

BAB II

LANDASAN TEORI

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan di gunakan sebagai sumber acuan yang relavan dan terkini yaitu :

Tabel II.1. Penelitian Terdahulu

No	Penulis	Judul	Hasil Penelitian
1	Wina Yusnaeni (2020)	MABAC Method Dalam Penentuan Kelayakan Penerima Bantuan SPP	Dalam pendataan pihak panitia mendata secara satu persatu serta dilakukan survei dan keputusan dari pihak panitia. Perlunya waktu dalam penginputan dan keputusan yang hanya dilakukan oleh pihak panitia tidak menjamin keputusan diambil secara tepat dan akurat. Oleh karena itu, perlunya sistem pendukung keputusan yang bisa membantu menentukan hasil yang lebih objektif sesuai dengan kriteria yang telah di tentukan. Salah satu metode yang dibahas penulis disini MABAC Method. Adanya kestabilan dalam solusi dan rasional dibandingkan beberapa metode sistem pendukung keputusan lainnya. Penggunaan metode dilakukan terhadap 128 pendaftar siswa dengan data kriteria yang telah ditentukan. Dari data tersebut di tentukan bobot penilaian serta bobot alternative terhadap kriteria yang menghasilkan layak tidak layaknya siswa menerima beasiswa atau bantuan SPP
2	Hengky Yosafat (2020)	Perancangan Sistem Pendukung Keputusan Pembelian Laptop Dengan Metode Mabac (Studi Kasus: Sigma	Hal ini membuat toko sigma komputer memerlukan sebuah sistem yang dapat membantu pemilik sekaligus pembeli untuk menentukan keputusan dalam pemilihan laptop yang akan dibeli secara cepat dan efisien. Sistem dibangun dengan menggunakan bahasa pemrograman C# dan database SQLite, serta menggunakan metode Mabac (Multi-Attributive Border Approximation Area Comparison) yang

		Komputer)	digunakan dalam perhitungan pada sistem pendukung keputusan. Data yang diperlukan berupa data laptop beserta spesifikasinya. Hasil dari perancangan sistem pendukung keputusan ini adalah dapat membantu baik dari sisi penjual yang dapat memberikan saran laptop yang cocok untuk pembeli secara efisien dan cepat, dan dari sisi pembeli yang dapat membantu mengambil keputusan memilih laptop sesuai dengan kriteria yang sesuai dengannya
3	Bagus Nur Ihwa, (2020)	Sistem Pendukung Keputusan Pemilihan Jaksa Terbaik dengan Menerapkan Metode MABAC (Studi Kasus: Kejaksaan Negeri Medan)	Kriteria-kriteria yang diambil dalam penilaian ialah segi orientasi, segi pelayanan, segi integritas, segi komitmen, segi disiplin, dan segi kerja sama memerlukan sistem pendukung keputusan (SPK) untuk mempercepat dan mempermudah dalam membuat suatu keputusan. Sistem pendukung keputusan digunakan untuk membantu pengambilan keputusan dalam sebuah organisasi untuk mempermudah dalam pengambilan sebuah keputusan. Di mana tidak seorang pun mengetahui secara pasti bagaimana seharusnya keputusan di buat. Salah satu metode yang digunakan pada sistem pendukung keputusan ini adalah MABAC. Sistem pendukung keputusan untuk membantu pemerintah daerah untuk pemilihan jaksa terbaik yang sesuai dengan syarat dan kriteria-kriteria yang ditentukan. Metode MABAC merupakan salah satu dari sekumpulan metode sistem pendukung keputusan yang termasuk dalam kelompok metode multi criteria decision making (MABAC)
4	Rivalri Kristianto Hondro (2018)	MABAC: Pemilihan Penerima Bantuan Rastra Menggunakan Metode MultiAttributive Border Approximation Area Comparison	Mendukung keputusan pemilihan penerima bantuan rastra yang sifatnya multi alternative maka diperlukan metode pendukung keputusan. Metode MABAC adalah salah satu metode pendukung keputusan yang jenis keputusannya multi alternative, metode ini dapat membantu proses pengambilan keputusan pemilihan bantuan rastra dengan optimal. MABAC merupakan metode perbandingan multikriteria. Metode ini dipilih karena, metode ini menyediakan stabil

			(konsisten) solusi dan handal untuk pengambilan keputusan rasional, dibandingkan dengan metode lain multi-kriteria pengambilan keputusan (SAW, COPRAS, Moora, TOPSIS dan VI-KOR).
--	--	--	---

II.2. Sistem

Sistem dapat terdiri dari sistem-sistem bagian (*subsystem*). Sebagai misal, sistem komputer dapat terdiri dari subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar, dan kemudian subsistem-subsistem tersebut akan berinteraksi sedemikian rupa sehingga dapat mencapai satu kesatuan yang terpadu. Dalam buku Analisa dan Design Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis, sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu (Han & goleman, daniel; boyatzis, Richard; Mckee, 2019)

