

BAB II

LANDASAN TEORI

2.1. Sistem

Sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan.

2.2. Sistem Pendukung Keputusan

Sistem pendukung keputusan (SPK) adalah bagian dari sistem informasi berbasis komputer termasuk sistem berbasis pengetahuan atau manajemen pengetahuan yang digunakan untuk mendukung pengambilan keputusan dalam suatu organisasi atau perusahaan. Dapat juga dikatakan sebagai sistem pendukung berbasis komputer bagi para pengambil keputusan manajemen yang menangani masalah-masalah yang tidak terstruktur (Karismariyanti, 2011).

Pengertian diatas dapat dijelaskan bahwa SPK bukan merupakan alat pengambilan keputusan, melainkan merupakan sistem yang membantu pengambil keputusan dengan melengkapi mereka dengan informasi dari data yang telah diolah dengan relevan dan diperlukan untuk membuat keputusan tentang suatu masalah dengan lebih cepat dan akurat. Sistem ini tidak dimaksudkan untuk menggantikan pengambilan keputusan dalam proses pembuatan keputusan.

2.3. *Fuzzy Multiple Attribute Decision Making (FMADM)*

Proses MADM dilakukan melalui dua tahap, yaitu penyusunan komponen-komponen situasi, analisis dan sintesis informasi. Tahap penyusunan komponen, komponen situasi akan dibentuk tabel taksiran yang berisi identifikasi alternatif dan spesifikasi tujuan, kriteria dan atribut. Salah satu cara untuk menspesifikasikan tujuan situasi $\{O_i, i=1, \dots, n\}$, selain itu juga disusun atribut-atribut yang akan digunakan $\{A_k, k=1, \dots, m\}$.

Tahap analisis dilakukan melalui dua langkah. Pertama mendatangkan taksiran dari besaran yang potensial, kemungkinan, dan ketidakpastian yang berhubungan dengan dampak-dampak yang mungkin pada setiap alternatif. Kedua meliputi pemilihan dari preferensi pengambil keputusan untuk setiap nilai dan ketidakpedulian terhadap resiko yang timbul. Pada langkah pertama, beberapa metode menggunakan fungsi distribusi $\{P_j(X)\}$ yang menyatakan probabilitas kumpulan-kumpulan atribut $\{a_k\}$ terhadap setiap alternatif $\{A_i\}$. Konsekuen juga dapat ditentukan secara langsung dari agregasi sederhana yang dilakukan pada informasi terbaik yang tersedia. Ada beberapa cara untuk menentukan preferensi pengambil keputusan pada setiap konsekuen yang dapat dilakukan pada langkah kedua. Metode yang paling sederhana untuk menurunkan bobot atribut dan kriteria adalah dengan fungsi utilitas atau penjumlahan berbobot. Model *multi-attribute decision making* secara umum sebagai berikut : Misalkan $A=\{a_i \mid i=1, \dots, n\}$ adalah himpunan alternatif-

alternatif keputusan dan $C=\{c_j \mid j=1,\dots,m\}$ adalah himpunan tujuan yang diharapkan, maka akan ditentukan alternatif x^0 yang memiliki derajat harapan tertinggi terhadap tujuan-tujuan yang relevan c_j .

Sebagian besar pendekatan MADM dilakukan melalui 2 langkah, yaitu: pertama, melakukan agregasi terhadap keputusan-keputusan yang dianggap terhadap semua tujuan pada setiap alternatif; kedua, melakukan perankingan alternatif-alternatif keputusan tersebut berdasarkan hasil agregasi keputusan. Dengan demikian, bisa dikatakan bahwa masalah MADM adalah mengevaluasi m alternatif A_i ($i=1,2,\dots,n$) terhadap sekumpulan atribut atau kriteria C_j ($j=1,2,\dots,n$), dimana setiap atribut saling tidak bergantung satu dengan yang lainnya. Matriks keputusan setiap alternatif terhadap setiap atribut X diberikan sebagai:

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{1n} \\ x_{21} & x_{22} & x_{2n} \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & x_{mn} \end{bmatrix}$$

dimana x_{ij} merupakan rating kinerja alternatif ke- i terhadap atribut ke- j . Nilai bobot yang menunjukkan tingkat kepentingan relatif setiap atribut, diberikan sebagai W : $W=\{w_1, w_2, \dots, w_n\}$

