

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Secara umum, fotografi adalah suatu seni menangkap cahaya menggunakan kamera, yang dilakukan melalui sensor digital atau film, dan kemudian membentuk gambar yang disebut foto. Dengan peralatan seperti kamera, seseorang dapat menangkap panjang gelombang cahaya yang tidak dapat dilihat oleh mata manusia, termasuk gelombang ultraviolet, inframerah, dan radio.

Kemajuan teknologi yang pesat membuat produsen kamera bersaing dalam produksi kamera dengan teknologi terbaru, yaitu kamera *mirrorless*. Dengan spek kamera yang mirip dengan DSLR bahkan bisa lebih baik, namun dengan wujud yang simpel dan praktis, konsumen dapat membawanya kemana saja tanpa harus meletakkan kamera di tempat yang luas. Kemajuan teknologi menyebabkan munculnya tren fotografi dengan menggunakan kamera digital, yang memudahkan kalangan mulai dari pemula hingga yang menyukai fotografi hingga yang lebih serius menekuni fotografi.

Kamera *mirrorless* hadir dalam banyak merek, tipe, dan spesifikasi. Beragamnya kamera *mirrorless* membuat calon konsumen kesulitan untuk menentukan pilihan yang tepat, apalagi dengan munculnya produk kamera *mirrorless* yang baru dengan teknologi yang semakin canggih. Hal ini dapat menjadi masalah bagi calon konsumen yang tidak cukup paham mengenai kamera *mirrorless*. Beberapa kriteria yang sering dijadikan rujukan dalam memilih kamera *mirrorless* adalah ISO maksimal, resolusi video, *megapixel*, *shutter speed* maksimal, jumlah titik fokus, daya tahan baterai, dan harga dimana masing-masing kamera memiliki spesifikasi yang berbeda. Selain itu merek juga menjadi bahan pertimbangan konsumen dalam memilih kamera yang juga dapat mempengaruhi hasil pengambilan foto nantinya.

Penelitian dengan menggunakan metode *Weighted Product* dan *Simple Additive Weighting* telah banyak dilakukan diantaranya penelitian oleh Dina Lorenza dengan judul Sistem Pendukung Keputusan Pemilihan *Driver* Terbaik

Menggunakan Metode *Weighted Product* (WP) tersebut membahas tentang bagaimana memilih *driver* terbaik dengan mempertimbangkan beberapa kriteria, yaitu kedisiplinan, kerjasama tim, perilaku, *skill*, ritase perbulan, dan masa kerja[1]. Penelitian oleh Fitri Yanti dengan judul Sistem Pendukung Keputusan Pemilihan *Customer* Terbaik Menggunakan Metode WP tersebut membahas bagaimana memilih *customer* terbaik dengan mempertimbangkan beberapa kriteria, yaitu penghasilan, intensitas belanja, total belanja, dan jarak rumah[2]. Sedangkan penelitian oleh Rudi Aryanto dengan judul Sistem Pendukung Keputusan Pemilihan Sepeda Motor Dengan Metode *Simple Additive Weighting* Berbasis Web tersebut membahas bagaimana memilih sepeda motor yang terbaik dengan mempertimbangkan beberapa kriteria, yaitu kapasitas silinder, kapasitas tangki, berat motor, dan harga[3]. Dan juga penelitian oleh Renny Puspita Sari dengan judul Sistem Pendukung Keputusan Pemilihan Bahan Bakar Sepeda Motor *Matic* Menggunakan Metode *Simple Additive Weighting* (SAW) tersebut membahas bagaimana memilih bahan bakar yang paling layak untuk sepeda motor *matic* dengan mempertimbangkan beberapa kriteria, yaitu harga, keiritan, kelangkaan, kualitas, dan antrian[4]. Kelebihan metode SAW adalah dapat menentukan nilai bobot setiap atribut, kemudian dilanjutkan dengan proses perengkingan yang akan menyeleksi alternatif terbaik dari sejumlah alternatif dan penilaian akan lebih tepat karena didasarkan pada nilai kriteria dan bobot preferensi yang sudah ditentukan[5]. Sedangkan kelebihan metode WP adalah dimana rating setiap atribut harus dipangkatkan dulu dengan bobot yang bersangkutan. Proses ini sama halnya dengan proses normalisasi. Sehingga dapat menentukan atau mengambil keputusan dari permasalahan yang ada dengan perhitungan yang akurat. Dari penelitian yang telah diutarakan tersebut, dapat disimpulkan bahwa sistem pendukung keputusan dengan menggunakan metode *weighted product* yang sudah diteliti tersebut dapat membantu dalam mencari solusi untuk pengambilan keputusan. Sehingga berdasarkan latar belakang tersebut pada penelitian ini dirancang sebuah sistem pendukung keputusan pemilihan kamera *mirrorless* menggunakan metode *weighted product* dan *simple additive weighting* guna mengatasi permasalahan bagi calon konsumen dalam mengambil keputusan untuk memilih kamera *mirrorless*

sesuai keinginan atau kebutuhannya agar lebih tepat dan meminimalisir kerugian yang didapat oleh calon konsumen.

1.2 Perumusan Masalah

1. Bagaimana membangun sistem pendukung keputusan pemilihan kamera *mirrorless* untuk mendapatkan rekomendasi dalam pemilihan atau pembelian kamera *mirrorless* sesuai dengan kriteria menggunakan metode *weighted product* dan *simple additive weighting* ?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah untuk mengimplementasikan metode *weighted product* dan *simple additive weighting* pada sistem pendukung keputusan pemilihan kamera *mirrorless* sesuai dengan kriteria yang ditentukan atau diinginkan.