II.3. Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah sistem informasi berbasis komputer yang interaktif, dengan cara mengolah data dengan berbagai model untuk memecahkan masalah-masalah yang tidak terstruktur sehingga dapat memberikan informasi yang bisa digunakan oleh para pengambil keputusan dalam membuat sebuah keputusan. Dalam sebuah sistem pendukung keputusan,

sumber daya intelektual yang dimiliki seseorang dipadukan dengan kemampuan computer untuk membantu meningkatkan kualitas dari keputusan yang diambil. Pengambilan keputusan merupakan sebuah proses memilih sebuah tindakan diantara beberapa alternatif yang ada, sehingga tujuan yang diinginkan dapat tercapai. (Alfina & Harahap, 2019)

Sistem pendukung keputusan (SPK) merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur. Tujuan dari SPK adalah untuk membantu pengambil keputusan memilih berbagai alternatif keputusan yang merupakan pengolahan informasi-informasi yang diperoleh atau tersedia dengan menggunakan model pengambilan keputusan. Ciri utama sekaligus keunggulan dari sistem pendukung keputusan tersebut adalah kemampuannya untuk menyelesaikan masalah-masalah yang tidak terstruktur. Pengambilan keputusan merupakan proses pemilihan alternatif tindakan untuk mencapai tujuan atau sasaran tertentu. Pengambilan keputusan dilakukan dengan pendekatan sistematis terhadap permasalahan melalui proses pengumpulan data menjadi informasi serta ditambah dengan faktor-faktor yang perlu dipertimbangkan dalam pengambilan keputusan. Tahap-tahap yang harus dilalui dalam proses pengambilan keputusan sebagai berikut :

- a. Tahap Pemahaman (*Intelligence Phase*) Tahap ini merupakan proses penelusuran dan pendeteksian dari lingkup problematika serta proses

pengenalan masalah. Data masukan diperoleh, diproses, dan diuji dalam rangkamenidentifikasi masalah.

- b. Tahap Perancangan (*Design Phase*) Tahap ini merupakan proses pengembangan dan pencarian alternatif tindakan atau solusi yang dapat diambil. Tersebut merupakan representasi kejadian nyata yang disederhanakan, sehingga diperlukan proses validasi dan verifikasi untuk mengetahui keakuratan model dalam meneliti masalah yang ada.
- c. Tahap Pemilihan (*Choice Phase*) Pada tahap ini dilakukan pemilihan terhadap berbagai alternatif solusi yang dimunculkan pada tahap perencanaan agar ditentukan atau dengan memperhatikan kriteria-kriteria berdasarkan tujuan yang akan dicapai. Tahap Implementasi (*Implementation Phase*) Pada tahap ini dilakukan penerapan terhadap rancangan sistem yang telah dibuat pada tahap perancangan serta pelaksanaan alternatif tindakan yang telah dipilih pada tahap pemilihan. (Alfina & Harahap, 2019)

Konsep Sistem Pendukung Keputusan (SPK) pertama kali diungkapkan pada tahun 1971 oleh Michael Scoot Morton dengan istilah Management Decision System. Kemudian sejumlah perusahaan, lembaga penelitian dan perguruan tinggi miulai melakuakn penelitian dan membangun Sistem Pendukung Keputusan, sehingga dari produksi yang dihasilkan dapat disimpulkan bahwa sistem ini merupakan suatu sistem berbasis komputer hang ditujukan untuk membantu pengambilan keputusan dalam memanfaatkan data dan model tertentu untuk memecahkanberbagai persoalan yang tidak terstruktur. (Rinaldy Manurung : 2020)

II.2.4. *Simple Additive Weighting (SAW)*

Menurut Harold Situmorang (2015) salah satu metode yang digunakan untuk menyelesaikan masalah dari Fuzzy Multiple Attribute Decision Making (FMADM) adalah metode *Simple Additive Weighting (SAW)* yaitu suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. *Definisi Metode Simple Additive Weighting (SAW)* sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut.

Metode SAW membutuhkan proses normalisasi matriks keputusan X ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Hasil terbesar adalah alternatif pilihan terbaik dimana berupa informasi tentang pilihan yang paling tepat. Metode ini cocok dengan sistem yang nantinya yang akan dibuat dimana menentukan suatu pilihan terbaik dari beberapa bobot yang telah disediakan oleh sistem.

Menurut Tomy Reza Adianto, dkk (2017) ada beberapa kelebihan metode *Simple Additive Weighting (SAW)* antara lain :

- a. Menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif terbaik dari sejumlah alternatif.
- b. Penilaian akan lebih tepat karena didasarkan pada nilai kriteria dari bobot preferensi yang sudah ditentukan.

- c. Adanya perhitungan normalisasi matriks sesuai dengan nilai atribut (antara nilai benefit dan cost) (Harold Situmorang ; 2015)

II.2.5. Tahapan Metode SAW

Langkah-langkah Penggunaan Metode SAW adalah sebagai berikut :

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan. Dalam tahap ini penulis berusaha menentukan beberapa alternatif dan kriteria-kriteria yang dibutuhkan untuk pembuatan Sistem Informasi yang akan dilakukan untuk memecahkan masalah yang ingin dipecahkan, dimana masing-masing kriteria nantinya akan diberikan nilai bobot untuk masing-masingnya.
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria. Setelah mencari alternatif dan kriteria-kriteria yang dibutuhkan, penulis mencoba mencocokkan dengan masing-masing rating dari alternative dan kriteria, dimana
3. Membuat matriks keputusan berdasarkan kriteria, kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan ataupun atribut biaya) sehingga diperoleh matriks ternormalisasi R.
4. Hasil akhir diperoleh dari proses perangkingan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik sebagai solusi.

