

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Penelitian terdahulu digunakan untuk mengetahui keberhasilan penelitian yang terkait dengan judul penelitian penulis. Beberapa penelitian terdahulu yang berkaitan dengan penelitian ini disajikan pada Tabel II.1.

Tabel II.1. Penelitian Terdahulu

No	Nama Pengarang/ Tahun	Judul	Keterangan
1	Jayati & Wibowo/ 2017	Sistem Pendukung Keputusan Penilaian Tenaga Non PNS Menggunakan <i>Fuzzy Simple Additive Weighting</i> (Studi Kasus: Kantor Disdukcapil Kota Semarang)	Jayati dan Wibowo menggunakan metode <i>Fuzzy Simple Additive Weighting</i> untuk keputusan Penilaian Tenaga Non PNS sedangkan penelitian ini menggunakan metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan customer terhadap produk <i>flooring</i>
2	Muslihudin, dkk/ 2017	Pembuatan Model Penilaian Indeks Kinerja Dosen Menggunakan Metode <i>Fuzzy Simple Additive Weighting</i>	Muslihudin, dkk menggunakan metode <i>Fuzzy Simple Additive Weighting</i> untuk Penilaian Indeks Kinerja Dosen sedangkan penelitian ini menggunakan metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan customer terhadap produk <i>flooring</i> .

3	Suhartati/ 2018	Analisis Faktor Penentu Kepuasan Mahasiswa Terhadap Layanan Proses Belajar Mengajar Dengan Metode SAW Pada STMIK Balikpapan	Suhartati menggunakan metode <i>Simple Additive Weighting</i> untuk Penentu Kepuasan Mahasiswa Terhadap Layanan Proses Belajar Mengajar sedangkan penelitian ini menggunakan metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan customer terhadap produk <i>flooring</i> .
4	Kudiantoro Widiyanto dan Dian Sahara/2019	Metode <i>Fuzzy Simple Additive Weighted</i> dalam mendapatkan karyawan terbaik PT. DEICON KREASINDO	Kudiantoro Widiyanto dan Dian Sahara menggunakan metode <i>Fuzzy Simple Additive Weighted</i> untuk mendapatkan karyawan terbaik sedangkan penelitian ini menggunakan metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan customer terhadap produk <i>flooring</i> .
5	Handayani dan Wibowo/ 2020	Analisis Pengaruh Pemilihan Jumlah Variabel Linguistik Membership Function pada Metode <i>Fuzzy Simple Additive Weighting</i> (FSAW) untuk Perankingan Penerimaan Beasiswa Bagi Siswa Kurang Mampu (Studi Kasus: Sekolah Dasar Negeri Petompon 02 Semarang)	Handayani dan Wibowo menggunakan metode <i>Simple Additive Weighting</i> untuk pemilihan jumlah variabel linguistik membership sedangkan penelitian ini menggunakan metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan customer terhadap produk <i>flooring</i>
6	Saputra dan Pakereng/ 2020	Analisis Perbandingan Metode TOPSIS dan SAW pada Penilaian	Saputra dan Pakereng menggunakan metode <i>Simple Additive Weighting</i> untuk Penilaian Karyawan sedangkan penelitian ini menggunakan

		Karyawan (Studi Kasus: PT Pura Barutama Unit Paper Mill 5, 6, 9)	metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan <i>customer</i> terhadap produk <i>flooring</i>
7	Yunus, dkk/ 2021	Sistem Pendukung Keputusan Penilaian Kinerja Tenaga Kerja Kontrak Menggunakan Metode <i>Simple Additive Weighting</i> (SAW)	Yunus, dkk menggunakan metode <i>Simple Additive Weighting</i> untuk penilaian kinerja tenaga kerja kontrak sedangkan penelitian ini menggunakan metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan <i>customer</i> terhadap produk <i>flooring</i> .
8	Kornelis letelay, dkk/2021	Penentu Status Gizi Buruk Balita Dengan Metode <i>Fuzzy Simple Additive Weighting</i> Pada Puskesmas Tetap, Kabupaten Timor Tengah Selatan	Kornelis letelay, dkk menggunakan Metode <i>Fuzzy Simple Additive Weighting</i> untuk penentu status gizi buruk balita sedangkan penelitian ini menggunakan metode <i>Fuzzy Simple Additive Weighting</i> dan <i>Simple Additive Weighting</i> untuk penilaian kepuasan <i>customer</i> terhadap produk <i>flooring</i> .

