

## BAB II TINJAUAN PUSTAKA

### 2.1 Pengertian Sistem

Secara umum pengertian sistem adalah suatu kumpulan prosedur-prosedur yang tersusun atau terangkai secara sistematis yang saling berkaitan, dengan tujuan agar dapat memudahkan tujuan yang diinginkan dapat tercapai. Dalam Bidang sistem Informasi diartikan sebagai suatu cara tertentu untuk menyediakan informasi yang dibutuhkan oleh organisasi dengan cara sukses dan menguntungkan. Pada dasarnya sistem informasi merupakan suatu sistem yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi (Hajjah, 2019).

#### 2.1.1 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu Mempunyai komponen-komponen (*component*), batas sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengelola (*process*), dan sasaran (*objectives*), Sasaran (*goal*) (Midi, 2020) :

##### 1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan.

##### 2. Batasan Sistem

Batas sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

##### 3. Lingkungan Luar Sistem

Lingkungan luar (*environment*) dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.

##### 4. Penghubung Sistem

Penghubung (*interface*) merupakan media penghubung antara satu sub sistem dengan sub sistem yang lainnya.

## 5. Masukan Sistem

Masukan (*input*) adalah energi yang dimasukkan kedalam sistem.

## 6. Keluaran Sistem

Keluaran (*output*) adalah hasil dari energy yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

## 7. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran.

## 8. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objectives*).

### 2.2 Sistem Pendukung Keputusan

Sistem pendukung keputusan merupakan sistem *informasi* interaktif yang menyediakan *informasi*, permodelan, dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi yang tidak terstruktur. Aplikasi sistem penunjang keputusan tidak dimaksudkan untuk mengotomatisasikan pengambilan keputusan, tetapi memberikan perangkat interaktif yang memungkinkan pengambil keputusan untuk melakukan berbagai analisis menggunakan model-model yang tersedia (Midi, 2020). Bagi seorang ahli sendiri, sistem ini akan membantu aktivitasnya sebagai seorang asisten yang sangat berpengalaman. Sistem pendukung keputusan mengkombinasikan kaidah-kaidah penarikan kesimpulan (*inference rules*) dengan basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu (Zufria & Santoso, 2021).

Ada beberapa jenis keputusan berdasarkan sifat dan jenisnya, menurut Herbert A. Simon (Hasugian & Cipta, 2018):

#### 1. Keputusan terprogram

Yaitu keputusan yang bersifat berulang dan rutin, sedemikian sehingga suatu prosedur pasti telah dibuat untuk menanganinya.

## 2. Keputusan tidak terprogram

Yaitu keputusan yang bersifat baru, tidak terstruktur dan jarang konsekuensi. Tidak ada metode yang pasti untuk menangani masalah tersebut.

### 2.3 Metode SAW (Simple Additive Weighth)

*Simple Additive Weighth* merupakan metode *multiply attribute* yang digunakan dalam menyeleksi alternatif kemudian dilakukan perengkingan dan didapatkan alternatif terbaik (Afsha Zahara et al., 2022).

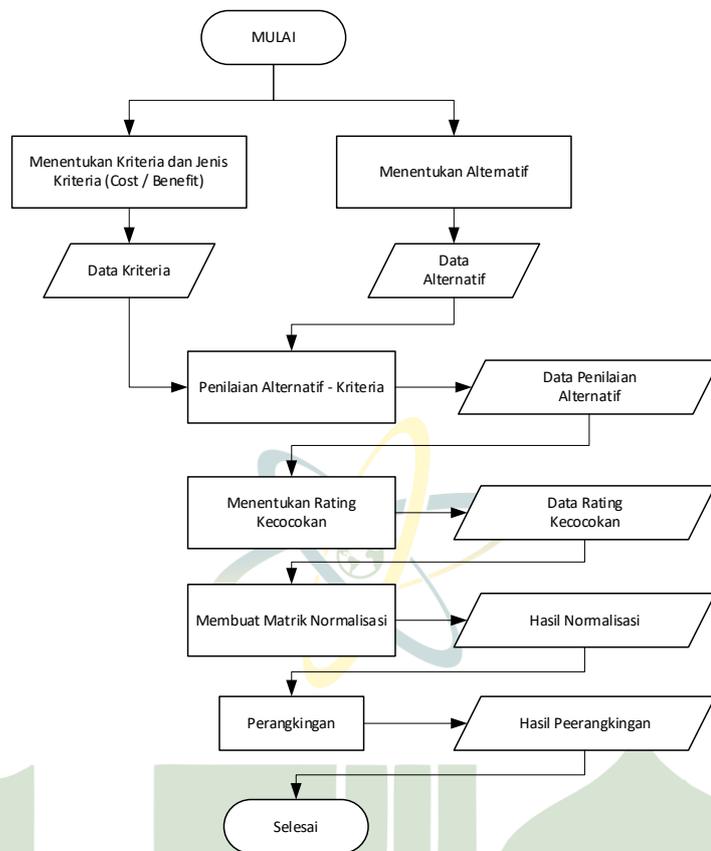
Metode ini adalah salah satu metode pendukung keputusan dengan menggunakan nilai bobot sederhana dengan fungsi utama untuk membuat atau mencari hasil perankingan atau disebut juga dengan skala prioritas untuk masing-masing alternatif. Persamaan pada metode SAW, yaitu:

$$rij = \frac{x_{ij}}{\text{Max}_i x_{ij}} \text{ (Jika Benefit)} \dots\dots\dots (2.1)$$

$$rij = \frac{\text{Min}_i x_{ij}}{x_{ij}} \text{ (Jika Cost)} \dots\dots\dots (2.2)$$

Keterangan:

- Rij = Nilai Bobot rating kinerja
- Max<sub>i</sub> = Nilai Maksimum x
- Min<sub>i</sub> = Nilai Minimum x
- Ij = Baris dan Kolom



**Gambar 2. 1 Flowchart Metode SAW**

Pada *flowchart* yang terlihat pada Gambar 2.1 proses awal dari metode SAW adalah dengan menentukan kriteria dan alternatif yang akan dinilai. Pada kriteria jika kriteria bersifat menguntungkan alternatif maka akan masuk kedalam jenis *benefit*, sedangkan jika bersifat merugikan alternatif maka akan masuk kedalam kategori *cost*. Pada tahapan selanjutnya adalah melakukan penilaian terhadap alternatif terhadap masing-masing kriteria, dilanjutkan dengan menentukan rating kecocokan dari masing-masing alternatif terhadap kriteria. Tahapan selanjutnya adalah membuat matrik normalisasi berdasarkan nilai matrik rating kecocokan, tahapan akhir dari metode ini adalah menentukan urutan ranking dari proses hasil perhitungan matrik normalisasi.