Formula untuk melakukan normalisasi tersebut adalah sebagai berikut :

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

Dimana :

R_{ij} = Nilai rating kinerja normalisasi

X_{ij} = Nilai atribut yang dimiliki dari setiap kriteria

Max_{ij} = Nilai terbesar dari setiap baris dan kriteria

Min_{ij} = Nilai terkecil dari setiap kriteria

Benefit = Nilai terbesar adalah yang terbaik

Cost = Nilai terkecil adalah yang terbaik

Dengan r_{ij} adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j ;

$i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, n$. Nilai preferensi untuk setiap alternatif (V_i)

diberikan sebagai :

$$V_i = \sum_{j=1}^n w_j r_{ij} \dots\dots\dots(2)$$

Dimana:

V_i = Rangking dari setiap alternatif

W_i = Nilai bobot yang telah ditentukan

R_{ij} = Nilai rating ternormalisasi

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih. (Harold Situmorang ; 2015)

Studi kasus :

Adapun langkah-langkah yang dilakukan dalam penerapan metode SAW dalam sistem pendukung keputusan pemilihan buah durian legit adalah sebagai berikut :

1. Menentukan preferensi kriteria dan alternatif dari setiap kriteria dalam pengambilan keputusan :

Tabel II.2. Preferensi Kriteria dan Alternatif

Kriteria	Alternatif				
Warna Kulit <i>Benefit (B)</i>	Kuning Kecoklatan	Kuning Muda	Kuning Muda	Hijau Kekuningan	Hijau
Jarak Duri <i>Benefit (B)</i>	Sangat Jarang	Jarang	Cukup Jarang	Kurang Jarang	Tidak Jarang
Panjang Duri <i>Cost (C)</i>	Sangat Pendek	Pendek	Cukup Pendek	Kurang Pendek	Tidak Pendek
Besar Tangkai <i>Benefit (B)</i>	Sangat Besar	Besar	Cukup Besar	Kurang Besar	Tidak Besar
Panjang Tangkai <i>Cost (C)</i>	Sangat Pendek	Pendek	Cukup Pendek	Kurang Pendek	Tidak Pendek
Suara Ketukan <i>Benefit (B)</i>	Sangat Nyaring	Nyaring	Cukup Nyaring	Kurang Nyaring	Tidak Nyaring
Aroma Buah <i>Benefit (B)</i>	Sangat Harum	Harum	Cukup Harum	Kurang Harum	Tidak Harum
Bobot	100	80	60	40	20

2. Memasukkan data dari tiap kriteria pada tiap alternatif buah durian.

Tabel II.3. Data Penilaian Durian

Kode	Nama	C1	C2	C3	C4	C5	C6	C6
D001	Durian 1	Hijau	Tidak Jarang	Tidak Pendek	Kurang Besar	Tidak Pendek	Tidak Nyaring	Tidak Harum
D002	Durian 2	Hijau	Sangat Jarang	Pendek	Cukup Besar	Sangat Pendek	Kurang Nyaring	Cukup Harum
D003	Durian 3	Hijau Kekuningan	Tidak Jarang	Tidak Pendek	Kurang Besar	Tidak Pendek	Cukup Nyaring	Tidak Harum
D004	Durian 4	Kuning	Cukup Jarang	Sangat Pendek	Besar	Kurang Pendek	Sangat Nyaring	Sangat Harum

D005	Durian 5	Kuning Muda	Kurang Jarang	Cukup Pendek	Tidak Besar	Pendek	Tidak Nyaring	Cukup Harum
D006	Durian 6	Kuning	Tidak Jarang	Sangat Pendek	Sangat Besar	Tidak Pendek	Nyaring	Harum
D007	Durian 7	Hijau Kekuningan	Jarang	Tidak Pendek	Cukup Besar	Sangat Pendek	Cukup Nyaring	Kurang Harum

3. Membuat matriks keputusan

Tabel II.4 Konversi Data Nilai Durian

Kode	Nama	C1	C2	C3	C4	C5	C6	C7
D001	Durian 1	20	20	20	40	20	20	20
D002	Durian 2	20	100	80	60	100	40	60
D003	Durian 3	40	20	20	40	20	60	20
D004	Durian 4	80	60	100	80	40	100	100
D005	Durian 5	60	40	60	20	80	20	60
D006	Durian 6	80	20	100	100	20	80	80
D007	Durian 7	40	80	20	60	100	60	40

