

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Untuk mendukung keberhasilan penelitian ini, penyusun melakukan pendekatan teoritis melalui beberapa literatur yang berhubungan dengan penelitian yang dilakukan. Beberapa uraian penelitian terdahulu yang menjadi acuan dalam penelitian ini yaitu :

Penelitian Yondi Andri Nurdin (2018) dengan judul “Sistem Pendukung Keputusan Penerimaan Bantuan Siswa Miskin (BSM) Berbasis *Online* Dengan Metode KNN (*K-NEAREST NEIGHBOR*) (Studi kasus : SMPN 1 Koto XI Tarusan). Untuk saat ini banyak masyarakat miskin atau kurang mampu yang pendapatannya hanya untuk kebutuhan sehari-hari mereka, sehingga mereka banyak yang tidak sanggup untuk membiayai sekolah anak-anak mereka dan dari situlah faktor banyaknya anak-anak yang tidak bersekolah yang disebabkan oleh ekonomi orang tua mereka. Saat ini biaya ekonomi terus meningkat sehingga banyaknya masyarakat yang mengeluh dengan kebutuhan keseharian mereka. Berdasarkan kriteria yang telah ditentukan Nilai rata - rata raport, Pekerjaan orang tua, Penghasilan orangtua, tanggungan orang tua.

Penelitian Muh Burhanudin (2019) dengan judul “Sistem Pendukung Keputusan Rekomendasi Penerimaan Bantuan Siswa Miskin (BSM) Menggunakan Metode *Simple Additive Weighting* (SAW)”. Tidak sedikit masyarakat Indonesia khususnya di Kota Kediri yang tidak bisa melanjutkan

pendidikan sampai jenjang perguruan tinggi karena keterbatasan biaya. Oleh sebab itu, peran pemerintah pusat dalam mengatasi permasalahan ini adalah dengan diberikan Bantuan Siswa Miskin atau BSM. BSM adalah bantuan dari pemerintah berupa sejumlah uang tunai yang diberikan secara langsung kepada siswa sesuai kriteria yang telah ditetapkan. Perangkingan siswa penerima bantuan siswa miskin. Metode yang digunakan dalam perhitungan adalah *Simple Additive Weighting* perhitungan manual dan perhitungan sistem menunjukkan nilai yang sama. Metode *Simple Additive Weighting* dipilih karena mampu menyeleksi alternatif terbaik dari sejumlah alternatif. Berdasarkan kriteria yang telah ditentukan Absensi siswa yang alfa, Nilai Raport, Penghasilan orang tua dan Jumlah anak yang menjadi tanggungan orang tua

Penelitian Ramadiani Ramadiani dan Auliana Rahmah (2019) dengan judul “Sistem pendukung keputusan pemilihan tenaga kesehatan teladan menggunakan metode *Multi-Attribute Utility Theory*”. Sistem Pendukung Keputusan (SPK) diperlukan untuk memudahkan pekerjaan tim juri dalam memutuskan petugas kesehatan teladan secara objektif, profesional dan transparan. Sistem yang dibangun ini berbasis *stand-alone*, dan dapat dijadikan alternatif terbaik untuk daerah yang akses internetnya belum cukup memadai.

Penelitian ini dilakukan melalui *literature review* dan wawancara langsung pada tim penilai tenaga kesehatan. Model yang digunakan adalah metode *Multi Attribute Utility Theory* (MAUT). Hasil metode MAUT dipilih karena tidak memiliki nilai cost dan benefit dalam menentukan keputusan. Penelitian ini telah menghasilkan rekomendasi untuk pemilihan tenaga kesehatan teladan dengan

hasil akurasi sebesar 86, 67%. . Berdasarkan kriteria yang telah ditentukan Sebagai penggerak pembangunan kesehatan, Tenaga pemberdayaan masyarakat, pelayanan kesehatan, pegawai kesehatan professional, sebagai anggota masyarakat

Penelitian Wira Apriani (2019) dengan judul “Sistem Pendukung Keputusan Pemilihan Pimpinan Dengan Metode *Multi Attribute Utility Theory* (MAUT) di PT. Sagami Indonesia” Metode yang digunakan dalam SPK Pemilihan Pimpinan menggunakan *Multi Attribute Utility Theory* (MAUT) untuk pembobotan kriteria. Perancangan SPK Pemilihan Pimpinan menggunakan model *Waterfall*. Model *Waterfall* terdiri atas tahapan analisis, perancangan, implementasi dan pengujian. SPK menggunakan 4 jenis hak akses (*roles*) yaitu *Administrator*, Manajemen, Kepala Produksi dan Pimpinan Perusahaan. Keluaran sistem disajikan dalam bentuk hasil angka perhitungan MAUT yang dapat dipertimbangkan lebih lanjut oleh pihak pengambil keputusan

Perbedaanya dengan penelitian saya adalah kriteria yang saya gunakan dalam penelitian ini meliputi Penghasilan orang tua, Pekerjaan orang tua, Tanggungan orang Tua, Tempat Tinggal, Identitas orang Tua. Dan disini saya menggunakan metode *Multi-Attribute Utility Theory* (MAUT) dalam pengambilan keputusan Bantuan Siswa Miskin pada SMP Alwasliyah Belawan

II.2. Uraian Teoritis

II.2.1. Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah sebuah sistem berbasis computer interaktif yang dapat membantu para pengambil keputusan untuk memecahkan masalah-masalah tidak terstruktur (Saputra et al., 2019).