Rating kinerja (X) dan nilai bobot (W) merupakan nilai utama yang merepresentasikan preferensi absolut dari pengambil keputusan. Masalah

MADM diakhiri dengan proses perankingan untuk mendapatkan alternatif terbaik yang diperoleh berdasarkan nilai keseluruhan preferensi yang diberikan. Ada beberapa metode yang dapat digunakan untuk menyelesaikan masalah FMADM, antara lain:

a. *Simple Additive Weighting (SAW)*

Metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Fishburn, 1967) (MacCrimmon, 1968). Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada.

b. *Weighted Product (WP)*

Metode Weighted Product (WP) menggunakan perkalian untuk menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan dulu dengan bobot atribut yang bersangkutan. Proses ini sama halnya dengan proses normalisasi.

c. *Elimination and Choise Exspressing Reality (ELECTRE)*

Electre merupakan salah satu metode pengambilan keputusan multikriteria berdasarkan pada konsep *outracking* dengan menggunakan perbandingan berpasangan dari alternatif-alternatif berdasarkan setiap kriteria yang sesuai. Metode *electre* digunakan pada kondisi dimana alternatif yang kurang sesuai dengan kriteria dieliminasi, dan alternatif

yang sesuai dapat dihasilkan. *Electre* digunakan untuk kasus-kasus dengan banyak alternatif namun hanya sedikit kriteria yang dilibatkan.

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

TOPSIS adalah salah satu metode pengambilan keputusan multukriteria yang pertama kali dikenalkan oleh Yoon dan Hwang (1981). TOPSIS menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan jarak terpanjang (terjauh) dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak *Eudiclien* (Jarak antara dua titik) untuk menentukan kedekatan relatif dari suatu alternatif dengan solusi optimal.

Solusi ideal positif didefinisikan sebagai jumlah dari seluruh nilai terbaik yang dapat dicapai untuk setiap atribut, sedangkan solusi negatif – ideal terdiri dari seluruh nilai terburuk yang dicapai untuk setiap atribut. TOPSIS mempertimbangkan keduanya, jarak terhadap solusi ideal positif dan jarak terhadap solusi ideal negatif dengan mengambil kedekatan relatif terhadap solusi ideal positif dan jarak terhadap solusi ideal negatif dengan mengambil kedekatan relatif terhadap solusi ideal positif.

e. *Analytic Hierarchy Proses (AHP)*

AHP adalah sebuah metode memecah permasalahan yang kompleks / rumit dalam situasi yang tidak terstruktur menjadi bagian-bagian komponen. Mengatur bagian atau variabel ini menjadi suatu bentuk hierarki, kemudian memberikan nilai numerik untuk penilaian subjektif terhadap kepentingan relatif dari setiap variabel dan mensintesis penilaian untuk variabel mana yang memiliki prioritas tertinggi yang akan mempengaruhi penyelesaian dari situasi tersebut.

2.4. Algoritma FMADM

Algoritma penyelesaian FMADM adalah:

- a. Memberikan nilai setiap alternatif (A_i) pada setiap kriteria (C_j) yang sudah ditentukan, dimana nilai tersebut di peroleh berdasarkan nilai crisp; $i=1,2,\dots,m$ dan $j=1,2,\dots,n$.
- b. Memberikan nilai bobot (W) yang juga didapatkan berdasarkan nilai crisp.
- c. 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 ($\text{MAX } X_{ij}$) dari tiap kolom, sedangkan

untuk atribut biaya, nilai crisp MIN ($\text{MIN } X_{ij}$) dari tiap kolom atribut dibagi dengan nilai crisp (X_{ij}) setiap kolom. Relatif dari alternatif-alternatif keputusan dalam bentuk matematis yang sederhana. Dan untuk merancang agar sistem ini lebih mudah di akses maka akan dirancang sebuah sistem pendukung keputusan yang berbasis web.

- d. Melakukan proses perankingan dengan cara mengalikan matriks ternormalisasi (R) dengan nilai bobot (W).
- e. Menentukan nilai preferensi untuk setiap alternatif (V_i) dengan cara menjumlahkan hasil kali antara matriks ternormalisasi (R) dengan nilai bobot (W). Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih.