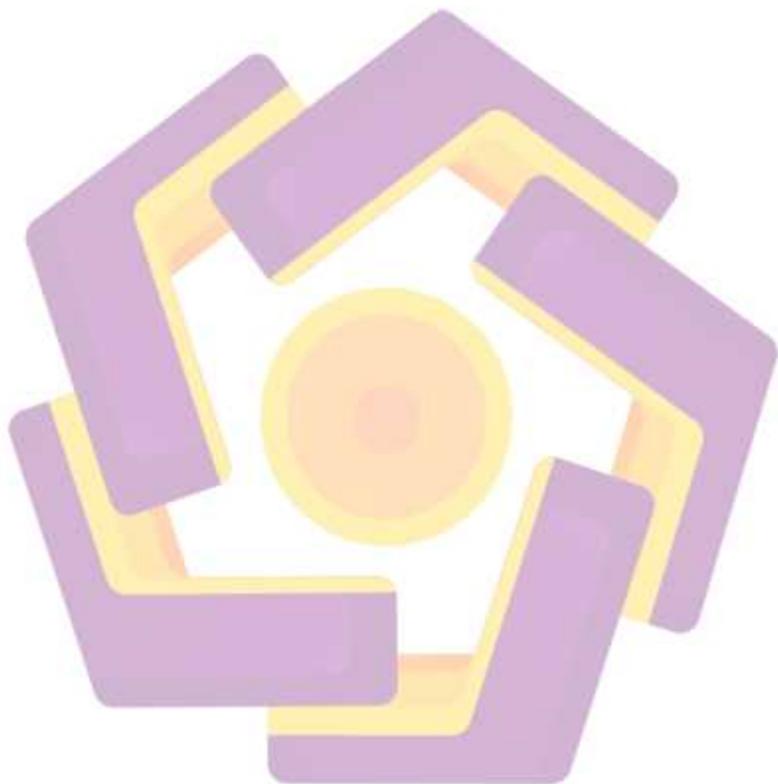
1.4 Batasan Masalah

- a. Data kamera *mirrorless* yang digunakan adalah data dari 3 merek, yaitu Canon, Sony, dan Fujifilm dari tahun produksi 2015 hingga 2021.
- b. Metode yang digunakan dalam sistem pendukung keputusan ini adalah metode *weighted product* dan *simple additive weighting*.
- c. Kriteria yang digunakan adalah ISO maksimal, *shutter speed* maksimal, resolusi video, *megapixel*, jumlah titik fokus, daya tahan baterai, dan harga.
- d. Ditujukan untuk fotografer pemula yang belum memiliki pengetahuan yang cukup dalam memilih kamera *mirrorless*.

1.5 Manfaat Penelitian

Penelitian ini dapat menambah kekayaan pengetahuan umum tentang sistem pendukung keputusan pemilihan kamera *mirrorless* dengan metode *weighted product* dan *simple additive weighting*, serta dapat juga dijadikan referensi kepustakaan untuk menjadi dasar penelitian selanjutnya di bidang yang sama di masa mendatang dan diharapkan dapat memberikan masukan atau bantuan dalam mempelajari lebih lanjut tentang sistem pendukung keputusan pemilihan kamera

mirrorless yang juga dapat digunakan sebagai dasar untuk merumuskan kebijakan perusahaan kedepannya.



BAB II

TINJAUAN PUSTAKA

2.1 *Literature Review*

Melakukan penelitian memerlukan penemuan dasar dalam penelitian yang telah dilakukan orang lain sebelumnya. Beberapa penelitian yang telah dilakukan oleh orang lain dapat dijadikan sebagai referensi dan pembandingan untuk penelitian selanjutnya.

Diantaranya penelitian dengan judul “Sistem Pemilihan *Smartphone* Terbaik Dengan Menggunakan Perhitungan Metode *Weighted Product*” oleh Linda Yuniarti (2022). Pada aplikasi yang dibangun dalam penelitian tersebut menggunakan lima kriteria yaitu CPU, RAM, baterai, kamera, dan harga. Berdasarkan hasil penelitian ini, dapat diambil kesimpulan bahwa metode *weighted product* adalah metode yang simpel dalam melakukan persamaan. Dengan menggunakan penerapan metode *weighted product*, maka dapat ditentukan sebuah pemilihan rekomendasi *smartphone* yang terbaik dan bagus untuk semua kebutuhan[6].

M. Syahril (2021) melakukan penelitian yang berjudul “Sistem Pendukung Keputusan Pemilihan Laptop Untuk Kebutuhan Kuliah Metode *Simple Additive Weighting* (SAW)”. Sistem ini dibuat untuk membantu para calon konsumen yang mengalami kebingungan dalam menentukan agar lebih mudah mempertimbangkan dalam memilih laptop yang sesuai dengan keinginan. Sistem ini menggunakan enam kriteria sebagai acuan pengguna untuk mendapatkan rekomendasi, yaitu: harga, merek prosesor, *harddisk*, RAM, VGA, dan ukuran layar. Hasilnya sistem ini dapat memberikan keputusan terbaik dalam pemilihan laptop sesuai dengan keinginan calon konsumen. Berdasarkan hasil uji coba pemeringkatan dari 6 alternatif, didapatkan alternatif terbaik yang dijadikan rekomendasi ke calon konsumen yaitu laptop ASUS Vivobook A412FA dan berdasarkan 26 data yang telah diajukan, diperoleh sebanyak 22 laptop (84,61%) yang sesuai dan 4 laptop (15,38%) yang tidak sesuai[7].

Diski Ijtima Putri (2021) dengan penelitian berjudul “Sistem Pendukung Keputusan Untuk Pemilihan *Smart* Televisi Menggunakan *Simple Additive*

Weighting”, membangun sebuah sistem pendukung keputusan untuk membantu menyelesaikan masalah pengguna dalam memilih *smart tv* yang sesuai dengan kebutuhannya. Sistem ini menggunakan empat kriteria sebagai acuan pengguna untuk mendapatkan rekomendasi, yaitu: harga, ukuran layar, resolusi layar, konsumsi daya, dan garansi produk. Hasil pengujian validitas perhitungan SAW menyatakan bahwa hasil kesimpulan dari 10 responden menggunakan pengujian pengguna dengan skala likert adalah presentase sebesar 78,8% sistem memiliki manfaat dan 84% sistem memiliki kemudahan untuk dipakai[8].

Muazir (2021) dengan penelitian berjudul “*Weighted Product* dalam Sistem Rekomendasi Pemilihan Karyawan Berbasis Web”, membangun sebuah sistem rekomendasi pemilihan karyawan untuk menyelesaikan masalah pengguna yang kesulitan untuk menyeleksi karyawan yang layak diterima sesuai dengan kebutuhannya berdasarkan perankingan metode *weighted product*. Sistem ini menggunakan enam kriteria sebagai acuan pengguna untuk mendapatkan rekomendasi, yaitu: keahlian, kemampuan berkomunikasi, pengalaman kerja, usia, tes tertulis, dan wawancara. Sistem rekomendasi yang dibuat dapat berjalan sesuai dengan tujuan untuk membantu menyeleksi karyawan yang layak diterima berdasarkan bobot kriteria dari pengguna[9].