Tabel II.1. Penelitian Terdahulu

II.2. Landasan Teori

II.2.1. Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan atau *Decision Support System* secara umum didefinisikan sebagai sebagai sebuah sistem yang mampu memberikan kemampuan baik kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah semi terstruktur. Secara khusus sistem pendukung keputusan didefinisikan sebagai sebuah sistem yang mendukung kerja seorang manager maupun sekelompok manager dalam memecahkan masalah semi terstruktur dengan cara memberikan informasi ataupun ulasan menuju pada keputusan tertentu. Sistem

pendukung keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan dan manipulasi data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat. (Suhartati & Sunarto, 2018).

II.2.1.1. Karakteristik Dan Kemampuan Sistem Pendukung Keputusan.

Dari pengertian Sistem Pendukung Keputusan maka dapat ditentukan karakteristik antara lain:

1. Mendukung proses pengambilan keputusan, menitik beratkan pada *management by perception*.
2. Adanya tatap muka manusia/mesin di mana manusia (*user*) tetap memegang kendali proses pengambilan keputusan.
3. Mendukung pengambilan keputusan untuk membahas masalah terstruktur, semi terstruktur dan tak struktur.
4. Memiliki kapasitas dialog untuk memperoleh informasi sesuai dengan kebutuhan.
5. Memiliki subsistem-subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan item.
6. Membutuhkan struktur data komprehensif yang dapat melayani kebutuhan informasi seluruh tingkatan manajemen. (Panggabean, 2016).

II.2.1.2. Proses Pengambilan Keputusan

Pengambilan keputusan meliputi beberapa tahap dan melalui beberapa proses, pengambilan keputusan meliputi empat tahap yang saling berhubungan dan berurutan. Empat proses tersebut adalah:

1. *Intelligence*

Tahap ini merupakan proses penelusuran dan pendekatan dari lingkup problematika serta proses pengenalan masalah. Data masukan diperoleh, diproses, dan diuji dalam rangka mengidentifikasi masalah.

2. *Design*

Tahap ini merupakan proses menemukan dan mengembangkan alternatif. Tahap ini meliputi proses untuk mengerti masalah, menurunkan solusi dan menguji kelayakan solusi.

3. *Choice*

Pada tahap ini dilakukan proses pemilihan di antara berbagai alternatif tindakan yang mungkin dijalankan. Tahap ini meliputi pencarian, evaluasi, dan rekomendasi solusi yang sesuai untuk variabel hasil pada alternatif yang dipilih.

4. *Implementation*

Tahap implementasi adalah tahap pelaksanaan dari keputusan yang telah diambil. Pada tahap ini perlu disusun serangkaian tindakan yang terencana, sehingga hasil keputusan dapat dipantau dan disesuaikan apabila diperlukan perbaikan. Dalam hal ini, model Simon juga menggambarkan kontribusi Sistem Informasi Manajemen (SIM) dan Ilmu Manajemen / Operations Research (IM / OR) terhadap proses pengambilan keputusan. (Panggabean, 2016).

II.2.1.3. Karakteristik Sistem Pendukung Keputusan

Karakteristik dan kapabilitas kunci dari Sistem Pendukung Keputusan adalah sebagai berikut:

1. Dukungan untuk pengambilan keputusan, terutama pada situasi semiterstruktur dan tak terstruktur.
2. Dukungan untuk semua level manajerial, dari eksekutif puncak sampai manajer lini.
3. Dukungan untuk individu dan kelompok.
4. Dukungan untuk semua keputusan independen dan 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 pengambilan keputusan dapat menghadapi masalah-masalah dengan cara mengadaptasi sistem terhadap kondisi-kondisi perubahan yang terjadi.
8. Pengguna merasa seperti rumah. *User-friendly*, kapabilitas grafis yang kuat, dan sebuah bahasa interaktif yang alami.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, timelines, 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 penganalisisan situasi pengambilan keputusan.
13. Disediakkannya akses untuk berbagai sumber data, format, dan tipe, mulai dari Sistem informasi geografis (GIS) sampai system berorientasi objek.
14. Dapat dilakukan sebagai alat *standalone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di satu organisasi keseluruhan dan di beberapa organisasi sepanjang rantai persediaan. (Panggabean, 2016).