2.5. Metode Topsis

TOPSIS didasarkan pada konsep dimana alternatif terpilih yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif, namun juga memiliki jarak terpanjang dari solusi ideal negatif. Konsep ini banyak digunakan pada beberapa model MADM untuk menyelesaikan masalah keputusan secara praktis. Hal ini disebabkan konsepnya sederhana dan mudah dipahami, komputasinya efisien, dan memiliki kemampuan untuk mengukur kinerja relatif dari alternatif-alternatif keputusan dalam bentuk matematis yang sederhana.

Secara umum, prosedur TOPSIS mengikuti langkah-langkah sebagai berikut:

- a. Membuat matriks keputusan yang ternormalisasi;
- b. Membuat matriks keputusan yang ternormalisasi terbobot;
- c. Menentukan matriks solusi ideal positif dan matriks solusi ideal negatif;
- d. Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif dan matriks solusi ideal negatif;
- e. Menentukan nilai preferensi untuk setiap alternatif.

TOPSIS membutuhkan rating kerja setiap alternatif A_i pada setiap kriteria C_j yang ternormalisasi. (Persamaan 3.1)

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}} \dots\dots\dots(3.1)$$

dengan $i=1,2,\dots,m$; dan $j=1,2,\dots,n$

dimana :

r_{ij} = matriks ternormalisasi $[i][j]$

x_{ij} = matriks keputusan $[i][j]$

Solusi ideal positif A^+ dan solusi ideal negatif A^- dapat ditentukan berdasarkan rating bobot ternormalisasi (y_{ij}) sebagai :

$y_{ij} = w_i \cdot r_{ij}$; dengan $i=1,2,\dots,m$; dan $j=1,2,\dots,n$

$A^+ = (y_1^+, y_2^+, \dots, y_n^+)$;

$A^- = (y_1^-, y_2^-, \dots, y_n^-)$;

dimana :

y_{ij} = matriks ternormalisasi terbobot $[i][j]$

w_i = vektor bobot $[i]$ dari proses AHP

y_j^+ = max y_{ij} , jika j adalah atribut keuntungan min y_{ij} , jika j adalah atribut biaya

y_j^- = min y_{ij} , jika j adalah atribut keuntungan max y_{ij} , jika j adalah atribut biaya

$j = 1, 2, \dots, n$

Jarak antara alternatif A_i dengan solusi ideal positif :

(lihat persamaan 3.2)

$$D_i^+ = \sqrt{\sum_{i=1}^n (y_i^+ - y_{ij})^2} \dots \dots \dots (3.2)$$

dimana :

D_i^+ = jarak alternatif A_i dengan solusi ideal positif

y_i^+ = solusi ideal positif $[i]$

y_{ij} = matriks normalisasi terbobot $[i][j]$

Jarak antara alternatif A_i dengan solusi ideal negatif

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_i^-)^2} \dots \dots \dots (3.3)$$

$i=1,2,\dots,m$ persamaan

dimana :

D_i^- = jarak alternatif A_i dengan solusi ideal negatif

y_i^- = solusi ideal positif $[i]$

y_{ij} = matriks normalisasi terbobot $[i][j]$

Nilai preferensi untuk setiap alternatif (V_i) dapat

dilihat pada persamaan 3.4

$$V_i = \frac{D_i^-}{D_i^- + D_i^+}; i = 1, 2, \dots, m \quad \dots\dots\dots(3.4)$$

dimana :

V_i = kedekatan tiap alternatif terhadap solusi ideal

D_i^+ = jarak alternatif A_i dengan solusi ideal positif

D_i^- = jarak alternatif A_i dengan solusi ideal negatif



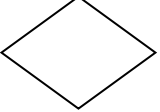

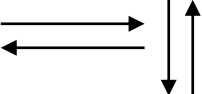

Nilai V_i yang lebih besar menunjukkan bahwa alternatif A_i lebih dipilih.

2.6. Desain Sistem

2.6.1. Flowchart

Flowchart (Bagian Alir Data) adalah bagan yang menunjukkan alir didalam program atau prosedur sistem secara logika. Bagan alir ini digunakan terutama untuk mendefinisikan hubungan antara bagian (pelaku proses), proses manusia maupun proses komputer dan aliran data (dalam bentuk masukan dan keluaran).