### 2.3.1 Contoh Perhitungan Metode SAW

Berikut pembahasan hasil analisis pengambilan keputusan pemilihan produk pandemi menggunakan metode *Simple Additive Weight* (SAW) :

**Tabel 2.1 Keterangan Kriteria Yang Digunakan**

<b>Kriteria</b>	<b>Keterangan</b>	<b>Jenis Kriteria</b>
C1	Harga (Rp)	<i>Cost</i>
C2	Lapisan	<i>Cost</i>
C3	Dapat dipakai berulang	<i>Cost</i>
C4	Efektifitas filtrasi	<i>Benefit</i>
C5	Kualitas	<i>Benefit</i>
C6	<i>Looks</i>	<i>Benefit</i>
C7	<i>Face seal fit</i>	<i>Benefit</i>
C8	Tingkat kenyamanan	<i>Benefit</i>

Berdasarkan Tabel 2.1 di atas kriteria yang digunakan ada 8 dengan masing-masing jenis kriterianya. Jenis kriteria ditentukan seorang pengambil keputusan dalam hal ini kelompok menentukan jenis kriteria berdasarkan sudut pandang kelompok dengan melihat masing-masing kriteria dimana terdapat 3 *Cost* dan 5 *Benefit* yaitu Harga (*cost*), Lapisan (*cost*), Dapat dipakai berulang (*cost*), Efektivitas filtrasi (*benefit*), Kualitas (*benefit*), *Looks* (*benefit*), *Face seal fit* (*benefit*), dan Tingkat kenyamanan (*benefit*).

1) Memberikan nilai pada setiap alternatif pada setiap kriteria yang sudah ditentukan.

Komponen kriteria diberi bobot sesuai dengan derajat. Nilai bobot komponen kriteria diperoleh dari hasil *browsing* dari media *internet*.

a) Pembobotan pada Kriteria Harga

**Tabel 2. 2 Pembobotan C1 = Harga**

<b>Kriteria</b>	<b>Nilai</b>	<b>Keterangan</b>
<Rp10.000	Sangat Murah	1
>Rp10.000-Rp30.000	Murah	0.8
>Rp30.000-Rp50.000	Sedang	0.6
>Rp50.000-Rp70.000	Mahal	0.4
>Rp70.000	Sangat Mahal	0.2

Berdasarkan Tabel 2.2 di atas dengan nilai sangat murah diberi bobot paling tinggi karena kriteria Harga termasuk *cost* dimana tujuannya yaitu untuk meminimalkan *cost*.

b) Pembobotan pada Kriteria Lapisan

**Tabel 2.3 Pembobotan C2 = Lapisan**

Kriteria	Nilai	Keterangan
5 lapis	SangatTebal	0.2
4 lapis	Tebal	0.4
3 lapis	Sedang	0.6
2 lapis	Tipis	0.8
1 lapis	SangatTipis	1

Berdasarkan Tabel 2.3 di atas dengan nilai sangat tebal diberi bobot paling rendah karena kriteria Lapisan termasuk *cost* dimana tujuannya yaitu untuk meminimalkan *cost*.

c) Pembobotan pada Kriteria Dapat Dipakai Berulang

**Tabel 2.4 Pembobotan C3 = Dapat Dipakai Berulang**

Kriteria	Nilai	Keterangan
<i>No</i>	Rendah	0.8
<i>Yes</i>	Tinggi	0.4

Berdasarkan Tabel 2.4 di atas dengan nilai rendah diberi bobot paling tinggi karena kriteria dapat dipakai berulang termasuk *cost* dimana tujuannya yaitu untuk meminimalkan *cost*.

d) Pembobotan pada Kriteria Efektivitas Filtrasi

**Tabel 2.5 Pembobotan C4 = Efektivitas Filtrasi**

Kriteria	Nilai	Keterangan
95-100	Sangat Tinggi	1
80-95	Tinggi	0.8
80-70	Cukup Tinggi	0.6

Kriteria	Nilai	Keterangan
50-70	Rendah	0.4
0-50	Sangat Rendah	0.2

Berdasarkan Tabel 2.5 di atas dengan nilai sangat tinggi diberi bobot paling tinggi karena dapat dipakai berulang termasuk *benefit* dimana tujuannya yaitu untuk memaksimalkan *benefit*.

e) Pembobotan pada Kriteria Kualitas

**Tabel 2. 6 Pembobotan C5 = Kualitas**

Kriteria	Nilai	Keterangan
>0.9	Tinggi	0.9
>0.1-0.9	Sedang	0.6
<0.1	Rendah	0.3

Berdasarkan Tabel 2.6 di atas dengan nilai sangat tinggi diberi bobot paling tinggi karena kriteria dapat dipakai berulang termasuk *benefit* dimana tujuannya yaitu untuk memaksimalkan *benefit*.

f) Pembobotan pada Kriteria *Looks*

**Tabel 2. 7 Pembobotan C6 = Looks**

Kriteria	Nilai	Keterangan
<i>Excellent</i>	Sangat Tinggi	1
<i>Good</i>	Tinggi	0.8
<i>Average</i>	Cukup Tinggi	0.6
<i>Below Average</i>	Rendah	0.4
<i>Low</i>	Sangat Rendah	0.2