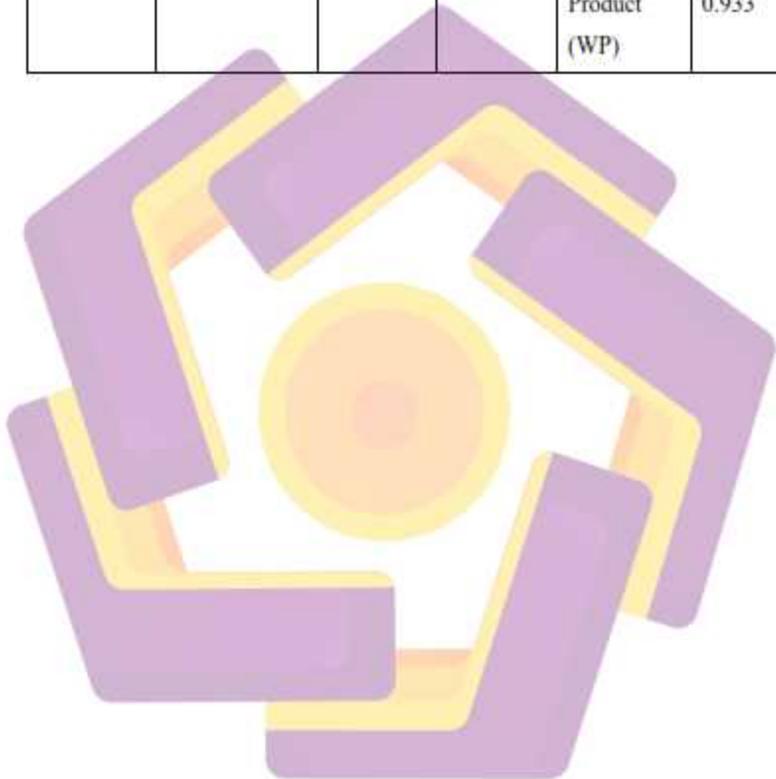
Sungguh Zebua (2021) melakukan penelitian dengan judul “Sistem Pendukung Keputusan Pemilihan Kain Terbaik Di Toko Crown Textile & Tailor Dengan Metode *Simple Additive Weighting* (SAW)”. Penelitian ini membangun sistem berbasis *website* untuk merekomendasikan kain terbaik untuk digunakan sesuai dengan kebutuhan penggunanya. Terdapat 4 kriteria yang digunakan sebagai acuan pengguna untuk mendapatkan rekomendasi dalam sistem ini, yaitu: jenis bahan, kualitas, motif, dan harga. Berdasarkan kesimpulan penelitian ini penerapan metode SAW berhasil diterapkan sehingga dapat menghasilkan rekomendasi kain terbaik yang sesuai dengan kebutuhan penggunanya[10].

Tabel 2.1 Tabel Perbandingan

KOMPO NEN	PENULIS DAN KETERANGAN				
	Linda Yuniarti, Putri Nur Amallianti, Moch Haidar Rafi, Risky Adi Surya. Perani Rosyani 2022	Muhammad Syahril, Imam Suharjo 2021	Diski Ijtima Putri, Mutaqin Akbar 2021	Muazir, Novia Hasdyna, Rahmat 2021	Sungguh Zebua, R. Mahdalena Simanjorang 2021
Basis Aplikasi	-	-	Website	Website	Website
Hal/Barang Topik	<i>Smartphone</i>	Laptop	<i>Smart TV</i>	Karyawan	Kain
Kriteria yang digunakan	CPU, RAM, Baterai, Kamera, Harga	Harga, Merek, Prosesor, <i>Harddisk</i> , RAM, VGA, Ukuran Layar	Harga, Ukuran Layar, Resolusi Layar, Konsumsi Daya, Garansi Produk	Keahlian, Kemampuan Berkomunikasi, Pengalaman Kerja, Usia, Tes Tertulis, Wawancara	Jenis Bahan, Kualitas, Motif, Harga

<p>Hasil Penelitian</p>	<p>Sistem yang dibangun dapat berjalan dan dari hasil perancangan, H2 atau Huawei Mate 30 Pro 5G adalah rekomendasi dan pilihan terbaik dalam semua penggunaan</p>	<p>Sistem ini dapat memberikan keputusan terbaik dalam pemilihan laptop sesuai dengan keinginan pengguna atau calon konsumen . Berdasarkan 26 data yang telah diajukan, diperoleh sebanyak 22 laptop (84,61%) yang sesuai dan 4 laptop (15,38%) yang tidak sesuai</p>	<p>Hasil pengujian validitas perhitungan SAW menyatakan bahwa hasil kesimpulan dari 10 responden menggunakan pengujian dengan skala likert adalah presentase sebesar 78,8% sistem memiliki manfaat dan 84% sistem memiliki kemudahan untuk dipakai</p>	<p>Sistem rekomendasi yang dibuat dapat berjalan dengan baik dan dari hasil perhitungan nilai rating tertinggi di raih oleh alternative ke – 8 yang bernama Nasir, sehingga A8 dapat disimpulkan masuk dalam kualifikasi penerimaan karyawan di Toko Sejahtera Swalayan Lhokseuma dengan</p>	<p>Penerapan metode SAW berhasil diterapkan sehingga dapat menghasilkan rekomendasi kain terbaik yang sesuai dengan kebutuhan pengguna. Hasil pengolahan data kriteria dan alternatif menggunakan SAW diperoleh alternatif kain terbaik adalah A1</p>
--------------------------------	--	---	--	--	---

				perhitungan menggunakan metode Weighted Product (WP)	yaitu kain viscose rayon dengan nilai 0.933
--	--	--	--	--	---



<p>Perbedaan</p>	<p>Sistem yang dibangun bertujuan untuk dapat membantu dalam pengambilan sebuah keputusan dengan memecahkan masalah dalam memilih <i>smartphone</i> mana yang terbaik yang sesuai dengan kebutuhan penggunaan yang akan dibutuhkan dengan mempertimbangkan kriteria-kriteria yang ada</p>	<p>Sistem yang dibangun memiliki tujuan untuk merancang serta membangun sebuah sistem pendukung keputusan pemilihan pada laptop yang fokusnya hanya untuk perkuliahan dan menggunakan metode <i>Simple Additive Weighting</i> (SAW)</p>	<p>Sistem yang dibangun dalam penggunaan Sistem Pendukung Keputusan (SPK) dapat membantu keputusan yang diambil dalam memilih <i>smart tv</i> yang tepat dan sesuai kriteria</p>	<p>Sistem yang dibangun bertujuan untuk membantu menyeleksi karyawan yang layak diterima berdasarkan bobot kriteria dari pengguna.</p>	<p>Sistem yang dibangun memiliki tujuan untuk dapat menentukan pemilihan kain terbaik menggunakan SPK dengan metode <i>Simple Additive Weighting</i> (SAW) dalam pemilihan kain terbaik di Crown Textile & Tailor</p>
-------------------------	---	---	--	--	---

2.2 Landasan Teori

2.2.1 Kamera

2.2.1.1 Pengertian Kamera

Kamera adalah sebuah perangkat yang digunakan untuk mengabadikan momen, atau memvisualisasikan keadaan sekitar menggunakan sebuah sensor dalam bentuk format digital dan disimpan dalam media penyimpanan (memori). Kamera adalah alat yang paling populer dan paling utama yang dibutuhkan dalam aktivitas fotografi maupun videografi (Rizky, 2020)[11].