II.2.1.4. Komponen-Komponen Sistem Pendukung Keputusan

Sistem Pendukung Keputusan terdiri dari empat subsistem, yaitu:

1. Manajemen Data, meliputi basis data yang berisi data-data yang relevan dengan keadaan dan dikelola oleh perangkat lunak yang disebut dengan *Database Management System (DBMS)*.
2. Manajemen Model berupa sebuah paket perangkat lunak yang berisi model-model *financial*, *statistic*, *management science*, atau model kuantitatif, yang menyediakan kemampuan analisa dan perangkat lunak manajemen yang sesuai.
3. Subsistem Dialog atau Komunikasi, merupakan subsistem yang dipakai oleh *user* untuk berkomunikasi dan member perintah (menyediakan *user interface*).
4. Manajemen *knowledge* yang mendukung subsistem lain atau sebagai komponen yang berdiri sendiri. (Panggabean, 2016).

II.2.2. Fuzzy Simple Additive Weighting (FSAW)

Metode *Fuzzy Simple Additive Weighting* merupakan salah satu metode FMADM yang sederhana dan paling banyak digunakan. Metode *Fuzzy Simple Additive Weighting* sering dikenal metode penjumlahan terbobot. Konsep dasar metode *Fuzzy Simple Additive Weighting* adalah mencari penjumlahan terbobot dari rating kinerja setiap alternatif pada semua atribut. Metode *Fuzzy Simple Additive Weighting* membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat dibandingkan dengan semua rating alternatif yang diberikan. Langkah penyelesaian *Fuzzy Simple Additive Weighting* menurut mencakup:

1. Menentukan kriteria yang akan digunakan sebagai referensi pembuatan keputusan, dinamakan (X).
2. Menentukan kesesuaian nilai setiap alternatif pada kriteria tertentu.
3. Membuat keputusan berdasarkan matriks kriteria yang berisikan elemen seperti persamaan (1).

$$X = \begin{bmatrix} X_{11} & X_{12} & X_{1n} \\ X_{21} & X_{22} & X_{2n} \\ X_{n1} & X_{n2} & X_{nn} \end{bmatrix} \dots\dots\dots(1)$$

4. Menentukan defuzzifikasi nilai setiap alternatif pada matriks kriteria (X) menghasilkan e_{ij} seperti persamaan (2) yang membentuk matriks E seperti persamaan (3) dimana a_i , b_i , c_i adalah nilai TFN.

$$e_{ij} = \frac{(a_i + b_i + c_i)}{3} \dots\dots\dots(2)$$

$$E = \begin{bmatrix} e_{11} & e_{12} & e_{1n} \\ e_{21} & e_{22} & e_{2n} \\ e_{n1} & e_{n2} & e_{nn} \end{bmatrix} \dots\dots\dots(3)$$

- Melakukan normalisasi matriks **E** yang sudah dilakukan defuzzifikasi sesuai tipe atribut (atribut benefit atau cost) untuk mendapatkan matriks **R** ternormalisasi seperti persamaan (4) dan (5).

$$R_{ij} = \frac{X_{ij}}{\text{Max } X_{ij}} \quad \text{atau} \quad R_{ij} = \frac{\text{Min } X_{ij}}{X_{ij}} \dots\dots\dots(4)$$

$$R_{ij} = \begin{matrix} r_{11} & r_{12} & r_{1n} \\ r_{21} & r_{22} & r_{2n} \\ r_{n1} & r_{n2} & r_{nn} \end{matrix} \dots\dots\dots(5)$$