Sesuai dengan Tabel 2.7 di atas dengan nilai sangat tinggi diberi bobot paling tinggi karena kriteria *Looks* termasuk *benefit* dimana tujuannya yaitu untuk memaksimalkan *benefit*.

g) Pembobotan pada Kriteria *Face Seal Fit*

**Tabel 2. 8 Pembobotan C7 = *Face Seal Fit***

Kriteria	Nilai	Keterangan
Longgar	Rendah	0.5
Ketat	Tinggi	1

Berdasarkan Tabel 2.8 di atas dengan nilai rendah diberi bobot paling rendah karena kriteria *Face Seal Fit* termasuk *benefit* dimana tujuannya yaitu untuk memaksimalkan *benefit*.

h) Pembobotan pada Kriteria Tingkat Kenyamanan

**Tabel 2. 9 Pembobotan C8 = Tingkat Kenyamanan**

Kriteria	Nilai	Keterangan
Kurang Nyaman	Sangat Rendah	0.2
Cukup Nyaman	Rendah	0.4
Cukup Nyaman	Cukup Tinggi	0.6
Nyaman	Tinggi	0.8
Sangat Nyaman	Sangat Tinggi	1

Berdasarkan Tabel 2.9 di atas dengan nilai rendah diberi bobot Sangat rendah karena kriteria Tingkat Kenyamanan termasuk *benefit* dimana tujuannya yaitu untuk memaksimalkan *benefit*.

2) Memberikan nilai bobot (W)

Pada metode *Simple Additive Weighting* (SAW), kita harus memberikan nilai bobot (w). Nilai bobot yang di dapat pada pemilihan produk pandemi dapat dilihat pada Tabel 2.10 dibawah ini :

**Tabel 2. 10 Bobot Untuk Setiap Kriteria**

Kriteria	Bobot
C1	12.5%
C2	12.5%
C3	12.5%

Kriteria	Bobot
C4	12.5%
C5	12.5%
C6	12.5%
C7	12.5%
C8	12.5%

Untuk menentukan bobot dari setiap kriteria dengan membagi 100% dengan jumlah kriteria yang digunakan sehingga diperoleh untuk setiap kriteria memiliki bobot sebesar 12.5%. Kemudian membuat penilaian dari setiap alternatif pada setiap kriteria yang dapat dilihat pada Tabel 2.11 di bawah ini.

**Tabel 2. 11 Data Penilaian**

Alternatif	Kriteria							
	C1	C2	C3	C4	C5	C6	C7	C8
K1	Rp40,000	5 lapis	Tidak	95-100	0,98	<i>Below Average</i>	Ketat	Kurang Nyaman
K2	Rp75,000	3 lapis	Tidak	80-95	0,95	<i>Good</i>	Longgar	Cukup Nyaman
K3	Rp22,500	2 lapis	Tidak	80-95	0,4	<i>Below Average</i>	Longgar	Cukup Nyaman
K4	Rp12,000	3 lapis	Ya	50-70	0,25	<i>Excelent</i>	Longgar	Nyaman
K5	Rp6000	1 lapis	Ya	0-50	0,025	<i>Excelent</i>	Longgar	Sangat Nyaman

Dari Tabel 2.11 di atas penilaian dari setiap alternatif pada setiap kriteria ditentukan berdasarkan nilai yang sudah ditentukan sebelumnya.

3) Penentuan rating kecocokan setiap alternatif

Dengan setiap kriteria dalam penentuan rating kecocokan maka nilai dari masing-masing kriteria dimasukkan kedalam Tabel rating kecocokan yang telah disesuaikan dengan nilai dari Tabel kriteria. Maka Gambar rating kecocokan dapat dilihat sebagai berikut.

**Tabel 2. 12 Rating Kecocokan**

Alternatif	Kriteria							
	C1	C2	C3	C4	C5	C6	C7	C8
K1	0.6	0.2	1	1	0.9	0.4	1	0.2
K2	0.2	0.4	1	0.8	0.9	0.8	0.5	0.4
K3	0.8	0.6	1	0.6	0.6	0.4	0.5	0.4
K4	0.8	0.8	0	0.4	0.6	1	0.5	0.8
K5	1	1	0	0.2	0.3	1	0.5	1

4) Normalisasi matriks

Membuat Normalisasi Matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan maupun atribut biaya) sehingga diperoleh matriks ternormalisasi R. Setelah ditentukan nilai R dengan rumus maka akan diperoleh hasil pada Tabel 2.13 sebagai berikut: .

**Tabel 2. 13 Normalisasi Matriks**

Alternatif	Kriteria							
	C1	C2	C3	C4	C5	C6	C7	C8
K1	0.3333	1	1	1	1	0.4	1	0.2
K2	1	0.5	1	0.8	1	0.8	0.5	0.4
K3	0.25	0.3333	1	0.6	0.6667	0.4	0.5	0.4
K4	0.25	0.25	1	0.4	0.6667	1	0.5	0.8
K5	0.2	0.2	1	0.2	0.3333	1	0.5	1

Maka R13, R23, R33, R43, dan R53 memiliki nilai R yang sama sebab semua angka pada kolom tersebut memiliki nilai yang sama. Jadi,  $R_{13}=R_{23}=R_{33}=R_{43}=R_{53}=1$ . Dari perhitungan diatas diperoleh matriks R yang ditunjukkan pada Tabel 2.14 sebagai berikut:

**Tabel 2. 14 Tabel Pembulatan Normalisasi**

0.3	1	0.5	1	1	0.4	1	0.2
1	0.5	0.5	0.8	1	0.8	0.5	0.4
0.3	0.3	0.5	0.6	0.6	0.4	0.5	0.4

0.3	0.25	1	0.4	0.6	1	0.5	0.8
0.2	0.2	1	0.2	0.3	1	0.5	1

5) Melakukan proses perangkingan

Melakukan proses perangkingan dengan menggunakan persamaan. Dari proses perhitungan nilai akhir maka didapatkan nilai pada Tabel 2.15 di bawah ini.