2.2.1.2 Kamera Mirrorless

Sesuai dengan namanya, kamera ini memang tidak memiliki cermin layaknya DSLR. Karena hal itu, kamera ini jadi mempunyai ukuran yang lebih kecil dan ringkas jika dibandingkan dengan DSLR. Jika kamera ringkas seperti pocket tidak bisa diganti lensanya, kamera *mirrorless* ini bisa dilepas, sama halnya dengan DSLR. Sehingga bisa menyesuaikan dengan situasi dan jenis fotografi yang akan digunakan (Rizky, 2020)[11].

2.2.2 Konsep Dasar Sistem Pendukung Keputusan

2.2.2.1 Definisi Sistem Pendukung Keputusan

Sistem Pendukung Keputusan atau *Decision Support System* (DSS) secara umum didefinisikan sebagai sebuah sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengomunikasian untuk masalah semi terstruktur (Turban, 2005)[12].

2.2.2.2 Karakteristik Sistem Pendukung Keputusan

Dari definisi sistem pendukung keputusan, menurut Turban (2005)[12] terdapat sejumlah karakteristik dan kemampuan SPK, antara lain :

1. Dukungan untuk pengambil keputusan, terutama pada situasi semi terstruktur dan tidak terstruktur.
2. Dukungan untuk semua tingkat manajerial.
3. Dukungan untuk individu dan kelompok.
4. Dukungan untuk semua keputusan independen dan atau sekuensial.
5. Dukungan di semua fase proses pengambilan keputusan: inteligensi, desain, pilihan, dan implementasi.

6. Dukungan pada berbagai proses dan gaya pengambilan keputusan.
7. Kemampuan sistem beradaptasi dengan cepat dimana pengambil keputusan dapat menghadapi masalah-masalah baru dan pada saat yang sama dapat teratasi dengan cara mengadaptasikan sistem terhadap kondisi perubahan yang terjadi.
8. Pengguna merasa seperti di rumah. *User friendly*, kapabilitas grafis yang kuat, dan sebuah bahasa interaktif yang alami.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, *timeliness*, kualitas) dari pada efisiensi (biaya).
10. Pengambil keputusan mengontrol penuh semua langkah proses pengambilan keputusan dalam memecahkan masalah.
11. Pengguna akhir dapat mengembangkan dan memodifikasi sistem sederhana.
12. Menggunakan model-model dalam menganalisis situasi pengambilan keputusan.
13. Disediakkannya akses untuk berbagai sumber data, format, dan tipe, mulai dari sistem informasi geografi (GIS) hingga sistem berorientasi objek.
14. Dapat dilakukan sebagai alat yang dapat berdiri sendiri yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di beberapa organisasi.

2.2.2.3 Komponen Sistem Pendukung Keputusan

Di bawah ini adalah komponen sistem pendukung keputusan menurut Turban (20015)[12] :

1. **Data Management.** Termasuk *database*, yang mengandung data yang relevan untuk berbagai situasi dan diatur oleh perangkat lunak yang disebut *Database Management Systems (DBMS)*.
2. **Model Management.** Melibatkan model finansial, statistik, *management science*, atau berbagai model kuantitatif lainnya, sehingga dapat memberikan ke sistem suatu kemampuan analitis, dan manajemen perangkat lunak yang diperlukan.

3. **Communication** (*dialogue subsystem*). Pengguna dapat berkomunikasi dan memberikan perintah pada SPK melalui subsistem ini. Ini berarti menyediakan antarmuka.
4. **Knowledge Management**. Subsistem opsional ini dapat mendukung subsistem lain atau bertindak sebagai komponen yang berdiri sendiri.

2.2.2.4 Kelebihan Sistem Pendukung Keputusan

Di bawah ini adalah kelebihan sistem pendukung keputusan menurut Kosasi (2002)[13] :

1. SPK dapat menunjang pembuatan keputusan manajemen dalam menangani masalah semi terstruktur dan tidak terstruktur.
2. SPK dapat membantu manajer pada berbagai tingkatan manajemen, mulai dari manajemen tingkat atas sampai manajemen tingkat bawah.
3. SPK memiliki kemampuan permodelan dan analisis pembuatan keputusan.
4. SPK dapat menunjang pembuatan keputusan yang saling berurutan baik secara kelompok maupun perorangan.
5. SPK menunjang berbagai bentuk proses pengambilan keputusan dan jenis keputusan.
6. SPK dapat melakukan adaptasi setiap saat dan bersifat fleksibel.
7. SPK mudah berinteraksi dengan sistem dan mudah dikembangkan oleh pengguna (*user*) terakhir.
8. SPK dapat meningkatkan efektivitas dalam pengambilan keputusan.
9. SPK mudah melakukan pengaksesan berbagai sumber dan format data.

2.2.3 Simple Additive Weighting (SAW)

Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua kriteria (Kusumadewi, 2006)[5]. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Metode SAW mengenal adanya 2 (dua) atribut yaitu kriteria keuntungan (*benefit*) dan kriteria biaya (*cost*). Perbedaan mendasar dari kedua kriteria ini adalah dalam pemilihan kriteria ketika mengambil keputusan.

2.2.3.1 Langkah-Langkah Metode Simple Additive Weighting (SAW)

Langkah-langkah dalam menggunakan metode ini (Eniyati, S. 2011)[5] adalah :

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu C_i .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria (C_i), 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 (A_i) sebagai solusi.

Persamaan untuk melakukan normalisasi tersebut adalah :

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_{ij}} & \text{jika } j \text{ adalah atribut keuntungan} \\ \frac{\text{Min}_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya} \end{cases} \quad (2-1)$$

Keterangan :

r_{ij} = rating kinerja ternormalisasi

Max_{ij} = nilai maksimum dari setiap baris dan kolom

Min_{ij} = nilai minimum dari setiap baris dan kolom

x_{ij} = baris dan kolom dari matriks

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 persamaan :

$$V_i = \sum_{j=1}^n W_j r_{ij} \quad (2-2)$$

Keterangan :

V_i = nilai akhir dari alternatif

W_j = bobot yang telah ditentukan

r_{ij} = normalisasi matriks

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih.

Dengan ketentuan :

- a. Dikatakan atribut keuntungan apabila atribut banyak memberikan keuntungan bagi pengambil keputusan, sedangkan atribut biaya merupakan atribut yang banyak memberikan pengeluaran jika nilainya semakin besar bagi pengambil keputusan.
- b. Apabila berupa atribut keuntungan maka nilai (X_{ij}) dari setiap kolom atribut dibagi dengan nilai (Max_{ij}) dari tiap kolom, sedangkan untuk atribut biaya, nilai (Min_{ij}) dari tiap kolom atribut dibagi dengan nilai (X_{ij}) setiap kolom.