4. Normalisasi matriks keputusan

$$r_{1,1} = \frac{20}{\max(20, 20, 40, 80, 60, 80, 40)} = 0,25$$

$$r_{1,2} = \frac{20}{\max(20, 100, 20, 60, 40, 20, 80)} = 0,20$$

$$r_{1,3} = \frac{\min(20, 80, 20, 100, 60, 100, 20)}{20} = 1,00$$

$$r_{1,4} = \frac{40}{\max(40, 60, 40, 80, 20, 100, 60)} = 0,40$$

$$r_{1,5} = \frac{\min(20, 100, 20, 40, 80, 20, 100)}{20} = 1,00$$

$$r_{1,6} = \frac{20}{\max(20, 40, 60, 100, 20, 80, 60)} = 0,20$$

$$r_{1,7} = \frac{20}{\max(20, 60, 20, 100, 60, 80, 40)} = 0,20$$

$$r_{7,7} = \frac{80}{\max(20,60,20,100,60,80,40)} = 0,80$$

Tabel II.5 Normalisasi Matriks Keputusan

Kode	Nama	C1	C2	C3	C4	C5	C6	C7
D001	Durian 1	0,250	0,200	1,000	0,400	1,000	0,200	0,200
D002	Durian 2	0,250	1,000	0,250	0,600	0,200	0,400	0,600
D003	Durian 3	0,500	0,200	1,000	0,400	1,000	0,600	0,200
D004	Durian 4	1,000	0,600	0,200	0,800	0,500	1,000	1,000
D005	Durian 5	0,750	0,400	0,333	0,200	0,250	0,200	0,600
D006	Durian 6	1,000	0,200	0,200	1,000	1,000	0,800	0,800
D007	Durian 7	0,500	0,800	1,000	0,600	0,200	0,600	0,400

5. Menetapkan Bobot Kriteria

Tabel II.6 Bobot Standar Penilaian

Kriteria	C1	C2	C3	C4	C5	C6	C7
Bobot Kriteria (%)	20%	15%	5%	10%	5%	15%	30%

6. Kalikan matriks keputusan dengan bobot kriteria

$$V_1 = (0,25 \times 0,2) + (0,2 \times 0,15) + (1 \times 0,05) + (0,4 \times 0,1) + (1 \times 0,05) + 0,2 \times 0,15 + (0,2 \times 0,3)$$

$$= (0,05) + (0,03) + (0,05) + (0,04) + (0,05) + (0,03) + (0,06)$$

$$= 0,310$$

$$V_2 = (0,25 \times 0,2) + (1,00 \times 0,15) + (0,25 \times 0,05) + (0,60 \times 0,1) + (1 \times 0,20) + (0,40 \times 0,15) + (0,60 \times 0,3)$$

$$= (0,05) + (0,15) + (0,013) + (0,06) + (0,01) + (0,08) + (0,18)$$

$$= 0,523$$

$$V_3 = (0,25 \times 0,2) + (0,2 \times 0,15) + (1 \times 0,05) + (0,4 \times 0,1) + (1 \times 0,05) + (0,2 \times 0,15) + (0,2 \times 0,3)$$

$$= 0,420$$

$$V_4 = (0,25 \times 0,2) + (0,2 \times 0,15) + (1 \times 0,05) + (0,4 \times 0,1) + (1 \times 0,05) + (0,2 \times 0,15) + (0,2 \times 0,3)$$

$$= 0,855$$

$$V_5 = (0,25 \times 0,2) + (0,2 \times 0,15) + (1 \times 0,05) + (0,4 \times 0,1) + (1 \times 0,05) + (0,2 \times 0,15) + (0,2 \times 0,3)$$

$$= 0,469$$

$$V_6 = (0,25 \times 0,2) + (0,2 \times 0,15) + (1 \times 0,05) + (0,4 \times 0,1) + (1 \times 0,05) + (0,2 \times 0,15) + (0,2 \times 0,3)$$

$$= 0,750$$

$$V_7 = (0,50 \times 0,2) + (0,80 \times 0,15) + (1 \times 0,05) + (0,60 \times 0,1) + (0,20 \times 0,05) + (0,60 \times 0,15) + (0,40 \times 0,3)$$

$$= 0,550$$

Tabel II.7 Pencarian Nilai Total V

Kode	Nama	Total (V)	Rangking
V[D001]	Durian 1	0,310	7
V[D002]	Durian 2	0,523	4
V[D003]	Durian 3	0,420	6
V[D004]	Durian 4	0,855	1
V[D005]	Durian 5	0,469	5
V[D006]	Durian 6	0,750	2
V[D007]	Durian 7	0,550	3

7. Preferensi tiap alternatif

Nilai terbesar ada pada V_3 , sehingga **D004** adalah alternatif yang terpilih sebagai alternatif terbaik. Dengan kata lain, **Durian4** adalah durian yang terpilih sebagai durian paling legit.