Sistem pendukung keputusan memiliki beberapa karakteristik antara lain (Pribadi et al., 2018):

- Mendukung beberapa aplikasi yang saling terkait.
- Dapat digunakan beberapa kali dan bersifat tetap.
- Memiliki dua komponen utama, seperti model dan data.
- Mempunyai kemampuan *what-if analysis* dan *goal seeking analysis*.
- Menggunakan beberapa macam model kuantitatif.

II.2.2. Multi-Attribute Utility Theory (MAUT)

Metode *Multi Attribute Utility Theory* (MAUT) merupakan suatu model pendukung keputusan yang akan menguraikan masalah multi faktor atau multi kriteria yang kompleks dengan cara merubah beberapa kepentingan atau kriteria kedalam nilai numerik dengan skala 0-1 dengan 0 mewakili pilihan terburuk dan 1 terbaik (Indri, 2018).

Nilai evaluasi seluruhnya dapat didefinisikan dengan persamaan:

$$v(x) = \sum_{i=1}^n w_i v_i(x) \dots\dots\dots (2.1)$$

Dimana $v_i(x)$ merupakan nilai evaluasi dari sebuah objek ke i dan w_i merupakan bobot yang menentukan nilai dari seberapa penting elemen ke i terhadap elemen

lainnya. Sedangkan n merupakan jumlah elemen. Total dari bobot adalah 1.

$$\sum_{i=1}^n W_i = 1 \quad \dots\dots\dots (2.2)$$

Untuk setiap dimensi , nilai *evaluation* $v_i(x)$ didefinisikan sebagai penjumlahan dari atribut-atribut yang relevan

$$v_i(x) = \sum_{a \in A}^n W_{ai} V_{ai}(I(a)) \quad \dots\dots\dots (2.3)$$

II.2.2.1. Algoritma Metode MAUT

Algoritma *Multi Attribute Utility Theory* adalah sebagai berikut:

1. Menentukan bobot pada setiap kriteria maupun atribut
2. TransFormasikan data berdasarkan bobot pada setiap kriteria maupun atribut
3. Normalisasikan data

$$U(x) = \frac{x - x_i^-}{x_i^+ - x_i^-} \quad \dots\dots\dots (2.4)$$

Keterangan :

x_i^- : Nilai terburuk dari x

x_i^+ : Nilai terbaik dari x

II.3. *Hypertext Preprocessor (PHP)*

PHP adalah bahasa pemrograman *server-side* yang dirancang untuk perkembangan web. PHP dikatakan *server-side* karena program yang diberikan akan dijalankan/diproses pada komputer yang bertindak sebagai server. PHP yang merupakan sebuah bahasa scripting yang terpasang pada HTML (*Hyper Text Model Language*) membuat halaman web yang menggunakan data dari *database* sangat mudah dilakukan. Berikut adalah daftar *database* yang didukung oleh PHP

(Nugroho, 2017 : 61). Berikut ini contoh script dari PHP bisa dilihat pada gambar

II.1.

A screenshot of a Notepad window titled "Untitled - Notepad". The menu bar includes File, Edit, Format, View, and Help. The text area contains the following PHP script:

```
<html>
<body>
<?php
    echo "hello word";
?>
</body>
</html>
```

Gambar II.1. Script PHP

II.4. *Hyper Text Markup Language (HTML)*

Menurut Dio Lavarino dan Wiyli Yustanti (2017), *Hypertext Markup Language (HTML)* adalah bahasa *markup* yang umum digunakan untuk membuat halaman web. Sebenarnya HTML bukanlah sebuah bahasa pemrograman. Apabila di tinjau dari namanya, HTML merupakan bahasa *markup* atau penandaan terhadap sebuah dokumen teks. Tanda tersebut di gunakan untuk menentukan *Format* atau *style* dari teks yang di tandai. Berikut ini contoh *script* dari HTML bisa dilihat pada Gambar II.2.

A screenshot of a Notepad window titled "Untitled - Notepad". The menu bar includes File, Edit, Format, View, and Help. The text area contains the following HTML script:

```
< html >
< head >
< title > hay </tilte>
< /head >
< body >
```

Gambar II.2. Script HTML

II.5. PhpMyAdmin

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, serverlah yang akan menerjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan (Astria Firman, et al, 2017). Berikut ini merupakan contoh tampilan PhpMyAdmin yang dapat dilihat pada Gambar II.3.