Tabel 2. 1 Simbol Flowchart

| No | Simbol | Keterangan |
|----|---|---|
| 1 |  | Dokumen, digunakan untuk menunjukan dokumen input dan output baik untuk proses manual, mekanik, atau komputer. |
| 2 |  | Penghubung ,digunakan untuk menunjukkan hubungan dengan bagian lain dalam satu halaman. |
| 2 |  | Simbol <i>decision</i> , yaitu menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya / tidak |
| 4 |  | Proses, digunakan untuk menunjukkan kegiatan proses dari operasi program komputer. |
| 5 |  | Garis Alir, digunakan untuk menunjukkan arus proses |
| 6 |  | Terminator yang berfungsi untuk eksekusi suatu data . |

2.6.2. *Data Flow Diagram*

Diagram yang menggunakan notasi simbol untuk menggambarkan arus data sistem. Ada beberapa simbol digunakan pada DFD untuk mewakili :

1. Kesatuan Luar

Kesatuan luar (*external 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 suatu panah. Arus data ini mengalir di antara 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

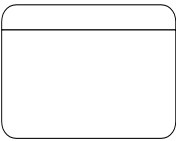
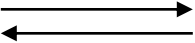
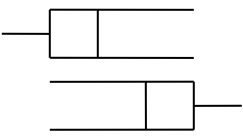
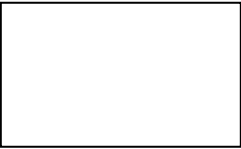
Proses (*process*) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

4. Data Store

Simpanan Data (*Data Store*) merupakan simpanan dari data yang dapat berupa suatu file atau data base pada sistem komputer.

Dalam menggambarkan diagram arus data atau *data flow diagram* menggunakan simbol-simbol seperti dibawah ini :

Tabel 2. 2 Simbol data flow diagram

| No | Simbol | Keterangan |
|----|---|---|
| 1 |  | <ul style="list-style-type: none"> • Simbol proses • Menunjukkan proses komputerisasi. |
| 2 |  | <ul style="list-style-type: none"> • Simbol Aliran Data • Menunjukkan arah ke bagian lain atau ke proses sebaliknya. |
| 2 |  | <ul style="list-style-type: none"> • Simbol penyimpanan • Menunjukkan sebagai komponen untuk memudahkan kumpulan data atau informasi |
| 4 |  | <ul style="list-style-type: none"> • Simbol terminator • Menunjukkan organisasi (kelompok organisasi) atau organisasi diluar sistem lain yang memberi atau menerima data. |

2.7. Basis Data

Basis Data merupakan kumpulan data yang berlatas yang disusun, diorganisasikan dan disimpan secara sistematis dalam media simpan komputer mengacu pada metode-metode tertentu sedemikian rupa sehingga dapat diakses secara cepat dan mudah menggunakan program/aplikasi komputer untuk memperoleh data dari basis data tersebut. (Fathansyah,2012).

2.7.1. Entitas Relationship Diagram

Entity Relationship Diagram (ERD) merupakan suatu kumpulan file-file yang saling berkaitan satu dengan yang lainnya pada model data relation hubungan antar file direlasikan dengan kunci relasi (*Relation Key*) yang merupakan kunci utama dimasing-masing file. Perancangan database yang tepat akan menyebabkan MySql/paket program lainnya akan bekerja dengan optimal.

Entitas Relationship Diagram atau disebut ERD, adalah Mendokumentasikan data perusahaan dengan mengidentifikasi jenis dan hubungannya. Komponen-komponen ERD yaitu:

1. Entitas

Jenis entitas (*Entity Type*) dapat berupa suatu elemen lingkungan, sumber daya, atau transaksi yang begitu pentingnya bagi perusahaan sehingga didokumentasikan dengan data jenis entitas didokumentasikan dengan simbol persegi panjang.

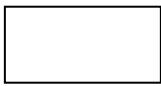
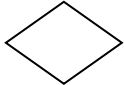
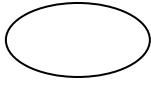
2. Hubungan

Hubungan adalah suatu asosiasi yang ada antara dua jenis entitas. Hubungan digambarkan dengan bentuk belah ketupat. Tiap belah ketupat diberi label kata kerja.

