

BAB II

LANDASAN TEORI

2.1 Sistem Pendukung Keputusan

Konsep Sistem Pendukung Keputusan (SPK) pertama kali diungkapkan pada awal tahun 1970-an oleh Michael S. Scott Morton dengan istilah Management Decision System. Sistem tersebut adalah suatu sistem yang berbasis komputer yang ditunjukan untuk membantu pengambil keputusan dengan memanfaatkan data dan model tertentu untuk memecahkan berbagai persoalan yang tidak terstruktur (Dewanto, 2015).

Sistem Pendukung Keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan dan manipulasi data. Sistem ini digunakan untuk pengambilan keputusan dalam situasi yang semiterstruktur dan tidak terstruktur. Pada dasarnya SPK dirancang untuk mendukung seluruh tahap pengambilan keputusan mulai dari mengidentifikasi masalah, memilih data yang relevan, menentukan pendekatan yang digunakan dalam proses pengambilan keputusan, sampai mengevaluasi pemilihan alternatif.

2.2 Metode Yang Digunakan

2.2.1 MFEP (Multifactor Evaluation Process)

Multifactor Evaluation Process (MFEP) adalah metode kuantitatif yang menggunakan weighting system. Dalam pengambilan keputusan multifaktor, pengambil keputusan secara subyektif dan intuitif menimbang berbagai faktor yang mempunyai pengaruh penting terhadap alternatif pilihan mereka. Untuk keputusan yang berpengaruh secara strategis, lebih dianjurkan menggunakan sebuah pendekatan kuantitatif seperti MFEP. Dalam MFEP pertama-tama seluruh kriteria yang menjadi faktor penting dalam melakukan pertimbangan diberikan pembobotan (weighting) yang sesuai. Langkah yang sama juga dilakukan terhadap alternatif-alternatif yang akan dipilih, yang kemudian dapat dievaluasi berkaitan dengan factor-faktor pertimbangan tersebut. Metode MFEP menentukan bahwa alternatif dengan nilai tertinggi adalah solusi. (Khaidir, 2014)

Proses pemilihan alternatif terbaik menggunakan weighting system, dimana metode tersebut merupakan metode kuantitatif, disebut sebagai metode Multifactor Evaluation Process (MFEP). Multifactor evaluation process banyak digunakan dengan alasan:

- a. Konsepnya sederhana dan mudah dipahami;
- b. Komputasinya efisien;
- c. Memiliki kemampuan untuk mengukur kinerja relatif dan alternatif-alternatif keputusan dalam bentuk matematis yang sederhana.

Pengambilan keputusan menggunakan metode Multi Factor Evaluation Process dilakukan secara subyektif dengan menimbang beberapa faktor yang berpengaruh terhadap alternatif (Y.Primadasa & V.Amalia, 2017). Langkah-langkah proses perhitungan menggunakan metode MFEP, yaitu :

1. Menentukan faktor dan bobot faktor dimana total pembobotan harus sama dengan 1 (Σ pembobotan = 1 atau disebut factor weight).
2. Mengisikan nilai tiap faktor yang di perlukan dalam proses pengambilan keputusan, nilai yang dimasukan merupakan nilai objektif yaitu factor evaluation yang nilainya 0 – 1
3. Proses perhitungan weight evaluation merupakan perhitungan antara factor weight dan factor evaluation dengan penjumlahan, dari hasil weight evaluation dapat menentukan hasil evaluasi.

Dari langkah proses perhitungan menggunakan metode MFEP di atas dapat direalisasikan sebagai berikut:

1. Perhitungan nilai bobot evaluasi factor ditunjukkan dalam persamaan (1).

$$EF = \frac{\sum x}{\sum x_{max}} \dots\dots\dots (1)$$

Keterangan:

EF : Evaluasi Faktor

X : Nilai Subkriteria

X_{max} : Nilai x max

2. Perhitungan nilai bobot evaluasi ditunjukkan dalam persamaan (2).

$$WE = FW * E \dots\dots\dots (2)$$

Keterangan:

WE : Nilai bobot evaluasi

FW : Nilai bobot factor

E : Nilai evaluasi faktor

3. Perhitungan nilai total evaluasi ditunjuk dalam persamaan (3).

$$\sum_{i=1}^n WE_i = WE_1 + WE_2 + WE_n \dots\dots\dots (3)$$

$\sum_{i=1}^n WE_i$: Totak nilai bobot evaluasi

WE_i : Nilai bobot evaluasi ke – i

2.2.2 MAUT (Metode Multi-Attribute Utility Theory)

Dalam Metode *Multi-Attribute Utility Theory* (MAUT) diperlukan pembangunan multi-attribute utility model, yaitu penspesifikasian dimensi dari permasalahan evaluasi dan keputusan secara spesifik. Langkah-langkah dalam proses MAUT adalah sebagai berikut :