2.2.4 Weighted Product (WP)

Menurut Putra Jaya (2012)[14], menjelaskan : “Metode *Weighted Product* merupakan metode pengambilan keputusan dengan cara perkalian untuk menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan dulu dengan bobot atribut yang bersangkutan.”

Berdasarkan pernyataan diatas dapat disimpulkan bahwa metode *weighted product* memerlukan proses normalisasi karena metode ini mengalihkan hasil penilaian setiap atribut dan tujuan dari metode ini adalah dapat menentukan atau mengambil keputusan dari permasalahan yang ada dengan perhitungan yang akurat.

2.2.4.1 Langkah-Langkah Metode Weighted Product (WP)

Langkah-langkah dalam menggunakan metode ini adalah :

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan.
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Menentukan bobot preferensi tiap kriteria.
4. Mengalikan seluruh atribut bagi sebuah alternatif dengan bobot berpangkat positif untuk atribut keuntungan dan bobot berpangkat negatif untuk atribut biaya.

Metode ini menggunakan perkalian sebagai perkalian sebagai untung menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan

dulu dengan bobot yang bersangkutan. Proses ini sama halnya dengan proses normalisasi. Preferensi untuk alternatif S_i diberikan sebagai berikut :

1. Penentuan nilai bobot W

$$W_j = \frac{w_j}{\sum w_j} \dots \dots \dots (1) \quad (2-3)$$

2. Penentuan nilai Vektor S

$$S = (W_{1j}^{AW_j}, w), (W_{in}^{AW_n}, w) \dots \dots \dots (2) \quad (2-4)$$

3. Penentuan nilai Vektor V

$$V_{jn} = \frac{s_j}{\sum s_j} \dots \dots \dots (3) \quad (2-5)$$

Keterangan :

- V = preferensi alternatif dianalogikan sebagai vektor V
- W = bobot kriteria / sub kriteria
- j = kriteria
- i = alternatif
- n = banyaknya kriteria
- S = preferensi alternatif dianalogikan sebagai vektor S

Langkah-langkah selanjutnya dalam perhitungan metode *weighted product* adalah sebagai berikut :

1. Mengalihkan seluruh atribut bagi seluruh alternatif dengan bobot sebagai pangkat positif bagi atribut keuntungan dan bobot sebagai pangkat negatif bagi atribut biaya.
2. Hasil perkalian dijumlahkan untuk menghasilkan nilai pada setiap alternatif.
3. Membagi nilai V bagi setiap alternatif dengan nilai pada setiap alternatif.
4. Ditemukan urutan alternatif terbaik yang akan menjadi keputusan.

2.2.5 Analisis Sistem

2.2.5.1 Analisis PIECES

Menurut Hanif Al Fatta (2007)[15], untuk mengidentifikasi masalah, harus dilakukan analisis terhadap kinerja, informasi, ekonomi, keamanan aplikasi,

efisiensi, dan pelayanan pelanggan. Panduan ini dikenal dengan analisis PIECES (*performance, information, economy, control, efficiency, dan services.*) Dari analisis ini biasanya didapatkan beberapa masalah utama. Hal ini penting karena biasanya yang muncul di permukaan bukan masalah utama, tetapi hanya gejala dari masalah utama saja.

Secara umum berikut penjelasan mengenai analisis PIECES :

1. Analisis Kinerja (*Performance*)

Adalah kemampuan dalam menyelesaikan tugas bisnis dengan cepat sehingga sasaran dapat segera tercapai. Kinerja diukur dengan jumlah produksi (*throughput*) dan waktu tanggap (*response time*) dari suatu sistem. Sistem yang dikembangkan ini akan menyediakan jumlah produksi dan waktu tanggap yang memadai untuk kebutuhan manajemen.

2. Analisis Informasi (*Information*)

Laporan-laporan yang sudah selesai diproses digunakan untuk menghasilkan informasi yang dibutuhkan oleh manajemen didalam pengambil keputusan. Informasi merupakan hal yang tidak kalah penting karena dengan informasi tersebut pihak manajemen akan merencanakan langkah-langkah selanjutnya.

3. Analisis Ekonomi (*Economy*)

Adalah penilaian sistem dalam pengurangan dan keuntungan yang akan didapatkan dari sistem yang dikembangkan. Sistem ini akan memberikan penghematan operasional dan meningkatkan keuntungan perusahaan. Penghematan didapat melalui pengurangan bahan baku dan perawatan. Sementara keuntungan didapat dari peningkatan nilai informasi dan keputusan yang dihasilkan.

4. Analisis Keamanan (*Control*)

Sistem keamanan yang digunakan harus dapat mengamankan data dari kerusakan, misalnya dengan membuat *back up* data. Selain itu sistem keamanan juga harus dapat mengamankan data dari akses yang tidak diijinkan, biasanya dilakukan dengan password terutama pada form aplikasi dan databasenya.

5. Analisis Efisiensi (*Efficiency*)

Berhubung dengan sumber daya yang ada guna meminimalkan pemborosan. Efisiensi dari sistem yang dikembangkan adalah pemakaian secara maksimal atas sumber daya yang tersedia yang meliputi manusia, informasi, waktu, uang, peralatan, ruang, dan keterlambatan pengolahan data.

6. Analisis Layanan (*Service*)

Perkembangan organisasi dipicu peningkatan pelayanan yang lebih baik. Peningkatan pelayanan terhadap sistem yang dikembangkan akan memberikan kehandalan terhadap konsistensi dalam pengolahan input dan outputnya serta kehandalan dalam menangani pengecualian dan kemampuan menangani masalah yang diluar kondisi norma serta mampu mengkoordinasi aktifitas untuk mencapai tujuan dan sasaran.

2.2.5.2 Analisis Kebutuhan Sistem

Menurut Hanif Al Fatta (2007)[15], analisis terhadap kebutuhan sistem bertujuan untuk memahami apa yang dibutuhkan oleh sistem baru dan mengembangkan sebuah sistem untuk memenuhi kebutuhan sistem tersebut, atau memutuskan bahwa pengembangan suatu sistem baru sebenarnya tidak perlu dilakukan. Kebutuhan sistem terbagi menjadi kebutuhan fungsional, nonfungsional dan bisnis *rule*.

2.2.6 Perancangan Sistem

2.2.6.1 Basis Data

2.2.6.1.1 Pengertian Basis Data

Menurut Muhammad Fikry (2019)[16], basis data adalah kumpulan dari item data yang saling berhubungan satu dengan lainnya yang diorganisasikan berdasar sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan *software* digunakan untuk melakukan manipulasi data (diperbaharui, dicari, diolah dengan perhitungan- perhitungan tertentu, dan dihapus) dengan tujuan tertentu.