II.5. Metode *Mabac*

Mabac (*Multi-Attributive Border Approximation area Comparison*) merupakan salah satu metode sistem pendukung keputusan yang bersifat multikriteria dan dianggap sebagai salah satu metode yang handal dalam pengambilan keputusan rasional. Langkah penyelesaian metode *Multi-Attributive Border Approximation Area Comparison* (MABAC) sebagai berikut:

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan dan penentuan bobot.
2. Membentuk tabel alternatif beserta kriteria masing – masing alternatif.
3. Membentuk matriks keputusan awal (X) berdasarkan rating
4. Normalisasi elemen matriks awal (X)

$$tij = \frac{xij - xi^+}{xi^- - xi^+}, \text{ jika jenis kriteria merupakan Biaya/Cost}$$

$$tij = \frac{xij - xi^-}{xi^+ - xi^-} \text{ jika jenis kriteria merupakan keuntungan/Benefit}$$

5. Perhitungan matriks tertimbang (V)

$$Vij = (wi * tij) + wi,$$

6. Penentuan matriks area perkiraan perbatasan (G)

$$G_i = [\prod_{j=1}^m V_{ij}]^{1/m}$$

7. Perhitungan elemen matriks jarak alternatif dari daerah perkiraan perbatasan (Q)

$$Q_{ij} = V_{ij} - G_i$$

8. Perankingan alternatif. Perankingan alternatif dilakukan dengan cara menambahkan setiap elemen kriteria dari masing – masing alternatif berdasarkan matriks jarak alternatif dari daerah perkirann perbatasan (Q).

(Hengky Yosafat : 2020)

Contoh kasus :

Tabel II.1. Bobot Kriteria

Kriteria	Bobot
Jiwa kepemimpinan (c1)	0.178
Integritas (c2)	0.284
Visioner (c3)	0.207
0.207	0.100
Disiplin (c5)	0.057
Bidang hukum (c6)	0.064
Keberanian (c7)	0.044
Independen (c8)	0.066

Sumber : (Hengky Yosafat : 2020)

Sistem penentu pemilihan jaksa terbaik di negeri medan merupakan sistem yang sederhana atau manual, dimana bagian pemilihan memberikan laporan berdasarkan pada setiap kriteria-kriteria kepala jaksa serta program pengambilan keputusan. Alternatif-alternatif yang akan diberikan dalam pemilihan jaksa terbaik antara lain seperti pada tabel di bawah ini :

Kode	Nama Alternatif	C1	C2	C3	C4	C5	C6	C7	C8
A1	Bahtiar, SH	1.10	3.12	3.89	4.2	2.21	1.03	3.00	5.00
A2	Kasim Pohan	3.05	3.98	2.96	3.02	4.10	2.99	1.10	4.03
A3	Nur Ainun, SH	1.90	4.95	3.01	2.90	4.95	4.06	5.00	1.10
A4	Rocky Sirait, SH	2.85	3.87	3.12	1.05	2.93	4.89	3.30	4.90
A5	Uock Yontha	4.77	3.00	4.87	3.01	1.97	3.99	2.04	4.00

Karena semua kriteria memiliki karakter deskriptif (linguistik), nilai-nilai kriteria didefinisikan melalui fuzzy deskriptor linguistik, gambar dibawah ini: Membentuk matriks keputusan awal (X) (Forming initial decision matrix (X)) pada langkah pertama dilakukan evaluasi alternatif “m” dengan “n” kriteria. Adapun matriks keputusan awal diperoleh dengan defuzzifikasi fuzzy linguistic deskriptor oleh masing-masing kriteria :

	C1	C2	C3	C4	C5	C6	C7	C8
A1	1.10	3.12	3.89	4.20	2.21	1.03	3.00	5.00
A2	3.05	3.98	2.96	3.02	4.10	2.99	1.10	4.03
A3	1.90	4.95	3.01	2.90	4.95	4.06	5.00	1.10
A4	2.85	3.87	3.12	1.05	2.93	4.89	3.30	4.90
A5	4.77	3.00	4.87	3.01	1.97	3.99	2.04	4.00

Hasil nilai normalisasi elemen matriks. Karena semua kriteria untuk membuat dinormalisasi Matriks (N) :

	C1	C2	C3	C4	C5	C6	C7	C8
A1	0.00	0.06	0.49	1.00	0.08	0.00	0.49	1.00
A2	0.53	0.50	0.00	0.63	0.71	0.51	0.00	0.75
A3	0.22	1.00	0.03	0.59	1.00	0.78	1.00	0.00
A4	0.48	0.45	0.08	0.00	0.34	1.00	0.55	0.97
A5	1.00	0.00	1.00	0.62	0.00	0.77	0.24	0.74

Menentukan matriks dari daerah perkiraan perbatasan (G) diambil dari nilai himpunan fuzzy dengan kode G yaitu 0.5 diambil dari nilai di belakang koma.

$$\begin{aligned} G_i &= (0.18 \cdot 0.27 \cdot 0.22 \cdot 0.26 \cdot 0.36)^{1/5} \\ &= 0.25 \end{aligned}$$

$$\begin{aligned} G_i &= (0.30 \cdot 0.43 \cdot 0.57 \cdot 0.41 \cdot 0.28)^{1/5} \\ &= 0.39 \end{aligned}$$

$$\begin{aligned} G_i &= (0.31 \cdot 0.21 \cdot 0.21 \cdot 0.22 \cdot 0.41)^{1/5} \\ &= 0.26 \end{aligned}$$

$$\begin{aligned} G_i &= (0.20 \cdot 0.16 \cdot 0.16 \cdot 0.10 \cdot 0.16)^{1/5} \\ &= 0.15 \end{aligned}$$