1. Buat kerangka keputusan, dengan mendefinisikan permasalahan.
2. *Generate*(bangkitkan) alternatif-alternatif yang mungkin dapat memecahkan masalah.
3. Buat daftar (*list*) semua aspek yang mempengaruhi keputusan.
4. Beri bobot untuk setiap aspek yang ada. Bobot yang ada harus mencerminkan seberapa penting aspek-aspek tersebut terhadap permasalahan.
5. Beri juga bobot dari alternatif-alternatif yang ada. Untuk setiap alternatif, tentukan seberapa memuaskan alternatif tersebut terhadap setiap aspek.
6. Proses evaluasi dari setiap alternatif pada aspek-aspek yang ada untuk mendapatkan keputusan.

Dalam metode *Multi-Attribute Utility Theory* digunakan untuk merubah dari beberapa kepentingan kedalam nilai *numerik* dengan skala 0-1 dengan 0 mewakili pilihan terburuk dan 1 terbaik. Hal ini memungkinkan perbandingan langsung beragam ukuran. Yaitu, dengan alat yang tepat, itu memungkinkan saja untuk membandingkan apel dengan jeruk. Hasil akhirnya adalah urutan peringkat dari

evaluasi alternatif yang menggambarkan pilihan dari para pembuat keputusan..

Untuk perhitungannya dapat dirumuskan sebagai berikut:

$$V(x) = \sum_{i=1}^n w_j \cdot x_{ij} \dots\dots\dots (1)$$

Secara ringkas (Puspitasari, dkk, 2013), langkah-langkah dalam metode MAUT adalah sebagai berikut :

1. Pecah sebuah keputusan ke dalam dimensi yang berbeda.
2. Tentukan bobot relatif pada masing-masing dimensi.
3. Daftar semua alternatif.
4. Masukkan utility untuk masing-masing alternative sesuai atributnya.
5. Kalikan utility dengan bobot untuk menemukan nilai masing-masing alternatif.

Contoh Penyelesaian Metode MAUT:

$$A_1 = \{21, 30, 32\}$$

$$A_2 = \{31, 31, 22\}$$

Normalisasi matrik:

$$U(x) = \frac{x - x_i^-}{x_i^+ - x_i^-} \dots\dots\dots (2)$$

Keterangan:

$U(x)$: Normalisasi bobot alternatif

x : Bobot alternative

x_i^- : Bobot terburuk (minimum) dari kriteria ke-x

x_i^+ : Bobot terbaik (maximum) dari kriteria ke-x

$$U_{(1,1)} = (21-21)/(31-21) = 0$$

$$U_{(1,2)} = (30-30)/(31-30) = 0$$

$$U_{(1,3)} = (32-22)/(32-22) = 1$$

$$U_{(2,1)} = (31-21)/(31-21) = 1$$

$$U_{(2,2)} = (31-30)/(31-30) = 1$$

$$U_{(2,3)} = (22-22)/(32-22) = 0$$

Perangkirngan alternative dengan persamaan (1):

$$A1 = (3.00 \cdot 0.00) + (5.00 \cdot 0.00) + A1 (4.00 \cdot 1.00) = 4$$

$$A2 = (3.00 \cdot 1.00) + (5.00 \cdot 1.00) + A1 (4.00 \cdot 0.00) = 3+5=8$$

Dari hasil di atas dapat disimpulkan bahwa alternative yang terpilih adalah alternative ke A2.

2.3 Script pada Aplikasi Web

2.3.1 HTML

Hypertext markup language (HTML) merupakan bahasa dasar pembuatan web. HTML menggunakan tanda (mark), untuk menandai bagian-bagian dari text. HTML disebut sebagai bahasa dasar, karena dalam membuat web, jika hanya menggunakan HTML maka tampilan web terasa hambar (Rerung, 2018:18).

Hypertext markup language (HTML) merupakan bahasa pemrograman dasar untuk mengelola website. Akan tetapi, HTML hanya terbatas pada pembuatan website statis (website yang tidak dapat berinteraksi aktif dengan user). Maka dari itu, HTML biasa dikombinasikan dengan bahasa pemrograman web lainnya (Wardana, 2016:3). Berdasarkan pengertian di atas, maka dapat disimpulkan bahwa hypertext markup language (HTML) merupakan bahasa pemrograman yang digunakan pada dokumen web atau bahasa standar untuk menyebarkan informasi pada web dan menampilkan halaman web dimana saja serta bersifat statis.