- Menentukan nilai defuzzifikasi (e_i) dimana a_i , b_i , c_i adalah nilai TFN, lalu nilai normalisasi dari bobot (w_j) setiap criteria. Kemudian, menentukan bobot normalisasi (w_j) yang dilakukan dengan membagi setiap nilai terdefuzzifikasi terhadap nilai total terdefuzzifikasi.
- Menentukan nilai preferensi untuk setiap alternatif menjadi matriks (v_i) seperti persamaan (5).

$$V_i = \sum_{j=1}^n W_j \cdot r_{ij} \dots\dots\dots(6)$$

- Melakukan perankingan hasil perhitungan matriks (v_i) dan memilih alternatif terbaik dengan nilai (v_i) terbesar. Hasil dari penentuan nilai (v_i) untuk alternatif (A_i) dapat dipilih urutan dengan nilai terbesar sebagai best alternatif (A_i) menjadi solusinya. Hasil dari perhitungan yang lebih besar menunjukkan alternatif (A_i) lebih diprioritaskan untuk dipilih. (Jatayi & Wibowo, 2017).

II.2.3. *Simple Additive Weighting (SAW)*

Simple Additive Weighting sering juga dikenal dengan istilah metode penjumlahan berbobot. 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. Menurunkan persamaan-persamaan yang ada dalam metode SAW adalah sebagai berikut:

$$R_{ij} = \frac{X_{ij}}{\text{Max } X_{ij}} \quad \text{atau} \quad R_{ij} = \frac{\text{Min } X_{ij}}{X_{ij}} \dots\dots\dots(1)$$

Keterangan:

Jika I = adalah atribut keuntungan (*benefit*)

Jika j = adalah atribut biaya (*cost*)

R_{ij} = Nilai rating kinerja ternormalisasi

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

Max x_{ij} = Nilai terbesar jika yang dicari adalah atribut keuntungan atau nilai tertinggi

Min x_{ij} = Nilai min jika yang dicari adalah biaya atau nilai terendah

Benefit = Jika nilai terbesar adalah terbaik

Cost = Jika nilai terkecil adalah terbaik.

V_i =

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

Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai:

Keterangan:

V_i = Ranking untuk setiap alternatif

W_j = Nilai bobot dari setiap kriteria

R_{ij} = Nilai rating kinerja Nilai V_i = Nilai yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih.

Tahapan – tahapan yang harus dilalui dalam metode ini diantaranya:

1. Menentukan kriteria apa saja yang akan dijadikan acuan dalam pengambilan keputusan. $C_1, C_2, C_3, \dots, C_i$.
2. Menentukan kandidat simulasi simple yaitu: $A_1, A_2, A_3, \dots, A_i$.
3. Memberikan nilai preferensi (W) oleh pengambil keputusan untuk masing-masing kriteria yang sudah ditentukan. $W = [W_1, W_2, W_3, \dots, W_i]$.
4. Melakukan proses prangkingan dengan cara mengalikan matriks ternormalisasi (N) dengan nilai bobot preferensi (W).
5. Menentukan nilai preferensi untuk setiap alternative (V_i) dengan cara menjumlahkan hasil kali Antara matriks ternormalisasi (N) dengan nilai bobot preferensi (W). (Suhartati & Sunarto, 2018).

II.2.4. Penilaian

Penilaian merupakan keputusan tentang nilai. Oleh sebab itu, langkah selanjutnya sesudah melaksanakan pengukuran adalah penilaian. Penilaian dilakukan setelah peserta didik menjawab beberapa soal yang terdapat pada tes. Kemudian hasil jawaban peserta didik tersebut ditafsirkan dalam bentuk nilai. Penilaian adalah mengambil suatu keputusan terhadap sesuatu dengan ukuran baik dan buruk. Penilaian tersebut bersifat kualitatif. Dalam PP.19/2005 tentang standar nasional pendidikan Bab 1 pasal 1 ayat 17, Penilaian merupakan proses pengumpulan dan pengolahan informasi untuk mengukur pencapaian hasil belajar peserta didik. Jadi dari beberapa pendapat ahli tersebut dan menurut standar nasional pendidikan dapat disimpulkan bahwa penilaian merupakan proses

pengumpulan berbagai data yang dapat memberikan gambaran perkembangan belajar peserta pelatihan, menjelaskan serta menafsirkan hasil pengukuran, menggambarkan informasi mengenai sejauh mana hasil belajar peserta pelatihan atau ketercapaian kompetensi peserta pelatihan. (Meidina, 2020).

