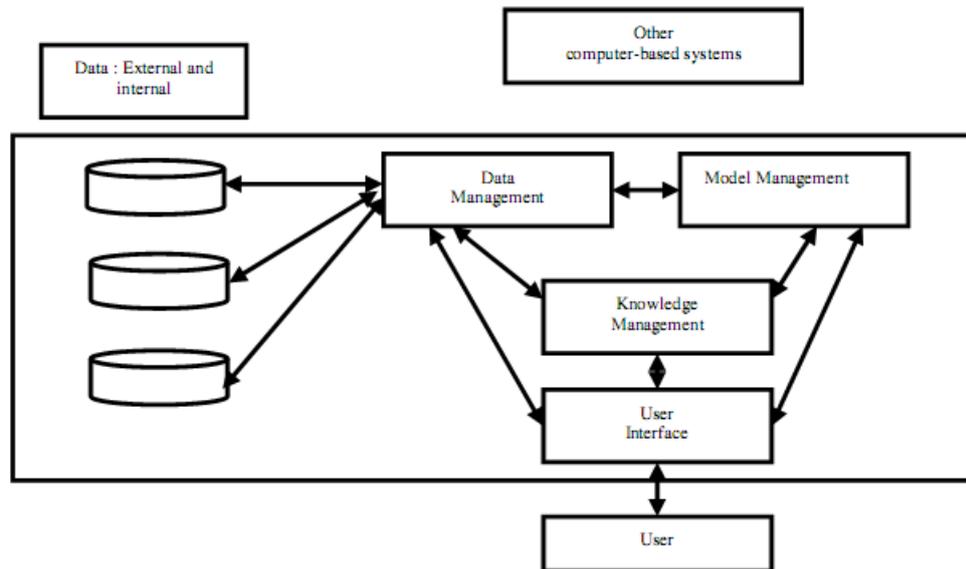
Sistem berasal dari bahasa Latin (*Systema*) dan bahasa Yunani (*Sustema*) adalah suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi, dan energi. Serta dapat diartikan sekumpulan beberapa komponen yang membentuk satu kesatuan untuk mencapai sebuah tujuan.

“Sistem adalah kumpulan elemen-elemen sistem yang saling berhubungan antara satu elemen dengan yang lainnya untuk membentuk suatu informasi”. (Sutabri, 2012)

2.2 Sistem Pendukung Keputusan (*Decision Support System*)

Sistem pendukung keputusan sebagai sebuah sistem berbasis komputer yang membantu dalam proses pengambilan keputusan. Sistem pendukung keputusan sebagai sistem informasi berbasis komputer yang adaptif, interaktif, fleksibel, yang secara khusus dikembangkan untuk mendukung solusi dari permasalahan manajemen yang tidak terstruktur untuk meningkatkan kualitas pengambilan keputusan. Dengan demikian dapat ditarik satu definisi tentang SPK yaitu sebuah sistem berbasis komputer yang adaptif, interaktif, fleksibel, yang digunakan untuk memecahkan masalah-masalah tidak terstruktur sehingga meningkatkan nilai keputusan yang diambil. (Bahtiar, 2012)

Komponen Sistem Pendukung Keputusan dapat dibangun dari subsistem berikut ini, dapat dilihat pada gambar dibawah ini :



Gambar 2.1 Komponen SPK

Keterangan :

1. Subsistem Manajemen Data (*Data Management Subsystem*), meliputi basis data – basis data yang berisi data yang relevan dengan keadaan dan dikelola *software* yang disebut DBMS (*Database Management System*).
2. Subsistem Manajemen Model (*Model Management Subsystem*), berupa sebuah paket *software* yang berisi model-model finansial, statistik, *management science*, atau model kuantitatif, yang menyediakan kemampuan analisa dan *software management* yang sesuai.
3. Subsistem Manajemen Pengetahuan (*Knowledge Management Subsystem*), merupakan subsistem (optional) yang dapat mendukung

subsistem lain atau berlaku sebagai komponen yang berdiri sendiri (*independent*).

4. Subsistem Antarmuka Pengguna (*User Interface Subsystem*), merupakan subsistem yang dapat dipakai oleh *user* untuk berkomunikasi dan memberi perintah (menyediakan *user interface*).
5. Pengguna (*user*), termasuk di dalamnya adalah pengguna (*user*), manager, dan pengambil keputusan.

2.3 *Simple Additive Weighting* (SAW)

Metode *Simple Additive Weighting* (SAW) sering juga dikenal dengan istilah penjumlahan terbobot, metode yang paling simpel dan masih banyak digunakan pada metode MADM. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Wibowo, 2010).

Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Metode SAW 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.