2.3.2 PHP

Menurut (Budi Raharjo, 2012) dalam Buku “Modul Pemrograman Web (HTML, PHP, & MYSQL)”. PHP adalah salah satu pemrograman skrip yang dirancang untuk membangun aplikasi web. Ketika dipanggil dari web browser, program yang ditulis dengan PHP akan di-parsing didalam web server oleh *interpreter* PHP dan diterjemahkan dalam dokumen HTML, yang selanjutnya akan ditampilkan kembali ke web browser. Karena pemrosesan program PHP dilakukan dilingkungan web server, PHP dikatakan sebagai bahasa sisi server (*server-side*)

oleh sebab itu, seperti yang telah dikemukakan sebelumnya, kode PHP tidak akan terlihat pada saat user memilih perintah “view source” pada web browser yang mereka gunakan. Selain menggunakan PHP, aplikasi web juga dapat dibangun dengan Java (JSP- *JavaServer Pages* dan Servlet), Perl, maupun ASP (*Active Server Pages*).

Syntax Program PHP adalah bahasa yang dirancang untuk mudah diletakkan dalam kode HTML. Banyak dijumpai kode PHP yang menyatu dengan kode HTML. Kode PHP diawali dengan tag `<?php` dan diakhiri dengan tag `?>`. apabila kita melakukan konfigurasi terhadap file PHP ini untuk mengizinkan pengguna tag pendek (short tag) dengan mengubah nilai `short_open_tag` menjadi on, maka tag tersebut dapat digantikan dengan `<? dan ?>`. dalam PHP, nilai default dari `short_open_tag` adalah off. Selain itu, PHP kita juga dapat menggunakan tag gaya ASP `<% dan %>`, dengan mengubah nilai `asp_tags` dalam file PHP ini menjadi on. Berikut ini contoh kode PHP :

```
<? Php  
  
echo “Hello World”;  
  
?>
```

Perintah echo di dalam PHP berguna untuk mencetak nilai, baik teks maupun numerik ke layar web browser.

2.3.3 CSS

Menurut Sibero (2013:112) mendefinisikan bahwa “cascading style sheets memiliki arti gaya menata halaman bertingkat, yang berarti setiap satu elemen yang telah di format, dan memiliki anak dan telah di format, maka anak dari elemen tersebut secara otomatis mengikuti format elemen induknya”.

Cascading style sheet (CSS) merupakan bahasa pemrograman yang berfungsi untuk mempercantik tampilan web (Solichin, 2016:10). Berdasarkan pengertian di atas, maka dapat disimpulkan bahwa cascading style sheets (CSS) merupakan bahasa yang digunakan untuk membantu programmer dalam merancang sebuah tampilan website dan menimbulkan efek animasi yang bagus.

2.3.4 JAVASCRIPT

Javascript berfokus pada proses pengolahan data di sisi client dan menyajikan komponen web yang lebih interaktif serta berfungsi untuk menambah fungsionalitas dan kenyamanan halaman web (Solichin, 2016:11). Menurut Sibero (2013:150) mengatakan bahwa “Javascript adalah suatu bahasa pemrograman yang di kembangkan untuk dapat berjalan pada web browser”.

Berdasarkan kutipan di atas, dapat disimpulkan bahwa javascript merupakan bahasa pemrograman yang berbasis client dan script untuk tampilan pendukung pada website sehingga membuat halaman menjadi lebih interaktif.

2.4 Database

2.4.1 XAMPP

Xampp adalah sebuah paket kumpulan software yang terdiri dari apache, mysql, phpmyadmin, php, perl, Freetype2, dll. Xampp berfungsi untuk memudahkan instalasi lingkungan PHP, dimana biasanya lingkungan pengembangan web memerlukan php, apache, mysql, dan phpmyadmin serta software-software yang terkait dengan pengembangan web. Dengan menggunakan xampp, kita tidak perlu menginstall aplikasi-aplikasi tersebut.

2.4.2 MySQL

Berikut ini ada beberapa definisi tentang MySQL yang disajikan oleh beberapa ahli :

Sibero (2014:97), “MySQL atau dibaca “My Sekuel” dengan adalah suatu RDBMS (Relational Database Management System) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data”.

Hidayatullah dan Kawistara (2014:231), “MySQL adalah salah satu aplikasi Database Management System (DBMS) yang sudah sangat banyak digunakan oleh para pemogram aplikasi web”.

Nugroho (2014:31), “MySQL adalah software atau program aplikasi database, yaitu software yang dapat kita pakai untuk menyimpan data berupa informasi teks dan juga angka”.

Dari definisi diatas, dapat disimpulkan bahwa MySQL adalah database yang biasa digunakan dalam menyimpan data dari sebuah program seperti visual basic. NET dan PHP.