3. Atribut

Atribut adalah karakteristik dari suatu entitas. Atribut-atribut tersebut sebenarnya adalah elemen-elemen data dan masing-masing diberikan satu nilai tunggal, yang disebut nilai atribut digambarkan dalam bentuk elips.

Tabel 2. 3 Tabel Entitas Relationship Diagram

| No | Simbol | Keterangan |
|----|---|------------|
| 1 |  | Entitas |
| 2 |  | Hubungan |
| 2 |  | Atribut |

2.8. MySql

MySQL (dibaca: mi-se-kyu-el) merupakan software yang tergolong sebagai DBMS (*database Management System*) yang bersifat open source. Open source menyatakan bahwa software ini dilengkapi dengan

source code (code yang dipakai untuk membuat MySQL). Selain tentu saja bentuk executable-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi. Dan bisa diperoleh secara gratis dengan mendownload di internet. MySQL awalnya dibuat oleh perusahaan konsultan bernama TeX yang berlokasi di Swedia. Saat ini pengembangan MySQL berada dibawah naungan MySQL AB. Sebagai software DBMS, MySQL memiliki sejumlah fitur seperti yang akan dijelaskan dibawah ini.

a. Mutli platform

MySQL tersedia pada beberapa platform (windows, linux, unix, dan lainlain)

b. Kuat, cepat dan mudah digunakan.

MySQL tergolong sebagai database server (server yang melayani permintaan terhadap database) yang andal, dapat menangani database database yang besar dengan kecepatan tinggi. Mendukung banyak sekali fungsi untuk mengakses database dan sekaligus mudah untuk digunakan.

c. Jaminan keamanan akses

MySQL mendukung pengamanan database dengan berbagai kriteria pengaksesan. Sebagai gambaran, dimungkinkan untuk mengatur user tertentu agar bisa mengakses data yang bersifat

rahasia (misal gaji pegawai), sedangkan user lain tidak boleh sesuai dengan hak aksesnya.

d. Dukungan SQL

Seperti tersirat namanya, SQL mendukung perintah SQL (*Structured Query Language*). Sebagaimana diketahui SQL merupakan bahasa standar dalam pengaksesan database rasional. Pengetahuan akan SQL akan memudahkan siapapun untuk menggunakan MySQL.

2.9. PHP (*Hypertext Preprocessor*)

2.9.1. Definisi PHP

PHP merupakan salah satu bahasa pemrograman yang berjalan dalam sebuah web server dan berfungsi sebagai pengolahan data pada sebuah server. Sebuah website akan lebih interaktif dan dinamis dengan menggunakan PHP. Data yang dikirim oleh pengunjung website / komputer *client* akan dikelola dan disimpan dalam database web server dan dapat ditampilkan kembali apabila diakses. Pada awalnya PHP merupakan singkatan dari *Personal Home Page*. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga

website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll.

2.9.2. Kelebihan PHP

Dalam perkembangannya, aplikasi PHP memiliki berbagai kemudahan dan keunggulan dibandingkan dengan beberapa produk sejenis yang ada. Hal inilah yang menjadi alasan menggunakan PHP dalam pembuatan pemrograman web, berikut adalah beberapa kelebihan yang dimiliki oleh PHP :

1. *PHP Open Source*

PHP adalah aplikasi bahasa web yang bisa diperoleh secara gratis dan kode program PHP diberikan secara cuma - cuma.

2. PHP mudah untuk dipelajari, dibandingkan dengan produk lain yang mempunyai fungsi yang sama.

3. *PHP embedded*

Penulisan script PHP menyatu dengan HTML, sehingga memudahkan pembuatannya dan membuat web menjadi dinamis.

4. PHP berjalan di banyak *platform*

PHP yang menyatu dalam dokumen HTML bisa diakses oleh seluruh browser di seluruh *platform* yang ada. Artinya dapat dijalankan oleh semua sistem operasi karena PHP berjalan secara *Web Base* yang artinya semua sistem operasi bahkan handphone

yang mempunyai *web browser* dapat menggunakan program PHP tersebut.

5. PHP bukan berbasis Tag

PHP adalah murni bahasa pemrograman, PHP dapat mendefinisikan fungsi-fungsi dengan menuliskan nama dan definisinya.