**Tabel 2. 15 Data Proses Perangkingan**

Alternatif	Kriteria								Total
	C1	C2	C3	C4	C5	C6	C7	C8	
K1	0.0417	0.125	0.0625	0.13	0.125	0.05	0.125	0.025	0.6792
K2	0.125	0.0625	0.0625	0.1	0.125	0.1	0.0625	0.05	0.6875
K3	0.0313	0.0417	0.0625	0.08	0.0833	0.05	0.0625	0.05	0.4563
K4	0.0313	0.0313	0.125	0.05	0.0833	0.125	0.0625	0.1	0.6083
K5	0.025	0.025	0.125	0.03	0.0417	0.125	0.0625	0.125	0.5542

Berdasarkan Tabel 2.15 di atas total nilai tertinggi yaitu sebesar 0.6792. Dari hasil perhitungan nilai  $V_i$  dari setiap alternatif yang akan menjadi pilihan produk pandemi maka dapat dibuatkan Tabel penentuan ranking sebagai berikut :

**Tabel 2. 16 Perangkingan**

Alternatif	Total	Rank
Masker N95	0.679166667	2
Masker Medis	0.6875	1
Masker Karbon Aktif	0.45625	5
Masker Kain	0.608333333	3
Masker Spons ( <i>Fashion</i> )	0.554166667	4

Maka alternatif yang memiliki nilai tertinggi yaitu A2 (masker medis) dengan nilai 0.6875 bisa dijadikan sebagai data pertimbangan untuk dipilih menjadi produk pandemi di masa Covid-19.

## 2.4 Metode MAUT (Multi Attribute Utility Theory)

Pada metode ini untuk menyelesaikan permasalahan dengan menggunakan skema evaluasi akhir dari suatu objek atau dapat dikatakan dengan nilai bobot yang dijumlahkan dengan nilai relevan tiap *utility*-nya. Persamaan pada metode ini yang ditunjukkan pada persamaan 2.3 yaitu :

$$U(x) = \frac{x - xi^-}{xi^+ - xi^-} \dots\dots\dots (2.3)$$

Keterangan :

$xi^-$  = Nilai terburuk pada x

$xi^+$  = Nilai terbaik pada x

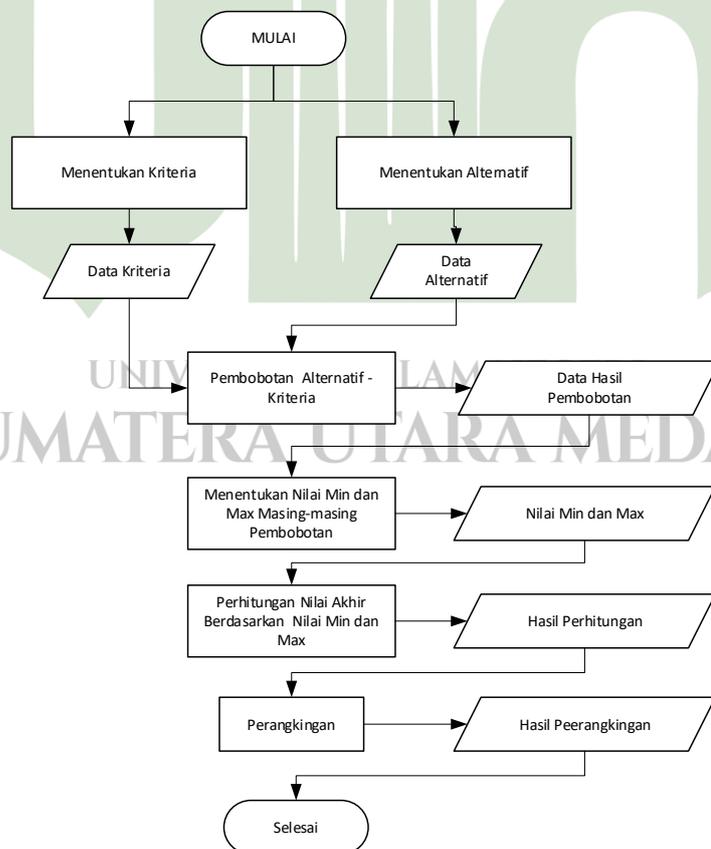
Setelah itu mencari nilai alternatif dengan persamaan 2.4 sebagai berikut:

$$(x) = \sum_{n=1}^n wi vi (x) \dots\dots\dots (2.4)$$

Keterangan :

$wi$  = Bobot relatif atribut ke -i

$vi$  = Utility dari hasil x untuk tiap-tiap atribut



## Gambar 2. 2Flowchart Metode MAUT

Pada *flowchart* metode MAUT seperti yang terlihat pada Gambar 2.2 tahap perhitungan yang dilakukan memiliki kemiripan dengan metode SAW. Hal yang membedakan antara metode SAW dan MAUT adalah penggunaan variabel perkalian dan pembagian yang dipergunakan. Pada metode SAW menggunakan variabel *benefit* dan *cost* sebagai variabel perhitungan, sedangkan pada metode MAUT menggunakan nilai max dan min dari setiap kolom pembobotan alternatif pada setiap kriteria.