2.2.6.1.2 Tujuan Basis Data

Menurut Muhammad Fikry (2019)[16], di bawah ini adalah tujuan basis data :

1. Mengatur data/mengorganisasikan data agar diperoleh kemudahan, ketepatan, dan kecepatan dalam pengambilan kembali.
2. Tidak ada duplikasi data sehingga konsistensi data mudah dijaga.
3. Data terintegrasi.
4. Informasi selalu mutakhir (*up to date*).
5. Data dapat dipakai secara bersama oleh beberapa pemakai.

2.2.6.1.3 Manfaat Basis Data

Menurut Muhammad Fikry (2019)[16], di bawah ini adalah manfaat basis data[12]:

1. Kecepatan dan Kemudahan (*Speed*)
Memungkinkan kita untuk dapat menyimpan dan melakukan perubahan/manipulasi terhadap data atau menampilkan kembali data tersebut dengan lebih cepat dan mudah.
2. Kebersamaan Pemakaian (*Shareability*)
Pemakai basis data tidak terbatas, pengisian data dapat dilakukan oleh beberapa orang dalam satu lokasi.
3. Pemusatan Kontrol Data (*Control*)
Data yang ada menjadi terpusat pada satu tempat penyimpanan. Sehingga kita dapat mengaksesnya kapan saja.
4. Efisiensi Ruang Penyimpanan (*Space*)
Tidak adanya redundansi data sehingga efisiensi/optimalisasi penggunaan ruang penyimpanan dapat dilakukan. Penekanan jumlah redundansi data, dilakukan dengan menerapkan sejumlah pengkodean atau membuat relasi antar kelompok data yang saling berhubungan.
5. Keakuratan (*Accuracy*)
Pemanfaatan pengkodean dengan batasan tertentu, yang membuat satu data menjadi unik dan berbeda dengan yang lain, sehingga ketika menyimpan data tidak akan ada data yang sama dalam penyimpanan.
6. Ketersediaan (*Availability*)
Karena kepentingan pemakaian data, sebuah basis data dapat memiliki data yang disebar di banyak lokasi. Dengan pemanfaatan teknologi jaringan

komputer, data nasabah yang berada di suatu cabang sebuah bank dapat diakses (menjadi tersedia/*availability*) di cabang lainnya.

7. Kelengkapan (*Completeness*)

Data yang di input ke dalam sebuah basis data memiliki ruang yang besar sehingga data dapat dimasukkan dalam jumlah yang banyak sesuai dengan kebutuhan pengguna.

8. Keamanan (*Security*)

Adanya *password* setiap pemakai basis data. Kita juga dapat menentukan siapa saja yang boleh mengakses data penting atau data biasa.

9. Kemudahan dalam Pembuatan Program Aplikasi Baru

Data yang disimpan dalam di ekspor ke program aplikasi lain dengan menjamin terjaga/terpeliharanya data.

10. *User View*

Pemakai dapat melihat langsung bentuk tampilan penginputan data, sehingga memudahkan pemakai dalam mengelola data.

2.2.6.2 Entity Relationship Diagram (ERD)

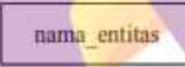
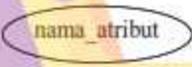
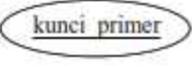
Untuk menggambarkan struktur data agar bisa dipahami secara langsung oleh pengguna teknik maupun teknik dapat menggunakan *entity relationship diagram* (ERD). *Entity relationship diagram* (ERD) menjelaskan setiap entitas dan hubungan yang terjadi antar entitas tersebut.

Menurut Rosa dan Shalahuddin (2015)[17] "*Entity Relationship Diagram* (ERD) merupakan pemodelan awal basis data yang sering digunakan. ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika". Sedangkan menurut Lubis (2016)[18] "ERD menjadi salah satu pemodelan data konseptual yang paling sering digunakan dalam proses pengembangan basis data bertipe relasional".

Maka dari itu, penulis menyimpulkan bahwa *entity relationship diagram* (ERD) merupakan teknik pemodelan struktur data secara konseptual yang menggambarkan entitas lengkap dengan atributnya dan hubungan yang terjadi antar entitas tersebut.

Penjelasan dari simbol-simbol yang digunakan untuk menggambarkan *entity relationship diagram* (ERD) menurut Chen dalam buku Rosa dan Shalauddin (2015)[17] disajikan ke dalam bentuk tabel.

Tabel 2.2 Komponen *Entity Relationship Diagram* (ERD)

Notasi	Komponen	Keterangan
	Entitas / Entity	Entitas merupakan data inti yang akan disimpan, bakal tabel pada basis data, benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer. Penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
	Atribut	Field atau kolom data yang butuh disimpan dalam suatu entitas.
	Atribut kunci primer	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan, biasanya berupa id. Kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa

Notasi	Komponen	Keterangan
		kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
	Atribut multivalai / <i>multivalue</i>	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki lebih dari satu.
	Relasi	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja.
	Asosiasi / <i>association</i>	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i>

Notasi	Komponen	Keterangan
		menghubungkan entitas A dan entitas B.

Sumber: (Rosa dan Salahuddin, 2015)

2.2.7 Flowchart

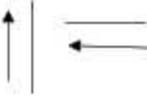
Menurut Kristanto (2003)[19], *Flowchart* adalah suatu fungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Biasanya *flowchart* ini sangat berguna untuk program yang sangat panjang, karena urutan – urutannya dapat dilanjutkan dengan jelas dalam *flowchart* yang dibuat.

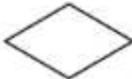
Flowchart (diagram alir) adalah bagan yang menunjukkan alir di dalam program atau prosedur sistem secara logika. *Flowchart* merupakan metode untuk menggambarkan tahap-tahap pemecahan masalah dengan mempresentasikan simbol-simbol tertentu yang mudah dimengerti, mudah digunakan dan standar. Tujuan penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi, dan jelas dengan menggunakan simbol-simbol yang standar. Tahapan penyelesaian masalah yang disajikan harus jelas, sederhana, dan tepat (Jogiyanto, 2005)[20].

Adapun simbol-simbol yang digunakan dalam *flowchart* adalah sebagai berikut :

Tabel 2.3 Simbol-Simbol *Flowchart*

Simbol	Nama	Keterangan
	Dokumen	Simbol ini digunakan untuk menggambarkan semua jenis dokumen, yang merupakan formulir yang digunakan untuk merekam terjadinya suatu transaksi.