$$\begin{aligned} G_i &= (0.06 \cdot 0.10 \cdot 0.11 \cdot 0.08 \cdot 0.06)^{1/5} \\ &= 0.08 \end{aligned}$$

$$\begin{aligned} G_i &= (0.06 \cdot 0.10 \cdot 0.11 \cdot 0.13 \cdot 0.11)^{1/5} \\ &= 0.10 \end{aligned}$$

$$G_i = (0.07 \cdot 0.04 \cdot 0.09 \cdot 0.07 \cdot 0.05)^{1/5}$$

$$= 0.06$$

$$G_i = (0.13 \cdot 0.12 \cdot 0.07 \cdot 0.13 \cdot 0.12)^{1/5}$$

$$= 0.11$$

Perhitungan matriks jarak elemen alternatif dari batas perkiraan daerah

(Q) :

$$Q = V - G$$

$$= 0.18 - 0.25$$

$$= -0.07$$

$$Q = V - G$$

$$= 0.27 - 0.25$$

$$= 0.02$$

$$Q = V - G$$

$$= 0.22 - 0.25$$

$$= -0.03$$

$$Q = V - G$$

$$= 0.26 - 0.25$$

$$= 0.01$$

$$Q = V - G$$

$$= 0.36 - 0.25$$

$$= 0.11$$

Langkah terakhir adalah perengkingan alternatif diambil nilai dari kode

M dengan himpunan fuzzy 0.8 diambil dari nilai dibelakang koma.

$$S1 = \sum q_{ij} \cdot 8^{j-1} = -0.07 - 0.08 + 0.04 + 0.05 - 0.02 - 0.04 + 0.000 + 0.02$$

$$= -0.1$$

$$S1 = \sum q_{ij} 8 j-1 = 0.02 + 0.04 - 0.06 + 0.01 + 0.02 + 0.000 - 0.02 + 0.01 \\ = -0.02$$

$$S1 = \sum q_{ij} 8 j-1 = -0.03 + 0.18 - 0.05 + 0.01 + 0.04 + 0.01 - 0.03 - 0.04 \\ = 0.18$$

$$S1 = \sum q_{ij} 8 j-1 = 0.01 + 0.03 - 0.04 - 0.05 + 0.0000 + 0.03 + 0.01 + 0.02 \\ = 0.01$$

$$S1 = \sum q_{ij} 8 j-1 = 0.11 + 0.03 - 0.04 - 0.05 + 0.000 + 0.03 + 0.01 + 0.01 \\ = 0.1$$

Tabel II.2 Bobot Kriteria

Nama Alternatif	Nilai	Rank	Keterangan
Bahtiar, SH	-0.1	4	Buruk
Kasim Pohan, SH	-0.02	5	Luar Biasa
Nur Ainun, SH	0.18	1	Sangat Baik
Rocky Sirait, SH	0.01	3	Sangat Buruk
Uock Yontha, SH	0.1	2	Sedang

Hasil dari Tabel 3 menunjukkan bahwa alternatif A3 mendapatkan ranking pertama dan sedangkan alternatif A2 sebagai yang terakhir. Jadi pemilihan jaksa terbaik adalah alternatif A3 karena memiliki perangkungan pertama didapatkan berdasarkan penilaian dari kriteria masing-masing.

II.6. PHP (*Hypertext Preprocessor*)

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa

PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, *Java*, dan *Perl* serta mudah untuk dipelajari. PHP merupakan bahasa *scripting server – side*, dimana pemrosesan datanya dilakukan pada sisi *server*. Sederhananya, *server* yang akan menterjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan. Adapun pengertian lain PHP adalah *akronim* dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode – kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML”. (Muhammad Faisal Widad : 2017).

PHP (singkatan rekursif: *PHP Hypertext Preprocessor*) merupakan script yang dapat disisipkan ke dalam HTML. PHP dipakai untuk membuat sistem berbasis web yang dinamis. PHP bahasa pemrograman yang *server-side* sehingga program tersebut akan dijalankan atau diproses oleh server . Hingga saat ini PHP sudah rilis versi 7.4.1 pada 18 desember 2019. (Roberto kaban,. Dkk : 2019)

II.7. MySQL (*My Structure Query Language*)

MySQL (*My Structure Query Language*) merupakan sebuah program pembuat *database* yang bersifat *Open Source*, artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua *platform* baik *Windows* maupun *linux*. MySQL juga merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi multi *user*. MySQL juga sering dikenal dengan nama sistem manajemen *database*

relasional. Suatu *database* relasional menyimpan data dalam tabel yang terpisah. Tabel -tabel tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan memperoleh kombinasi data dari beberapa tabel dalam suatu permintaan. Untuk administrasi *database*, seperti pembuatan *database*, pembuatan tabel, dan sebagainya dapat digunakan aplikasi berbasis web seperti *PHP MyAdmin* dengan aplikasi *XAMPP*. (Muhammad Faisal Widad : 2017)

II.8. Database

Database adalah basis dari data, dengan kata lain database merupakan kumpulan data, dasar yang digunakan untuk menampilkan data atau informasi, sekumpulan data atau informasi teratur berdasarkan criteria tertentu yang saling berhubungan. Selain itu juga database dapat di definisikan sebagai susunan *record* data operasional lengkap dari suatu organisasi perusahaan, yang terorganisir dan disimpan secara terintegrasi. (Herlina Trisnawati, 2016)