### 2.4.1 Contoh Perhitungan Metode MAUT

Berikut ini adalah contoh perhitungan SPK dengan menggunakan metode MAUT dalam pemilihan produk *deodorant*. Setiap kriteria dilambangkan dengan C1,C2,C3, dan C4, Adapun kriteria yang digunakan ditunjukkan pada Tabel 2.17 sebagai berikut:

Tabel 2. 17 Kriteria dan Bobot Kriteria

Kriteria	Sub Kriteria	Nilai	Bobot
Harga	< 25000	80	15%
	25000 – 50000	50	
	> 50000	30	
Jenis	Spray	80	20%
	Roll On	30	
	Stick	50	
Kandungan	Tinggi	30	40%
	Cukup	50	
	Rendah	80	
Daya Tahan	< 18 Jam	30	25%
	18 - 24 Jam	50	
	> 24 Jam	80	

Tabel 2. 18 Matrix Keputusan

Alternatif	Kriteria			
	C1	C2	C3	C4
<i>Fresh Sugar Roll On Deodorant</i>	30	30	80	80

<i>Rexona Free Spirit</i>	50	30	50	50
<i>Dove Dry Spry Shooting Chamomile</i>	30	80	30	80
<i>Nature's Gate Organics Fruit Blend</i>	30	50	80	80
<i>Nivea Deodorant Dry Comfort Plus</i>	50	30	50	80
<i>Sebamed Balsam Deodorant</i>	30	30	50	50
<i>Avon On Duty Deodorant</i>	30	30	50	80
<b>MAX</b>	50	80	80	80
<b>MIN</b>	30	30	30	50

Tabel 2.18 adalah merupakan Tabel matrik keputusan yang isinya adalah berupa penilaian dari masing-masing bobot alternatif berdasarkan Tabel 2.17.

**Tabel 2. 19 Matrix Normalisasi**

Alternatif	Kriteria			
	C1	C2	C3	C4
<i>Fresh Sugar Roll On Deodorant</i>	0	0	1	1
<i>Rexona Free Spirit</i>	1	0	0.4	0
<i>Dove Dry Spry Shooting Chamomile</i>	0	1	0	1
<i>Nature's Gate Organics Fruit Blend</i>	0	0.4	1	1
<i>Nivea Deodorant Dry Comfort Plus</i>	1	0	0.4	1
<i>Sebamed Balsam Deodorant</i>	0	0	0.4	0
<i>Avon On Duty Deodorant</i>	0	0	0.4	1

Matrik normalisasi merupakan hasil perhitungan dari matrik keputusan dengan menggunakan persamaan 2.3, nilai maks adalah merupakan nilai tertinggi dari kriteria ada setiap alternatif, nilai min adalah merupakan nilai terendah dari masing-masing alternatif yang ditunjukkan pada Tabel 2.20 sebagai berikut:

**Tabel 2. 20 Hasil Perangkingan**

Alternatif	Jumlah	Rangking
<i>Fresh Sugar Roll On Deodorant</i>	0.65	2
<i>Rexona Free Spirit</i>	0.31	6
<i>Dove Dry Spry Shooting Chamomile</i>	0.45	4
<i>Nature's Gate Organics Fruit Blend</i>	0.73	1
<i>Nivea Deodorant Dry Comfort Plus</i>	0.56	3
<i>Sebamed Balsam Deodorant</i>	0.16	7
<i>Avon On Duty Deodorant</i>	0.41	5

Tabel 2.20 adalah merupakan hasil perhitungan akhir dari proses metode MAUT. Hasil dari kolom jumlah adalah merupakan hasil dari penjumlahan seluruh kriteria dari masing-masing alternatif, sedangkan urutan perankingan adalah merupakan urutan dari perolehan nilai tertinggi hingga nilai terendah.

## 2.5 Tanaman Hias

Tanaman hias adalah tanaman yang mempunyai nilai keindahan baik bentuk, warna daun, tajuk maupun bunganya, sering digunakan untuk penghias pekarangan dan lain sebagainya. Tanaman hias cukup diminati di Indonesia hal ini dibuktikan dengan luas lahan dan produksi tanaman hias yang terus ada peningkatan setiap tahunnya. Bahkan, sudah di ekspor ke berbagai negara diantaranya ke Amerika, Jepang, Korea dan Singapura. Berdasarkan satuan luas panen dan bentuk hasilnya, tanaman hias dapat dibagi menjadi tiga kelompok, yakni: kelompok bunga potong, kelompok tanaman hias dalam pot dan kelompok tanaman hias lainnya. Tanaman hias dalam pot adalah salah satu kelompok tanaman hias yang cukup populer, karena mudah untuk didapatkan, bersih, memiliki pilihan desain pot yang banyak dan unik, mudah untuk dipindahkan lokasinya dan tidak harus memiliki ruangan yang besar (Fathurrahmani & Agustiannoor, 2019).

## 2.6 Website

*Website* atau disingkat *web*, dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi *informasi* dalam bentuk data digital baik berupa text, gambar, video, audio dan animasi lainnya yang disediakan melalui jalur koneksi internet. Pengertian lain dari *website* adalah kumpulan halaman-halaman yang digunakan untuk menampilkan *informasi* teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman (Manullang et al., 2021).

Dari pengertian *website* tersebut dapat dibedakan menjadi 2 yaitu *web* bersifat statis dan dinamis. Bersifat statis apabila isi *informasinya* tetap dan isi

informasinya hanya dari pemilik *website* sedangkan *web* yang bersifat dinamis apabila isi informasinya selalu berubah-ubah dan dapat diubah-ubah oleh pemilik maupun pengguna *website*. Contoh *web* statis: *website* profil perusahaan, sedangkan contoh *web* dinamis seperti facebook, twitter dan lain-lain (Manullang et al., 2021).

## 2.7 PHP (Hypertext Preprocessor)

Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat *website* yang bersifat *server-side scripting*. PHP bersifat dinamis. PHP dapat dijalankan pada berbagai macam sistem operasi seperti *Windows*, *Linux*, dan *Mac Os*. Selain Apache, PHP juga mendukung beberapa *web server* lain, seperti *Microsoft ISS*, *Caudium*, dan *PWS*. PHP dapat memanfaatkan *database* untuk menghasilkan halaman *web* yang dinamis. Sistem manajemen *database* yang sering digunakan bersama PHP adalah *MYSQL*. Namun, PHP juga mendukung sistem manajemen *Database Oracle*, *Microsoft Access*, *Interbase*, *d-Base*, dan *PostgreSQL* (Saed Novendri et al., n.d.).

Ketika dipanggil dari *web browser*, program yang ditulis dengan PHP akan di-*parsing* di dalam *web server* oleh *interpreter* PHP dan diterjemahkan kedalam dokumen HTML, yang selanjutnya akan ditampilkan kembali *web server*. Karena pemrosesan program PHP dilakukan didalam lingkungan *web browser*, 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. Penulisan *script* PHP diawali dengan tanda lebih kecil (<) dan diakhiri tanda lebih besar (>) (Saed Novendri et al., 2019).