Simbol	Nama	Keterangan
	<p><i>Input Output</i></p>	<p>Simbol ini digunakan untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.</p>
	<p>Penghubung pada halaman yang sama</p>	<p>Untuk memungkinkan aliran dokumen berhenti di suatu halaman tertentu.</p>
	<p>Penghubung halaman berbeda</p>	<p>Untuk menunjukkan kemana dan bagaimana bagan alir terkait atau dengan yang lainnya. Nomor yang tercantum di dalam simbol penghubung menunjukkan bagaimana bagan alir yang tercantum pada simbol dihalaman yang lain.</p>
	<p>Proses</p>	<p>Mengambarkan pengolahan data dengan komputer.</p>
	<p>Garis alir</p>	<p>Mengambarkan arah proses pengolahan data. Anak panah tidak digambarkan jika arus dokumen mengarah kebawah dan kekanan.</p>

Simbol	Nama	Keterangan
	Mulai/Akhir	Untuk menggambarkan awal dan akhir suatu sistem.
	Kondisi	Simbol keputusan yang menunjukkan suatu kondisi.
	Arsip sementara	Arsip sementara adalah dokumen yang disimpan dan akan diambil kembali. Untuk menunjukan urutan pengarsipan dokumen digunakan simbol sebagai berikut : A=menurut abjad N=menurut nomor urut T=menurut tanggal
	Kegiatan manual	Untuk menggambarkan suatu kegiatan yang dilakukan manual.
	Arsip permanen	Menggambarkan arsip permanen yang tidak akan diproses lagi dalam sistem yang bersangkutan.

Sumber: (Jogiyanto, 2005)

2.2.8 Unified Modeling Language (UML)

Pada perkembangan teknik pemrograman berorientasi objek, muncul sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek yaitu *Unified*

Modeling Language (UML). UML merupakan bahasa visual yang menjadi standar untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak (Sugiarti, 2013)[21].

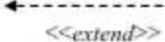
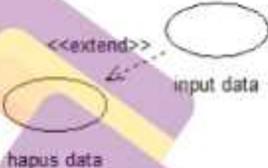
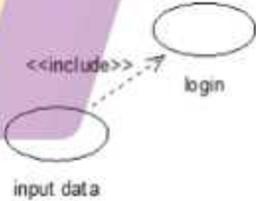
2.2.8.1 Use Case Diagram

Use case atau diagram *use case* menggambarkan kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Sugiarti, 2013)[21].

Terdapat beberapa simbol yang digunakan dalam pembuatan *use case diagram* sebagai berikut :

Tabel 2.4 Simbol Use Case Diagram

Simbol	Deskripsi
<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau aktor; sering dinyatakan dengan menggunakan kata kerja, misal input data.</p>
<p>Aktor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dikembangkan di luar sistem tersebut, sehingga meskipun simbol aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; sering dinyatakan menggunakan kata benda, misal admin, mahasiswa, dll.</p>
<p>Asosiasi</p> 	<p>Menunjukkan komunikasi atau hubungan antara aktor dan <i>use case</i> atau antar <i>use case</i>.</p>

Simbol	Deskripsi
<p data-bbox="232 273 303 295"><i>Extend</i></p> 	<p data-bbox="623 273 991 500">Hubungan use case tambahan ke use case dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; anak panah menuju <i>use case</i> yang dituju, contoh :</p> 
<p data-bbox="232 715 303 737"><i>Include</i></p> 	<p data-bbox="623 715 991 1063">Hubungan antara <i>use case</i> tambahan ke sebuah use case dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya, sehingga <i>use case</i> harus dijalankan terlebih dulu sebelum menjalankan <i>use case</i> tambahan; arah panah menunjuk pada <i>use case</i> yang perlu dijalankan terlebih dahulu, contoh :</p> 

Sumber: (Sugiarti, 2013)

2.2.8.2 Activity Diagram

Diagram aktivitas menggambarkan aliran kerja atau aktivitas dari sebuah sistem, tetapi bukan aktivitas aktor. Diagram aktivitas juga menggambarkan

bagaimana alur sistem berawal, pilihan (*decision*) yang mungkin terjadi, dan bagaimana akhir alur sistem tersebut (Sugiarti, 2013)[21].

Tabel 2.5 Simbol Activity Diagram

Simbol	Nama	Keterangan
	<i>Start state</i>	Titik awal pemulaan.
	<i>End state</i>	Titik akhir dari aktivitas.
	<i>Activity</i>	Aktivitas yang dilakukan oleh aktor.
	<i>Decision</i>	Pilihan untuk mengambil keputusan.
	<i>Interaction</i>	Alur.

Sumber: (Sugiarti, 2013)

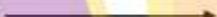
2.2.8.3 Class Diagram

Class diagram digunakan untuk menggambarkan struktur dalam objek sistem. Diagram ini menunjukkan *class object* yang menyusun sistem dan juga hubungan antara *class object* (Sugiarti, 2013)[21].

Kelas memiliki atribut dan metode atau operasi. Atribut adalah variabel-variabel yang mendeskripsikan properti dengan bentuk sebaris teks dalam kelas tersebut, sedangkan metode adalah fungsi yang dimiliki oleh kelas yang dalam *class diagram* dilambangkan menggunakan simbol-simbol. (Sugiarti, 2013)[21].

Tabel 2.6 Simbol Class Diagram

Simbol	Deskripsi
	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih kelas.

<p>Kelas</p> <table border="1" data-bbox="333 270 503 477"> <tr> <td>Nama Kelas</td> </tr> <tr> <td>+ <i>Attribute1</i></td> </tr> <tr> <td>+ <i>Attribute2</i></td> </tr> <tr> <td>+ <i>Operation1()</i></td> </tr> <tr> <td>+ <i>Operation2()</i></td> </tr> </table>	Nama Kelas	+ <i>Attribute1</i>	+ <i>Attribute2</i>	+ <i>Operation1()</i>	+ <i>Operation2()</i>	<p>Kelas pada struktur sistem, tiap kelas memiliki nama, <i>attribute</i>, dan <i>operation</i> atau <i>method</i>.</p>
Nama Kelas						
+ <i>Attribute1</i>						
+ <i>Attribute2</i>						
+ <i>Operation1()</i>						
+ <i>Operation2()</i>						
<p>Antarmuka / <i>interface</i></p> 	<p>Sama seperti konsep <i>interface</i> dalam pemrograman berorientasi objek.</p>					
<p>Asosiasi</p> 	<p>Relasi antar kelas dengan pengertian umum.</p>					
<p>Asosiasi berarah</p> 	<p>Relasi antar kelas dengan pengertian kelas yang satu digunakan oleh kelas yang lain.</p>					
<p>Generalisasi</p> 	<p>Relasi antar kelas dengan pengertian generalisasi-spesialisasi (umum-khusus).</p>					
<p>Kebergantungan</p> 	<p>Relasi antar kelas dengan pengertian kebergantungan antar kelas.</p>					
<p>Agregasi</p> 	<p>Relasi antar kelas dengan makna semua-sebagian (<i>whole-part</i>).</p>					

Sumber: (Sugiarti, 2013)

2.2.8.4 Sequence Diagram

Diagram sekuen merupakan diagram yang digunakan untuk menggambarkan perilaku objek pada *use case* dengan mendeskripsikan proses objek dengan pesan yang dikirimkan. Oleh karena itu, untuk menggambar diagram sekuen harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu (Sugiarti, 2013)[21].