2.5 Permodelan dengan Sistem UML

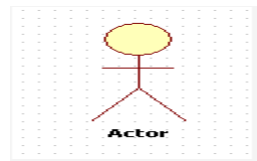
Menurut (Rosa A.S dan M. Shalahudin, 2014:133). UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

Menurut (Touseef, Anwer, Hussain, & Nadeem, 2015). *Unified Modeling Language* (UML) adalah Bahasa standar untuk pemodelan perangkat lunak dan untuk model proses bisnis dan memiliki muncul sebagai standar umum untuk pemodelan berorientasi objek.

2.5.1 Diagram Use Case

Menurut Sukamto dan Shalahuddin (2013:155), “Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat”. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case sebagai berikut :

1. Aktor : merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang. Yang disimbolkan sabagai berikut :



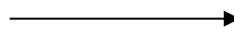
Gambar 2.1 Simbol Aktor

2. Use Case : merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.



Gambar 2.2 Simbol Use Case

3. Relasi : Merupakan garis atau simbol yang menghubungkan antara element use case dan aktor.



Gambar 2.3 Simbol Relasi

Relasi memiliki tipe relasi yang mungkin terjadi pada diagram Use Case :


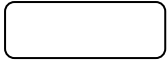
1. <<include>> yaitu kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya.
2. <<extends>> yaitu kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.

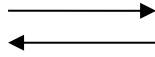



2.5.2 Diagram Activity

Menurut (Jeffry L Whitten , 2004) dalam Buku “ Systems Analysis and Design Methods”. Diagram Activity merupakan sebuah diagram yang dapat digunakan untuk menggambarkan secara grafis aliran proses bisnis, langkah-langkah sebuah use case atau logika behaviour (metode) object.

Komponen-komponen pembentuk diagram Activity sebagai berikut :

Tabel 2.1 Komponen Pembentuk Diagram Activity (Jeffry L Whitten , 2004)

NO	SIMBOL	KETERANGAN
1		Menggambarkan awal sebuah proses
2		Menggambarkan sebuah kegiatan atau tugas yang perlu dilakukan



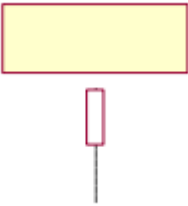
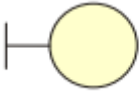
3		Menggambarkan sasaran yang mengawali kegiatan
4		Sebuah bar sinkronisasi. Simbol ini memperbolehkan anda untuk menggambarkan kegiatan yang dapat muncul secara paralel
5	[]	Menggambarkan sebuah sasaran yang merupakan sebuah hasil dari kegiatan keputusan
6		Menggambarkan sebuah kegiatan keputusan / tindakan yang harus diambil pada kondisi tertentu
7		Menggambarkan akhir dari sebuah proses







2.5.3 Diagram Sequence

Sukanto dan Shalahuddin (2013:165), “Sequence diagram atau diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirim dan diterima antar objek”. Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksinya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak use case yang didefinisikan maka diagram sekuen yang harus

dibuat juga semakin banyak. Adapun simbol simbol dari sequence diagram terdapat pada daftar simbol:

Tabel 2.2 Komponen Pembentuk Sequence Diagram

No	Simbol	Keterangan
1	 Aktor	Aktor merepresentasikan entitas yang berada di luar sistem dan berinteraksi dengan sistem . Mereka bisa berupa manusia , perangkat keras ataupun sistem yang lain.
2	 Lifeline	Fungsi dari simbol ini adalah Mengeksekusi objek selama sequence (message dikirim atau diterima dan aktifasinya).
3	 General	Fungsinya adalah Merepresentasikan entitas tunggal dalam sequence diagram . Entitas ini memiliki nama , <i>stereotype</i> atau berupa <i>instance (class)</i> .
4	 Boundary	Boundary biasanya berupa tepi dari sistem , seperti user interface atau suatu alat yang berinteraksi dengan sistem yang lain.

5	 <p>Control</p>	Control elemenr mengatur aliran dari informasi untuk sebuah skenario . Objek ini umumnya mengatur perilaku dan perilaku bisnis.
6	 <p>Entity</p>	Entitas biasanya elemen yang bertanggung jawab menyimpan data atau informasi . Ini dapat berupa <i>beans</i> atau <i>model object</i> .
7	 <p>Activation</p>	Yaitu suatu titik dimana sebuah objek mulai berpartisipasi di dalam sebuah sequence yang menunjukkan kapan sebuah objek mengirim atau menerima objek.
8	 <p>Message Entry</p>	Simbol ini berfungsi untuk menggambarkan pesa/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.
9	 <p>Message to Self</p>	Simbol ini menggambarkan pesa/hubungan objek itu sendiri , yang menunjukkan urutan kejadian yang terjadi.
10	 <p>Message Return</p>	Simbol ini menggambarkan hasil dari pengiriman message dan digambarkan dengan arah dari kanan ke kiri.