## 2.8 CSS (Cascade Style Sheet)

CSS merupakan kependekan dari *Cascading Style Sheet* yang berfungsi untuk mengatur tampilan dengan kemampuan jauh lebih baik dari tag maupun atribut standar HTML (*Hypertext Markup Language*). CSS sebenarnya adalah suatu kumpulan atribut untuk fungsi *format* tampilan dan dapat digunakan untuk

mangontrol tampilan banyak dokumen secara bersamaan. Keuntungan menggunakan CSS yaitu jika ingin mengubah *format* dokumen, maka tidak perlu mengedit satu persatu (Saed Novendri et al., 2019).

## 2.9 HTML (Hypertext Markup Language)

*Hypertext Merkup Languange* (HTML) adalah bahasa standard yang digunakan untuk menampilkan halaman *web*". Yang bisa dilakukan dengan HTML yaitu (Saed Novendri et al., 2019) :

- a. Mengatur tampilan dari halaman *web* dan isinya.
- b. Membuat Tabel dalam halaman *web*.
- c. Mempublikasikan halam *web* secara *online*.
- d. Membuat *form* yang bisa digunakan untuk menangani registrasi dan transaksi via *web*. Contoh: Setiap dokumen HTML diawali dan diakhiri dengan tag HTML.

## 2.10 MySQL Database

MySQL merupakan sebuah konsep pengoprassian *database*, terutama ntuk pemilihan atau seleksi pemasukkan data, yang memungkinkan pengoprassian dikerjakan lebih mudah dikerjakan secara otomatis untuk dan mengelolah *database* dari yang kecil sampe dengan yang besar (Husna et al., 2023).

MySQL termasuk jenis RDBMS (*Relational Database Manajement System*). MySQL mendukung bahasa pemrograman PHP, bahasa permintaan yang terstruktur, karena pada penggunaannya SQL memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama ANSI. MySQL merupakan RDBMS (*Relational Database Management System*) *server*. RDBMS adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan data pada suatu model *relational*. Dengan demikian, Tabel-Tabel yang ada pada *database* memiliki relasi antara satu Tabel dengan Tabel lainnya. Beberapa keunggulan dari MySQL yaitu (Hermiati et al., 2021) :

- a. Cepat, handal dan mudah dalam penggunaannya. MySQL lebih cepat tiga sampai empat kali dari pada *database server* komersial yang beredar saat

ini, mudah diatur dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan MySQL.

- b. Didukung oleh berbagai bahasa *Database Server* MySQL dapat memberikan pesan *Error* dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.
- c. Mampu membuat Tabel berukuran sangat besar. Ukuran maksimal dari setiap Tabel yang dapat dibuat dengan MySQL adalah 4 GB sampai dengan ukuran file yang dapat ditangani oleh sistem operasi yang dipakai.
- d. Lebih murah MySQL bersifat open source dan didistribusikan dengan gratis tanpa biaya untuk UNIX platform, OS/2 dan *Windows Platform*.

### 2.10.1 Perintah DDL, DML dan DCL

Secara umum, SQL terdiri dari tiga bahasa, yaitu *Data Definition Language* (DDL), *Data Manipulation Language* (DML) dan *Data Control Language* (DCL). Implementasi DDL, DML dan DCL berbeda untuk tiap sistem manajemen basis data (SMBD), namun secara umum implementasi tiap bahasa ini memiliki bentuk standar yang ditetapkan ANSI. Jadi, perintah SQL pada semua *software database* hampir sama.

#### 1. DDL (*Data Definition Language*)

DDL merupakan perintah SQL yang berhubungan dengan pendefinisian suatu struktur *database* yang digunakan untuk membuat, mengubah dan menghapus struktur dan definisi metadata dari objek-objek *database*. Beberapa perintah dasar yang termasuk DDL ini antara lain :

- a. *CREATE* : Perintah ini digunakan untuk membuat, termasuk diantaranya membuat *database* baru, Tabel baru, *view* baru, dan kolom. Contoh : *CREATE DATABASE* nama\_ *database*.
- b. *ALTER* : Perintah ini digunakan untuk mengubah struktur Tabel yang telah dibuat. Pekerjaannya mencakup mengganti nama Tabel, menambah kolom, mengubah kolom, menghapus kolom, maupun memberikan atribut pada kolom. Contoh: *ALTER TABEL* nama\_ *Tabel* *ADD* nama\_ *kolom datatype*

- c. *RENAME* : Perintah yang digunakan untuk merubah nama Objek
- d. *DROP* : Perintah ini digunakan untuk menghapus *database* dan Tabel.  
Contoh: *DROP DATABASE* nama\_ *database*

## 2. DML (*Data Manipulation Language*)

DML merupakan perintah SQL yang digunakan untuk proses pengolahan isi data di dalam Tabel seperti memasukkan, merubah dan menghapus isi data dan tidak terkait dengan perubahan struktur dan definisi tipe data dari objek *database*. Perintah SQL yang termasuk dalam DML antara lain :

- a. *UPDATE* : Perintah ini digunakan untuk memperbaharui data lama menjadi data terkini. Jika anda memiliki data yang salah atau kurang *Up To Date* dengan kondisi sekarang, maka dapat diubah isi datanya dengan menggunakan perintah *UPDATE*. Contoh : *UPDATE* nama\_ *Tabel SET* kolom1=data1, kolom2=data2,... *WHERE* kolom=data;
- b. *INSERT* : perintah ini digunakan untuk menyisipkan atau memasukkan data baru ke dalam Tabel. Penggunaannya setelah *database* dan Tabel selesai dibuat. Contoh: *INSERT INTO* nama\_ *Tabel VALUES* (data1, data2, dst...);
- c. *SELECT* : Perintah ini digunakan untuk mengambil data atau menampilkan data dari satu Tabel atau beberapa Tabel dalam relasi. Data yang diambil dapat kita tampilkan dalam layar *prompt* MySQL secara langsung maupun ditampilkan pada tampilan aplikasi. Contoh : *SELECT* nama\_ *kolom1, nama\_ kolom2 FROM* nama\_ *Tabel*;
- d. *DELETE* : Perintah ini digunakan untuk menghapus data dari Tabel. Biasanya data yang dihapus adalah data yang tidak diperlukan lagi. Pada saat menghapus data, perintah yang telah dijalankan tidak dapat digagalkan, sehingga data yang telah hilang tidak dapat dikembalikan lagi. Contoh : *DELETE FROM* nama\_ *Tabel WHERE* kolom=data;