2.2.9 Bahasa Pemrograman Yang Digunakan

2.2.9.1 HTML (Hyper Text Markup Language)

Menurut Nugroho (2006)[22], HTML adalah bahasa pemformatan teks untuk dokumen-dokumen pada jaringan komputer yang sering disebut sebagai *world wide web*. Sedangkan menurut Arief (2011)[23], HTML merupakan salah satu format yang digunakan dalam pembuatan dokumen atau aplikasi yang berjalan di halaman web, dan menurut Suyanto (2007), HTML itu adalah bahasa yang digunakan untuk menulis halaman web, biasanya menggunakan ekstensi .htm, .html atau .shtml.

2.2.9.2 PHP (Hypertext Preprocessor)

PHP (*PHP Hypertext Preprocessor*) adalah kode atau *script* yang akan dieksekusi pada *server side*. *Script* PHP akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML, sehingga suatu halaman web tidak lagi bersifat statis, namun menjadi bersifat dinamis. Sifat *server side* berarti dilakukan di server, baru kemudian hasilnya dikirim ke browser. (Sutaji, 2012)[24].

Ketika seorang pengguna internet akan membuka situs yang menggunakan fasilitas *server-side scripting* PHP, maka terlebih dahulu server yang bersangkutan akan memproses semua perintah PHP di server kemudian mengirimkan hasilnya dalam format HTML ke *web browser* pengguna internet. Dengan demikian, pengguna internet tidak dapat melihat kode program yang ditulis dalam PHP sehingga keamanan dari halaman web menjadi lebih terjamin (Sunarfrihantono, 2002)[25].

2.2.9.3 CSS (Cascading Style Sheets)

Cascading style sheet (CSS) digunakan untuk menampilkan sebuah web dengan tampilan yang menarik, memperindah tampilan web dan mudah digunakan.

Menurut Sakur (2005)[26], *Cascading Style Sheets* merupakan salah satu bagian dari desain *web* yang berfungsi untuk mengatur tampilan setiap unsur yang ada di dalam *website*. Konsep lahirnya CSS adalah untuk memisahkan isi dan tampilan. CSS mengatur tampilan secara langsung dengan menggunakan format html. Penggunaan CSS sangatlah mudah bila dibandingkan dengan mengatur *style* secara langsung pada format html karena bila suatu saat ada perubahan pada salah satu *style* dari sebuah unsur yang ada di *website* maka tinggal mengubah *script* dari

CSS dan secara otomatis seluruh objek atau unsur yang menggunakan *style* dari sebuah CSS akan berubah sesuai aturan baru yang telah dibuat.

2.2.9.4 Javascript

Menurut Wicaksono (2011)[27], *Javascript* adalah bahasa *scripting* yang bekerja disisi *client/browser* sehingga *website* bisa lebih interaktif.

Menurut Agung (2012)[28], *Javascript* adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti Google Chrome dan Mozilla Firefox. Bahasa pemrograman *Client Side* berbeda dengan bahasa pemrograman *Server Side* seperti PHP, dimana untuk *server side* seluruh kode program dijalankan di sisi server. Untuk menjalankan *Javascript*, kita hanya membutuhkan aplikasi text editor dan web browser. *Javascript* memiliki fitur: *high-level programming language, client-side, loosely typed* dan berorientasi objek.

2.2.10 Metode Pengujian

2.2.10.1 White Box Testing

Menurut Rosa dan Shalahuddin (2013)[29], *White-Box Testing* (pengujian kotak putih) yaitu menguji perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi – fungsi, masukan, dan keluaran yang sesuai dengan spesifikasi kebutuhan. Pengujian kotak putih dilakukan dengan memeriksa logik dari kode program. Pembuatan kasus uji bisa mengikuti standar-pengujian dari standar pemrograman yang seharusnya. Contoh dari pengujian kotak putih misalkan menguji alur (dengan menelusuri) pengulangan (*looping*) pada logika pemrograman.

2.2.10.2 Black Box Testing

Menurut Rosa dan Shalahuddin (2013)[29], *Black-Box Testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dilakukan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Black-Box Testing dilakukan dengan membuat kasus uji (skenario) yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk menguji harus dibuat dengan skenario benar dan salah.

2.2.11 Framework Yang Digunakan

2.2.11.1 Definisi Framework

Menurut Purbadian (2016)[30], *Framework* adalah sebuah struktur konseptual dasar yang digunakan untuk memecahkan sebuah permasalahan, bahkan isu-isu kompleks yang ada. Sebuah *framework* telah berisi sekumpulan arsitektur/konsep-konsep yang dapat mempermudah dalam pemecahan sebuah permasalahan.

2.2.11.2 Keuntungan Framework

Berikut adalah keuntungan *framework*[30]:

1. **Menghemat waktu pengembangan.** Dengan struktur dan *library* yang telah disediakan oleh *framework* maka tidak perlu lagi memikirkan hal-hal tersebut, jadi Anda hanya fokus ke proses bisnis yang akan dikerjakan.
2. **Reuse of code.** Dengan menggunakan *framework* maka pekerjaan kita akan memiliki struktur yang baku, sehingga kita dapat menggunakannya kembali di proyek-proyek lainnya.
3. **Bantuan komunitas.** Ada komunitas yang siap membantu jika ada permasalahan, selain itu juga bisa berbagi ilmu sehingga dapat meningkatkan kemampuan programming kita.
4. **Kumpulan best practice.** Sebuah *framework* merupakan kumpulan *best practice* yang sudah teruji. Jadi kita dapat meningkatkan kualitas kode kita.

2.2.11.3 Definisi CodeIgniter

Menurut Purbadian (2016)[30], *CodeIgniter* adalah sebuah *web application framework* yang bersifat *open source* yang digunakan untuk membangun aplikasi php dinamis. Tujuan utama pengembangan *CodeIgniter* adalah untuk membantu *developer* untuk mengerjakan aplikasi lebih cepat daripada menulis semua *code* dari awal. *CodeIgniter* menyediakan berbagai macam *library* yang dapat mempermudah dalam pengembangan.

2.2.12 Perangkat Lunak Yang Digunakan

2.2.12.1 XAMPP

Menurut Indrawan (2013)[31], XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

Nama XAMPP merupakan singkatan dari X (sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

2.2.12.2 Web Browser

Menurut Sibero (2013)[32], *web browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi web.

2.2.12.3 Visual Studio Code

Menurut Ummy (2021)[33], *Visual Studio Code* (VS Code) ini adalah sebuah teks *editor* ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks *editor* ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang via *marketplace Visual Studio Code* (seperti C++, C#, Python, Go, Java, dst).