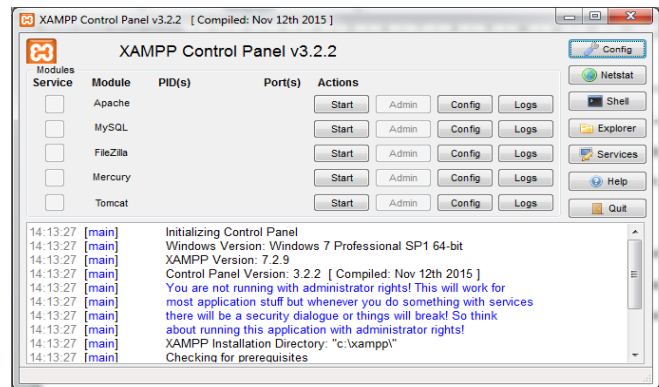


Gambar II.3. PhpMyAdmin

II.6. Xampp

Menurut Afifah (2018), XAMPP merupakan paket web server berbasis *open source* yang dapat dipasang pada beberapa sistem operasi yang ada (*Windows, Linux, dan Mac OS*). XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan meng-*install* XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi web server *Apache*, PHP dan MySQL secara manual.

Berikut ini merupakan contoh tampilan Xampp yang dapat dilihat pada Gambar II.4.



Gambar II.4. Xampp

II.7. UML

Menurut Gellysa Urva dan Helmi Fauzi Siregar (2019: 94), *Unified Modeling Language* (UML) adalah bahasa spesifik 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 digunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML yaitu *Use Case Diagram*, *Activity Diagram*, *Class Diagram* dan *Sequence Diagram*.

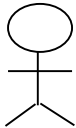
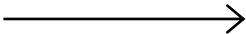
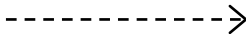
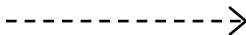
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

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. Simbol-simbol yang digunakan dalam *Use Case* diagram dapat dilihat pada Tabel II.1.

Tabel II.1. Simbol *Use Case* Diagram

Simbol	Deskripsi
<i>Use Case</i> 	Fungsionalitas yang disediakan <i>system</i> sebagai unit-unit yang saling bertukar pesan antar unit atau <i>actor</i> .
Aktor/ <i>Actor</i> 	Orang, proses atau <i>system</i> lain yang berinteraksi dengan <i>system</i> inFormasi yang akan dibuat diluar <i>system</i> yang akan dibuat itu sendiri.
<i>Relationship/Association</i> 	Dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
<i>Extensi/Extend</i> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>Use Case</i> tambahan itu.
<i>Include</i> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan memerlukan <i>Use Case</i> ini untuk menjalankan fungsinya.

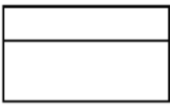
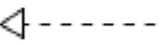
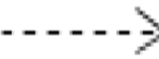
(Sumber : Gellysa Urva dan Helmi Fauzi Siregar : 2019: 94)

2. Class Diagram

Class Diagram adalah deskripsi kelompok obyek -obyek dengan *property* perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *Class Diagram* dapat memberikan pandangan *global* atas sebuah *system*. Hal tersebut tercermin dari *class-class* yang ada dan relasinya satu dengan yang lainnya.

Berikut ini merupakan simbol-simbol yang terdapat pada *Class Diagram*, dapat dilihat pada Tabel II.2.

Tabel II. 2. Simbol Class Diagram

No	Gambar	Nama	Keterangan
1		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
2		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
3		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
4		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
5		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

(Sumber : Gellysa Urva dan Helmi Fauzi Siregar : 2019: 94)

3. Activity Diagram

Activity Diagram adalah *diagram* yang menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Berikut ini merupakan simbol-simbol yang terdapat pada *Activity Diagram*, dapat dilihat pada Tabel II.3.

Tabel II.3. Simbol Activity Diagram

No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antar muka saling berinteraksi satu sama lain
2		<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

(Sumber : Gellysa Urva dan Helmi Fauzi Siregar : 2019: 94)

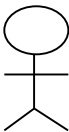
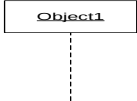
4. Sequence Diagram

Sequence Diagram merupakan salah satu *diagram interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan, *message* (pesan) apa yang dikirim dan kapan pelaksanaannya. *Diagram* ini diatur berdasarkan waktu.

Obyek-obyek yang berkaitan dengan berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut.

Berikut ini merupakan simbol-simbol yang terdapat pada *Sequence Diagram*, dapat dilihat pada Tabel II.4.

Tabel II.4. Simbol *Sequence Diagram*

Simbol	Deskripsi
Aktor/Actor 	Menggambarkan orang yang berinteraksi dengan <i>system</i>
Object Lifeline 	Menggambarkan interaksi antara satu atau lebih <i>actor</i> dengan <i>system</i> memodelkan bagian dari <i>system</i> yang bergabung ada pihak lain disekitarnya
Message (Masukan) 	Menyatakan bahwa suatu objek mengirimkan data atau masukan atau inFormasi ke objek lainnya
Activation 	Menyatakan objek dalam keadaan aktif dan berinteraksi, sebuah yang berhubungan dengan <i>activation</i> ini adalah sebuah tahapan yang dilakukan didalamnya
Message (Keluaran) 	Menyatakan bahwa suatu objek sudah menjalankan suatu operasi atau metode menghasilkan kembalian.

(Sumber : Gellysa Urva dan Helmi Fauzi Siregar : 2019: 94)