II.2.5. Kepuasan *Customer*

Kepuasan *customer* merupakan perusahaan yang mampu memberikan kepuasan pada para *customernya* untuk memenangkan persaingan yang baik. Kepuasan *customer* merupakan cara memenuhi keinginan dan kebutuhan *customer*, melakukan lebih dari pada apa yang diharapkan oleh *customer*, dan membuat *customer* akan kembali lagi untuk membeli. (Terhadap et al., 2020).

II.2.6. Produk

Pengertian produk secara umum adalah sesuatu yang diproduksi oleh tenaga kerja atau sebuah usaha (*effort*) atau hasil dari sebuah tindakan atau sebuah proses. Secara umum produk merupakan sesuatu yang dibuat oleh perorangan atau sekelompok orang sebagai ajang untuk mendapatkan keuntungan melalui proses pertukaran ataupun jual beli dalam transaksi. Produk dengan sesuatu yang memiliki ciri khas yang dapat ditawarkan, diperhatikan, dipakai, dimiliki atau dikonsumsi serta memberikan kepuasan dan memenuhi kebutuhan seseorang maupun perkelompok. Kata produk berasal dari bahasa Inggris *product* artinya “sesuatu yang diproduksi oleh tenaga kerja atau sejenisnya“. Sedangkan bentuk kerja dari kata *product*, yaitu *produce*, merupakan serapan dari bahasa latin *prōdūce(re)*, yang

berarti (untuk) memimpin atau membawa sesuatu untuk maju. (Afif & Krisdianto, 2020). Jenis-jenis produk diantaranya:

1. Produk Konsumsi

Produk Konsumsi sering juga disebut dengan barang kebutuhan sehari-hari (*convenience goods*), Sedangkan pengertian produk konsumsi adalah barang yang umumnya dikonsumsi atau digunakan sendiri oleh anggota keluarga atau perorangan maupun diberikan kepada orang lain dan pembeliannya didasarkan atas kebiasaan dari konsumen itu sendiri. Kategori produk konsumsi dan contohnya: *Convenience Goods* (barang kebutuhan sehari-hari), contoh: sabun, odol, sikat gigi, gula, beras.

- a. *Specialty Goods* (barang-barang istimewa atau produk khusus), contoh: *mobile* Mercedes, TV, radio, perhiasan, alat-alat kosmetik, kamera, *smartphone*.
- b. *Shopping Goods* (barang belanjaan), contoh: jas, sepatu, tas, kaca mata.
- c. *Unsought Goods* (barang-barang yang tidak dicari) adalah barang yang pada umumnya tidak diketahui maupun diketahui hanya saja tidak kepikiran untuk membelinya.

2. Produk Industri

Produk industri adalah barang yang dapat membantu manusia mengerjakan suatu kegiatan produksinya. Contoh barang produk seperti mesin penggiling beras, kulkas, magic jar, kompor, blender, mesin-mesin di pabrik yang membantu proses pengepakan maupun produksinya. Barang terbagi menjadi dua yaitu barang berwujud atau kelihatan bentuk fisiknya dan barang tak berwujud tapi dapat

dirasakan keberadaannya yang disebut barang jasa. Produk barang yang berwujud fisik adalah yang dilihat, diraba atau disentuh, dirasa, dipegang, disimpan, dipindahkan, dan perlakuan fisik lain. Sedangkan pengertian produk jasa adalah setiap tindakan atau kegiatan yang dapat ditawarkan oleh satu pihak kepada pihak lain, yang pada dasarnya tidak berwujud dan tidak mengakibatkan kepemilikan apapun. Produknya dapat dikaitkan atau tidak dikaitkan dengan suatu produk fisik. Contoh layanan gojek, layanan *service* ponsel, layanan taxi. (Afif & Krisdianto, 2020).