Data adalah penggambaran suatu fakta atau keadaan. Informasi merupakan sekumpulan data yang telah diolah menjadi suatu bentuk sehingga berguna bagi penerima dan pemakai. Dalam dunia komputer *database* bisa dikategorikan bisa sangat spesial karena selalu menjadi hal utama dalam perancangan sistem suatu perusahaan, tentunya ada alasan tertentu mengapa *database* menjadi prioritas sendiri dalam kinerja manajemen perusahaan. (Herlina Trisnawati, 2016)

II.9. Normalisasi

Normalisasi adalah proses pembentukan struktur basis data sehingga sebagian besar ambiguity bisa dihilangkan. Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data relasional yang mengelompokkan atribut dari suatu tabel sehingga membentuk struktur tabel yang normal. Adapun kriteria tabel dikatakan normal adalah ketika tidak ada kerangkapan data (redudansi data). (Puspitasari et al., 2016)

Tujuan dari normalisasi adalah untuk :

1. Untuk menghilangkan kerangkapan data sehingga meminimumkan pemakaian *storage* yang dipakai oleh *base relations (file)*.
2. Untuk mengurangi kompleksitas.
3. Untuk mempermudah pemodifikasian data.

II.9.1. Proses Normalisasi

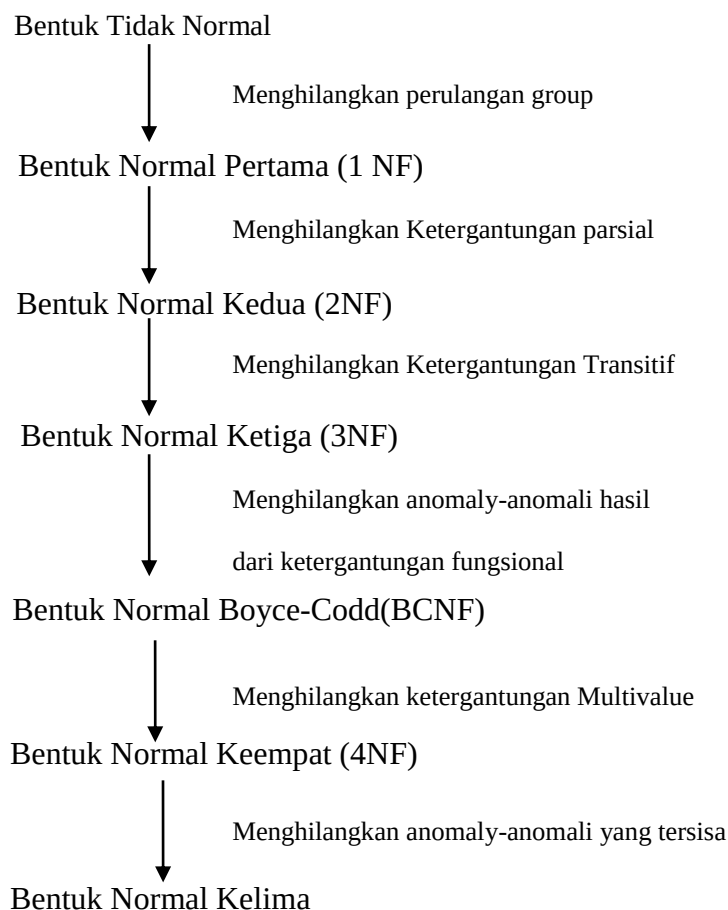
Gambaran proses normalisasi yaitu:

1. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat, kemudian.
2. Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal. Untuk melakukan proses tersebut dibutuhkan beberapa tahapan.

Tahapan dalam normalisasi dimulai dari tahap paling ringan (*1NF*) hingga paling ketat (*5NF*). Biasanya hanya sampai pada tingkat *3NF* atau *BCNF* karena

sudah cukup memadai untuk menghasilkan tabel-tabel yang berkualitas baik.

Urutan tahapan normalisasi tampak seperti gambar II.1.



Gambar II.1. Tahapan pada Normalisasi
(Sumber : Dwi Puspitasari, dkk ; 2016)

Adapun aturan dalam normalisasi adalah suatu tabel dikatakan baik (efisien) atau normal jika memenuhi 3 kriteria sbb:

1. Jika ada dekomposisi (penguraian) tabel, maka dekomposisinya harus dijamin aman (*Lossless-Join Decomposition*). Artinya, setelah tabel tersebut diuraikan / didekomposisi menjadi tabel-tabel baru, tabel-tabel baru tersebut bisa menghasilkan tabel semula dengan sama persis.

2. Terpeliharanya ketergantungan fungsional pada saat perubahan data (*Dependency Preservation*).
3. Tidak melanggar *Boyce-Codd Normal Form (BCNF)*.

Jika kriteria ketiga (*BCNF*) tidak dapat terpenuhi, maka paling tidak tabel tersebut tidak melanggar Bentuk Normal tahap ketiga (*3rd Normal Form / 3NF*). Pada penelitian ini formula yang dibuat sampai memenuhi bentuk normal ke 3 yaitu *3NF*.