### 3. DCL (*Data Control Language*)

DCL merupakan perintah SQL yang berhubungan dengan manipulasi user dan hak akses (*priviledges*). Perintah SQL yang termasuk dalam DCL antara lain :

- a. *GRANT* : Perintah ini digunakan untuk memberikan hak / izin akses oleh administrator (pemilik utama) *server* kepada *user* (pengguna biasa). Hak akses tersebut berupa hak membuat (*CREATE*), mengambil (*SELECT*), menghapus (*DELETE*), mengubah (*UPDATE*) dan hak khusus berkenaan dengan sistem *database*-nya.
- b. *REVOKE* : perintah ini memiliki kegunaan terbalik dengan *GRAND*, yaitu untuk menghilangkan atau mencabut hak akses yang telah diberikan kepada *user* oleh *administrator* (Yandi Nyong, 2019).

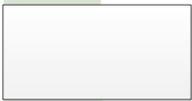
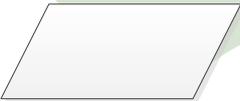
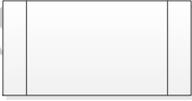
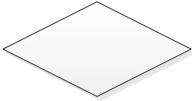
### 2.11 *Flowchart*

*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Ada lima macam bagan alir, di antaranya

1. Bagan Alir Sistem (*system flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem.
2. Bagan Alir Dokumen (*document flowchart*) disebut juga bagan alir *formulir* (*form flowchart*) merupakan bagan alir yang menunjukkan arus dari laporan dan *formulir* termasuk tembusan-tembusannya.
3. Bagan Alir Skematik (*schematic flowchart*) merupakan bagan alir yang mengGambarkan prosedur di dalam sistem dengan menggunakan simbol-simbol bagan alir sistem dan Gambar-Gambar komputer serta peralatan lainnya yang digunakan oleh sistem.
4. Bagan Alir Program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program.
5. Bagan Alir Proses (*process flowchart*) merupakan bagan alir yang banyak

digunakan di teknik industri untuk mengGambarkan proses dalam suatu prosedur (Verawati & Liksha, 2018).

**Tabel 2. 21 Simbol-simbol *Flowchart***

Simbol	Nama	Fungsi
	<i>TERMINATOR</i>	Permulaan/akhir program
	GARIS ALIR ( <i>FLOW LINE</i> )	Arah aliran program
	<i>PREPARATION</i>	Proses inisialisasi/pemberian harga awal
	PROCESS	Proses perhitungan/proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINE PROCESS	Permulaan sub program/proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjut

	<p style="text-align: center;">ON PAGE CONNECTOR</p>	<p>Penghubung bagian-bagian flowchat yang berada pada satu halaman</p>
	<p style="text-align: center;">OFF PAGE CONNECTOR</p>	<p>Penghubung bagian-bagian flowchat yang berada pada halaman berbeda</p>

## 2.12 UML (Unified Modelling Language)

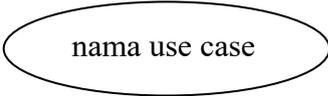
*Unified Modeling Language* merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah *software* yang berorientasikan pada objek. UML merupakan sebuah standar penulisan atau semacam *blue print* dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik (Simatupang & Sianturi, 2019).

*Unified Modeling Language* (UML) adalah sebuah bahasa yang berdasarkan grafik atau Gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software* (Suendri, 2018).

### 2.12.1 Use Case Diagram

*Use Case Diagram* merupakan pemodelan untuk kelakuan sistem informasi yang akan dibangun. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibangun. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada pada sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

**Tabel 2.21 Simbol *Use Case Diagram***

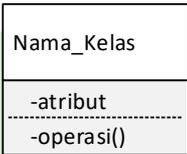
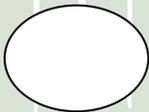
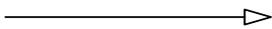
Nama	Simbol	Deskripsi
<i>Use Case</i>		<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor: biasanya dinyatakan dengan menggunakan kata kerja diawal di awal frase nama <i>use case</i>.</p>
Aktor/ <i>actor</i>		<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat</p>
Asosiasi/ <i>association</i>		<p>Komunikasi antara aktor dan use case yang berpartisipasi pada <i>use case</i> atau use case memiliki interaksi dengan aktor.</p>
Ekstensi/ <i>extend</i> << <i>extends</i> >>		<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan misal Arah panah mengarah pada <i>use case</i> yang ditambahkan.</p>
Generalisasi / <i>generalization</i>		<p>Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum lainnya.</p>

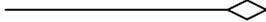
Nama	Simbol	Deskripsi
Menggunakan/ <i>include</i>		<i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan

### 2.12.2 Class Diagram

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

**Tabel 2.22 Simbol Class Diagram**

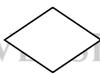
Nama	Simbol	Diskripsi
Kelas		Kelas pada struktur sistem.
Antar muka/ <i>interface</i>		Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
Asosiasi / <i>association</i>		Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Asosiasi berarah / <i>directed association</i>		Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Generalisasi		Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).
Kebergantungan/ <i>dependency</i>		Relasi antar kelas dengan makna kebergantungan antar kelas

Agregasi <i>/aggregation</i>		Relasi antar kelas dengan makna semua- bagian ( <i>whole-part</i> ).