II.2.7. *Hypertext Preprocessor* (PHP)

PHP merupakan singkatan dari “*Hypertext Preprocessor*”. PHP adalah sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaknya mirip dengan bahasa pemrograman C, Java, ASP dan Perl ditambah beberapa fungsi PHP yang spesifik dan mudah dimengerti. PHP digunakan untuk membuat tampilan *web* menjadi lebih dinamis, dengan PHP anda bisa menampilkan atau menjalankan beberapa *file* dalam 1 *file* dengan cara di *include* dan *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa *database* walaupun dengan kelengkapan yang berbeda yaitu seperti DBM, MySQL, Oracle. (Huzaeni et al., 2019).

PHP adalah bahasa pemrograman yang sering disisipkan ke dalam HTML. PHP sendiri berasal dari kata *Hypertext Preprocessor*. Sejarah PHP pada awalnya merupakan kependekan dari *Personal Home Page* (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih

bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari *web*. Bahasa pemrograman ini menggunakan sistem *server-side*. *Server-side programming* adalah jenis bahasa pemrograman yang nantinya *script*/program tersebut akan dijalankan/diproses oleh *server*. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari. (Dwi Imaniawan, 2020).

II.2.8. *Hyper Text Markup Language* (HTML)

HTML adalah singkatan dari *Hypertext Markup Language*. HTML memungkinkan seorang *user* untuk membuat dan menyusun bagian paragraf, *heading*, *link* atau tautan, dan *blockquote* untuk halaman *web* dan aplikasi. HTML bukanlah bahasa pemrograman, dan itu berarti HTML tidak punya kemampuan untuk membuat fungsionalitas yang dinamis. Sebagai gantinya, HTML memungkinkan *user* untuk mengorganisir dan memformat dokumen, sama seperti Microsoft Word. (Huzaeni et al., 2019).

HTML ialah kepanjangan dari *Hypertext Markup Language*. Definisi HTML adalah bahasa yang digunakan untuk menulis halaman *web*. fungsi utama HTML ialah memberi perintah pada *browser* untuk melakukan manipulasi tampilan melalui *tag-tag* yang ditulis dalam HTML. (Dwi Imaniawan, 2020).

II.2.9. MySQL

Definisi MySQL merupakan *software* RDMS (*Relational Database Management System*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak pengguna dan dapat melakukan suatu proses secara sinkron atau bersamaan. (Huzaeni et al., 2019).

MySQL merupakan *database engine* atau *server database* yang mendukung bahasa *database* pencarian SQL. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. (Dwi Imaniawan, 2020).

II.2.10. Unified Modeling Language (UML)

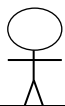
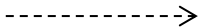
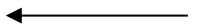
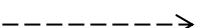
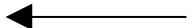
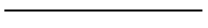
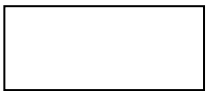
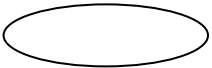
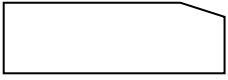
UML yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan system yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. (Andikos, 2019).

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.2.

Tabel II.2. Simbol Use Case

Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa use case sumber secara eksplisit.
	<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang mnghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih

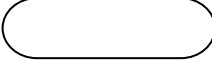
		besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

(Sumber:(Andikos, 2019))

2. Diagram Aktivitas (*Activity Diagram*)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.3.

Tabel II.3. Simbol *Activity Diagram*

Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

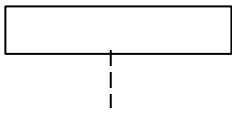
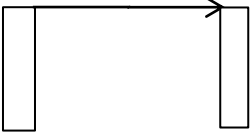
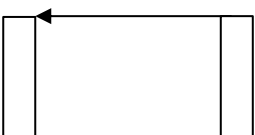
(Sumber:(Andikos, 2019))

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang

digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. Simbol *Sequence Diagram*

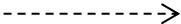
Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

(Sumber: (Andikos, 2019))

4. *Class Diagram* (Diagram Kelas)

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.5.

Tabel II.5. Class Diagram

Gambar	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	<i>Assocation</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber:(Andikos, 2019))