Pada metode *Simple Additive Weighting*, ada kriteria yang dipergunakan sebagai kriteria "*Benefit*" dan "*Cost*". Dimana kriteria "*Benefit*" digunakan jika kriteria lebih baik ketika bernilai besar, sedangkan "*Cost*" jika kriteria lebih baik ketika nilainya lebih kecil. Besar dan kecilnya

nilai tersebut dilihat dari permasalahan yang diangkat. Berikut ini adalah langkah kerja dari metode SAW :

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu C_i .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria (C_i).
4. 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 perankingan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik (A_i) sebagai solusi.

Kelebihan dari model *Simple Additive Weighting* (SAW) dibandingkan dengan model pengambilan keputusan yang lain terletak pada kemampuannya untuk melakukan penilaian secara lebih tepat karena didasarkan pada nilai kriteria dan bobot preferensi yang sudah ditentukan, selain itu SAW juga dapat menyeleksi alternatif terbaik dari sejumlah alternatif yang ada karena adanya proses perankingan setelah menentukan nilai bobot untuk setiap atribut.

Contoh kasus yang menggunakan metode Simple Additive Weighting adalah sebagai berikut :

1. Sistem pendukung keputusan pemilihan karyawan terbaik dengan metode SAW (Simple Additive Weighting) (Studi Kasus di Pamela Swalayan Ariyanto (Skripsi) Setelah melalui proses perhitungan kriteria dan perangkingan, pemilihan karyawan terbaik adalah Paryanto.
2. Sistem Pendukung Keputusan Penerimaan Beasiswa Dengan Metode Simple Additive Weighting (SAW) Pada SD Negeri Wirun 05 Mojolaban Sukoharjo Danang Putut Hartono (Skripsi) Setelah proses perhitungan kriteria dan kemudian diranking mendapatkan siswa yang berhak mendapatkan beasiswa adalah Adi.

2.4 Logika Fuzzy

Fuzzy secara bahasa dapat diartikan samar, dengan kata lain logika *fuzzy* adalah logika yang samar. Dimana pada logika *fuzzy* suatu nilai dapat bernilai “*True*” dan “*False*” secara bersamaan. Tingkat “*True*” atau “*False*” nilai dalam logika *fuzzy* tergantung pada bobot keanggotaan yang dimilikinya.

Logika *fuzzy* memiliki derajat keanggotaan rentang antara 0 hingga 1, berbeda dengan logika digital yang hanya memiliki dua keanggotaan 0 atau 1 saja pada satu waktu. Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang input dalam suatu ruang output dan memiliki nilai yang berlanjut.

Konsep tentang logika *fuzzy* diperkenalkan oleh Prof. Lotfi Astor Zadeh pada 1962. Logika *fuzzy* adalah metode sistem kontrol pemecahan masalah, yang cocok diimplementasikan pada sistem, mulai dari sistem sederhana hingga yang kompleks. Kelebihan logika *fuzzy* adalah kemampuannya dalam

proses penalaran secara bahasa sehingga dalam perancangannya tidak membutuhkan persamaan matematik yang rumit. (Kusumadewi, 2010)

2.5 *Fuzzy Multy Atribut Decision Making (FMADM)*

Fuzzy Multiple Attribute Decision Making FMADM adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Inti dari *Fuzzy* MADM adalah menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan. Pada dasarnya, ada 3 pendekatan untuk mencari nilai bobot atribut, yaitu pendekatan subyektif, pendekatan obyektif dan pendekatan integrasi antara subyektif & obyektif. Masing-masing pendekatan memiliki kelebihan dan kelemahan. Pada pendekatan subyektif, nilai bobot ditentukan berdasarkan subyektifitas dari para pengambil keputusan, sehingga beberapa faktor dalam proses perankingan alternatif bisa ditentukan secara bebas. Sedangkan pada pendekatan obyektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektifitas dari pengambil keputusan (Wibowo, 2010).

Ada beberapa metode yang dapat digunakan untuk mnyelesaikan masalah *Fuzzy* MADM. antara lain :

- a. *Simple Additive Weighting* (SAW)
- b. *Weighted Product* (WP)
- c. *ELECTRE*
- d. *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS)
- e. *Analytic Hierarchy Process* (AHP)

2.6 Hypertext Preprocessor (PHP)

Menurut dokumen resmi PHP singkatan dari *Hypertext Preprocessor*. Merupakan bahasa berbentuk script yang ditempatkan dalam *server* dan diproses di *server*. Hasilnya yang dikirimkan ke *client* melalui browser.

Script PHP diawali dengan tag `<?>` Dan diakhiri tag `?>`. Setiap baris perintah atau statement diakhiri dengan tanda titik koma. Secara umum cara penulisan script PHP dibedakan menjadi *embedded script* dan *non embedded script*. *Embedded script* adalah script PHP yang berada diantara tag-tag HTML, sedangkan *non embedded script* jika script PHP tersebut berdiri sendiri (Prasetio, 2015).