Proses Normalisasi Berdasarkan tahapan normalisasi, terdapat enam bentuk normal yaitu Elmasri & Navathe (2013):

1. Bentuk Normal Tahap Pertama (*1st Normal Form / 1NF*).
2. Bentuk Normal Tahap Kedua (*2nd Normal Form / 2NF*).
3. Bentuk Normal Tahap (*3rd Normal Form / 3NF*).
4. *Boyce-Code Normal Form (BCNF)*.
5. Bentuk Normal Tahap (*4th Normal Form / 4NF*) 6. Bentuk Normal Tahap (*5th Normal Form / 5NF*)

Bentuk Normal Pertama / *1NF*, memiliki aturan:

- a. Tidak adanya atribut *multi-value*, atribut komposit atau kombinasinya.
- b. Mendefinisikan atribut kunci.
- c. Setiap atribut dalam tabel tersebut harus bernilai atomic (tidak dapat dibagi-bagi lagi).

Bentuk Normal Kedua / *2NF*, memiliki aturan :

- a. Sudah memenuhi dalam bentuk normal kesatu (*1NF*).

- b. Semua atribut bukan kunci hanya boleh tergantung (*functional dependency*) pada atribut kunci. Jika ada ketergantungan parsial maka atribut tersebut harus dipisah pada tabel yang lain.
- c. Perlu ada tabel penghubung ataupun kehadiran foreign key bagi atribut-atribut yang telah dipisah tadi.

Bentuk Normal Ketiga / *3NF*, memiliki aturan:

- a. Sudah memenuhi dalam bentuk normal kedua (*2NF*)
- b. Tidak ada ketergantungan transitif (dimana atribut bukan kunci tergantung pada atribut bukan kunci lainnya). (Puspitasari et al., 2016)

II.10. UML (*Unified Modelling Language*)

UML (Unified Modelling Language) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena *UML* menyediakan Bahasa pemodelan visual yang memungkinkan bagi pengembangan sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*Sharing*) dan mengkomunikasikan rancangan dengan baik. (Helmi Kurniawan : 2015)

Unified Modelling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek. UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek

karena UML menyediakan Bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku. UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi.

Unified Modeling Language (UML) biasa digunakan untuk :

- a. Menggambarkan batasan sistem dan fungsi - fungsi sistem secara umum, dibuat dengan *use case* dan *actor*.
- b. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*.
- c. Menggambarkan representasi struktur *static* sebuah sistem dalam bentuk *class diagrams*.
- d. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan *state transition diagrams*.
- e. Menyatakan arsitektur implementasi fisik menggunakan *component and development*.
- f. Menyampaikan atau memperluas *functionality* dengan *stereotypes*.

(Alfina & Harahap, 2019)

UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar

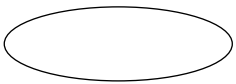
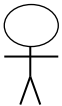
bahasa pemodelan umum dalam industri perangkat dan pengembangan sistem.
(Urva et al., 2015)

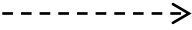
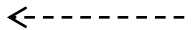
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah Empat alat bantu sebagai berikut: *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram* dan *Class Diagram*.

1. *Use Case Diagram*

Use Case Diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. Use Case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan Use Case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Symbol-simbol yang digunakan dalam Use Case Diagram Yaitu

Tabel II.3. Simbol Use Case

Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.

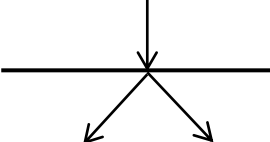
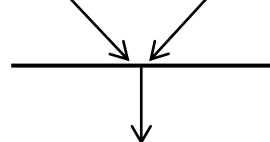
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

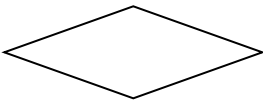
(Sumber : Gellysa Urva, dkk ; 2015)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol- simbol yang digunakan dalam *activity diagram* yaitu:

Tabel II.4. Simbol *Activity Diagram*

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.

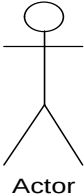
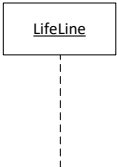
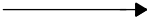
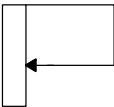
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

(Sumber : Gellysa Urva,dkk ; 2015)

3.Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antara objek. Simbol-simbol yang digunakan dalam *Sequence Diagram* yaitu:

Tabel II.5. Sequence Diagram

Gambar	Keterangan
	Orang, Proses atau sistem lain yang berinteraksi dengan sistem informasi.
	Merepresentasikan entitas tunggal dalam sequence diagram, digambarkan dengan kotak. Entitas ini memiliki nama, <i>Stereotype</i> atau berupa <i>instance</i> .
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.

	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .
---	---

(Sumber : Gellysa Urva, dkk ; 2015)

4. Class Diagram (Diagram Kelas)

Merupakan hubungan antara kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang merupakan perilaku sistem. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class Diagram* secara khas meliputi: Kelas (*Class*), *Relasi*, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/ Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

Tabel II.6. Multiplicity Class Diagram

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Gellysa Urva, dkk ; 2015)