### 2.12.3 Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Penekanan pada diagram aktivitas adalah menggambarkan aktivitas sistem atau aktivitas yang dapat dilakukan oleh sistem, bukan apa yang dilakukan *actor* yang ditunjukkan pada Tabel 2.23 sebagai berikut:

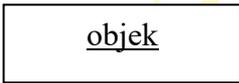
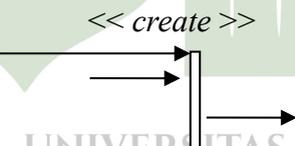
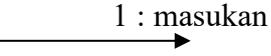
**Tabel 2.23 Simbol Activity Diagram**

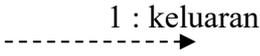
Nama	Simbol	Deskripsi
Status awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>decision</i>		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan <i>/ Join</i>		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir

### 2.12.4 Sequence Diagram

*Sequence Diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek yang ditunjukkan pada Tabel 2.24 sebagai berikut:

**Tabel 2.24 Simbol *Sequence Diagram***

Nama	Simbol	Deskripsi
Aktor		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
Garis hidup / <i>lifeline</i>		Menyatakan kehidupan suatu objek.
Objek		Menyatakan objek yang berinteraksi pesan.
Waktu aktif		Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
Pesan tipe <i>create</i>		Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.
Pesan tipe <i>send</i>		Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.

Nama	Simbol	Deskripsi
Pesan tipe <i>return</i>		Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan tipe <i>destroy</i>		Menyatakan suatu objek mengakhiri hidup objek lain, arah panah yang mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

### 2.13 Penelitian Terkait

Penelitian ini dilaksanakan berdasarkan beberapa penelitian terdahulu. Beberapa penelitian terdahulu yang dijadikan acuan pada penelitian ini dapat dilihat pada Tabel 2.25.

**Tabel 2.25 Penelitian Terdahulu**

No	Peneliti	Judul	Kesimpulan
1	Manullang, M. A., & Fahmi, H. (2021)	Sistem Pendukung Keputusan Pemberian Kredit Sepeda Motor Pada PT Adira Finance Medan Menggunakan Metode SAW. Jurnal Nasional Komputasi dan Teknologi Informasi, 4(2).	Penggunaan Metode <i>Simple Additive Weighting</i> dalam penelitian ini mampu memberikan hasil keputusan dalam Sistem Pendukung Keputusan Pemberian Kredit Sepeda Motor Pada PT Adira Finance Medan. Kriteria dalam penelitian ini dijelaskan lebih terperinci

No	Peneliti	Judul	Kesimpulan
			<p>supaya tidak menimbulkan pertanyaan dari pihak manapun. Mempermudah PT Adira Finance Medan untuk menentukan penerimaan Kredit Sepeda Motor bagi Nasabah baru dan dapat di print.</p>
2	<p>Muqorobin, M., et. al. (2019)</p>	<p>Sistem Pendukung Keputusan Penerimaan Beasiswa dengan Metode SAW. Respati, 14(1).</p>	<p>Penerapan metode SAW dapat menghasilkan perbandingan sebagai rujukan terbaik bagi calon penerima beasiswa dan sebagai pendukung dalam pengambilan keputusan oleh Tim Seleksi Beasiswa. Metode SAW dapat menerapkan jenis beasiswa BP dan BKM sesuai dengan kriteria dan bobot yang telah ditentukan. Sistem Pendukung keputusan ini juga dapat menentukan penerima beasiswa berdasarkan jumlah kuota sesuai dengan kebutuhan. Pengujian akhir dengan uji validistas telah menunjukkan bahwa Aplikasi sistem</p>

No	Peneliti	Judul	Kesimpulan
			<p>pendukung keputusan dengan metode SAW dinyatakan Valid karena berdasarkan hasil perhitungan manual dengan program diperoleh hasil yang sama, Sehingga sistem dinyatakan valid.</p>
3	<p>Mahendra, G. S., &amp; Aryanto, K. Y. E. (2019)</p>	<p>SPK Penentuan Lokasi ATM Menggunakan Metode AHP Dan SAW. <i>Jurnal Nasional Teknologi dan Sistem Informasi</i>, 5(1), 49-56.</p>	<p>Rancang bangun SPK Penentuan Lokasi ATM menggunakan AHP dan SAW dapat diimplementasikan dan dilakukan analisis terhadap hasil rekomendasi dibandingkan dengan data realisasi deployment ATM dan dapat membantu pengambil keputusan dalam masalah penentuan lokasi ATM secara cepat dan mudah.</p>
4	<p>Najar, A., et. al. (2021)</p>	<p>Sistem Pendukung Keputusan Perekrutan Anggota Bem Menggunakan Metode Saw Dan Topsis. <i>Jurnal Tekinkom (Teknik Informasi dan Komputer)</i>, 4(1), 18-24.</p>	<p>Penggunaan sistem ini dapat mempermudah BEM dalam melakukan perangkan perekrutan anggota BEM dengan cepat dan efisien. SPK ini menggunakan metode Topsis dan SAW</p>

No	Peneliti	Judul	Kesimpulan
			dengan kriteria yang terdiri IPK, Nilai Wawancara, Sertifikat Pendukung, Pengalaman Berorganisasi, Komitmen
5	Aisyah, S. (2019)	Aplikasi Sistem Pendukung Keputusan Analisis Kelayakan Pemberian Kredit Menggunakan Metode Saw Pada Perusahaan Leasing. Jurnal Teknovasi: Jurnal Teknik dan Inovasi Mesin Otomotif, Komputer, Industri dan Elektronika, 6(1), 1-16.	Aplikasi untuk membantu proses penilaian kelayakan pemberian kredit dengan menggunakan metode SAW telah diselesaikan dan berjalan dengan baik. Pengguna dapat menyesuaikan nilai crisp dan nilai bobot dari masing- masing kriteria serta kriteria kredit yang bermasalah di dalam aplikasi, sehingga proses penilaian dapat disesuaikan dengan kebutuhan.