2.7 MySQL

MySQL merupakan sebuah perangkat lunak sistem manajemen basis data SQL / DBMS (*Database Management System*) yang multithread, multi-user dan sekitar 6 juta instalasi diseluruh indonesia. Didistribusikan secara gratis dibawah lisensi GPL (*General Public License*).Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat komersial (Prasetio, 2015).

Keistimewaan MySQL adalah sebagai berikut :

- a. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
- b. *Open Source*. MySQL didistribusikan secara open source, dibawah lisensi GPL sehingga dapat digunakan secara gratis.

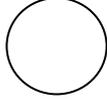
- c. *Multiuser*. MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
- d. *Performance tuning*. MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
- e. Jenis Kolom. MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.
- f. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).
- g. Keamanan. MySQL memiliki beberapa lapisan sekuritas seperti *level subnetmask*, nama *host*, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.

2.8 Context Diagram

Context Diagram merupakan pola penggambaran yang berfungsi untuk memperlihatkan interaksi sistem informasi dengan lingkungan dimana sistem itu berada (Nugroho, 2011).

Context Diagram adalah diagram tingkat atas yaitu diagram yang paling tidak detail dari sistem informasi yang menggambarkan alur kedalam dan keluar.

Tabel 2.1 Simbol *Context Diagram*

SIMBOL	KETERANGAN
	Menggambarkan entitas atau proses dimana aliran data ditransformasikan ke aliran data keluar
	Simbol entitas eksternal, menggambarkan asal atau tujuan data
	Sistem aliran data atau aliran data

2.9 Data Flow Diagram (DFD)

Data flow diagram adalah suatu model logika data atau proses yang dibuat untuk menggambarkan awal data, tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut, interaksi data yang tersimpan, dan proses yang dikenakan pada data tersebut (Nugroho, 2011). Dalam basis data ada beberapa simbol yang digunakan pada DFD untuk mewakili:

1. Kesatuan Luar

Kesatuan luar (*Eksternal Entity*) merupakan kesatuan (*Entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan *input* atau menerima *output* dari sistem.

2. Arus Data

Arus Data (*Data Flow*) di DFD diberi simbol satu panah. Arus data ini mengalir diantara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

3. Proses

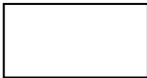
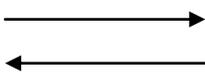
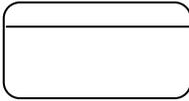
Proses menunjukkan bagian yang mengubah *input* menjadi *output*, yaitu menunjukkan bagaimana satu atau lebih *input* diubah menjadi beberapa *output*. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

4. Data Store

Merupakan tempat menyimpan data yang dapat berupa suatu file atau *database* pada sistem komputer.

Beberapa simbol yang digunakan dalam DFD dapat dilihat pada tabel 2.2 sebagai berikut.

Tabel 2.2 *Data Flow Diagram*

SIMBOL	ARTI	KETERANGAN
	Entitas	Entitas merupakan sumber tujuan dari aliran data ke sistem.
	Aliran Data	Aliran data menggambarkan aliran data dari satu proses ke proses lainnya.
	Proses	Proses atau fungsi yang mentransformasikan data secara umum.
	Penyimpanan Data	Berkas atau tempat penyimpanan fungsi untuk menyimpan data atau file.

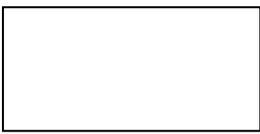
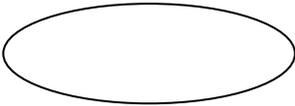
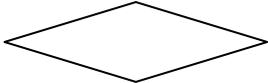
2.10 Entity Relationship Diagram (ERD)

Entity relationship diagram (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. Tujuan utama dari penggambaran ERD adalah untuk menunjukkan struktur objek data (*Entity*) dan hubungan (*Relationship*) yang ada pada objek

tersebut. ERD berguna bagi profesional sistem karena ERD memperlihatkan hubungan antara *data store* pada DFD (Nugroho, 2011).

Entity relationship diagram menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data. Pada dasarnya ada 4 simbol yang digunakan. Simbol-simbol ERD ditunjukkan pada tabel 2.3.

Tabel 2.3 *Entity Relationship Diagram*

SIMBOL	ARTI	KETERANGAN
	Entitas	Entitas digunakan untuk menggambarkan objek yang dapat di identifikasikan dalam lingkungan pemakai.
	Atribut	Atribut digunakan untuk menggambarkan elemen-elemen dari satu entitas, yang menggambarkan karakter entitas.
	Hubungan	Entitas dapat berhubungan satu sama lain. Hubungan ini disebut dengan relasi.
	Garis	Garis digunakan untuk menghubungkan entitas dengan atribut.