

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 :

Agus Perdana Windarto; 2017, dengan judul “Penilaian Prestasi Kerja Karyawan PTPN III Pematang Siantar Dengan Metode Simple Additive Weighted (SAW)”. Dengan adanya penelitian ini penulis telah merancang dan membangun suatu sistem pendukung keputusan menggunakan program web dengan menggunakan database MySQL serta menggunakan sebagai metode, sehingga dapat membantu pihak pimpinan memberikan suatu informasi tentang penilaian prestasi kerja karyawan yang sesuai dengan keinginannya.

Dharyana Suryadijaya; 2017, dengan judul “Spk SAW Penentuan Pagelaran Teater Terbaik”. Sistem pendukung keputusan ini dapat membantu proses menentukan grup teater terbaik yang akan dipilih sebagai pagelaran teater terbaik berdasarkan penjumlahan terbobot dari prioritas kriteria-kriteria yang telah ditentukan oleh panitia pagelaran teater sebelumnya. Perhitungan nilai bobot yang berbeda dari setiap kriterianya akan menghasilkan nilai yang berbeda dan menghasilkan keputusan yang berbeda, karena penilaian para juri pagelaran teater terhadap seni pertunjukan yang disajikan oleh grup teater memiliki sentuhan

tersendiri di dalam penilaian dan hal ini menyebabkan perbedaan pada penilaian per kriterianya.

Adil Setiawan, 2017, dengan judul “Implementasi Metode SAW Dalam Penerimaan Siswa Baru Pada SMA Negeri 16 Medan” Penelitian ini menghasilkan suatu sistem pendukung keputusan yang berfungsi sebagai alat bantu SMA Negeri 16 Medan dalam mengambil keputusan pada proses seleksi penerimaan siswa baru.

Muqorobin,et.al; 2019, dengan judul “Sistem Pendukung Keputusan Penerimaan Beasiswa dengan Metode SAW” Berdasarkan pembahasan dan pengujian sistem yang dilakukan sebelumnya dapat diambil kesimpulan bahwa sistem pendukung keputusan penerimaan beasiswa dengan menggunakan metode SAW telah dibuat dan dapat berjalan dengan baik sesuai dengan hasil scenario uji pada pengujian fungsionalitas menghasilkan output yang diharapkan. Penerapan metode SAW dapat menghasilkan perankingan sebagai rujukan terbaik bagi calon penerima beasiswa dan sebagai pendukung dalam pengambilan keputusan oleh Tim Seleksi Beasiswa. Metode SAW dapat menerapkan jenis beasiswa BP dan BKM sesuai dengan kriteria dan bobot yang telah ditentukan.

Alex Rikki,et.al; 2016, dengan judul “Sistem Pendukung Keputusan Penerimaan Karyawan Baru Di Departemen Kehakiman TIMOR - LESTE Dengan Menggunakan Metode SAW” Aplikasi sistem pendukung keputusan pemilihan karyawan terbaik berhasil dibuat dengan metode Simple Additive Wheigting (SAW) pada PT ISS Indonesia dan perlu pemahaman dan ketelitian yang tinggi dalam penggunaannya. Dalam penerapan metode Simple Additive

Wheighting (SAW) pada PT ISS Indonesia Medan harus memenuhi empat kriteria yang diproses yaitu, kehadiran, SOP, Kedisiplinan, dan Grooming. Maka sistem berhasil menentukan karyawan terbaik sesuai dengan kriteria dan bobot yang ditentukan semula sebelum perhitungan.

Berdasarkan hasil penelitian yang terdahulu, maka dibuatlah kesimpulan untuk merancang sebuah aplikasi sistem pengambilan keputusan untuk mengimplementasikan sebuah aplikasi yang belum pernah dibuat atau dibangun oleh orang lain. Sehingga pada penulisan Skripsi ini dibuatlah sebuah judul **“Sistem Pengambilan Keputusan Penerimaan Karyawan Pada PT. Adipa Karya Menggunakan Algoritma SAW.** Dengan adanya sistem pengambilan keputusan ini dapat mempermudah PT. Adipa Karya dalam penerimaan karyawan dengan beberapa aspek kriteria penilaian yang sudah ditentukan seperti, keahlian, kedisiplinan, kepribadian, kerja team, komunikasi, penampilan dan sikap, sistem pengambilan keputusan dilakukan secara otomatis dengan menampilkan perhitungan pada sistem aplikasi dengan penilaian yang sudah di inputkan.

II.2. Landasan Teori

II.2.1. Sistem Pendukung Keputusan

Sistem pendukung Keputusan sebagai sebuah sistem berbasis komputer yang terdiri atas komponen komponen antara lain komponen sistem bahasa (*language*), komponen sistem pengetahuan (*knowledge*) dan komponen sistem pemrosesan masalah (*problem processing*) yang saling berinteraksi satu dengan yang lainnya. Hal yang perlu diperhatikan di sini adalah bahwa keberadaan SPK

bukan untuk menggantikan tugas manajer, tetapi untuk menjadi sarana penunjang bagi mereka. SPK merupakan implementasi teori-teori pengambilan keputusan yang telah diperkenalkan oleh ilmu-ilmu seperti *operation research* dan *management science*. Hanya bedanya adalah bahwa dahulu untuk mencari penyelesaian masalah yang dihadapi harus dilakukan perhitungan iterasi secara manual. Dalam kedua bidang ilmu di atas, dikenal istilah *decision modeling*, *decision theory*, *decision analysis* yang pada hakekatnya adalah merepresentasikan permasalahan manajemen yang dihadapi setiap hari ke dalam bentuk kuantitatif (Frans Ikorasaki, 2018; 2).

II.2.2. Rekrutmen

Rekrutmen merupakan suatu cara dalam perusahaan untuk mendapatkan sumber daya manusia yang benar-benar tepat untuk menduduki suatu posisi tertentu yang ditawarkan oleh perusahaan. Selain itu juga rekrutmen menjadi cara perusahaan untuk mengeliminasi karyawan lain untuk menduduki posisi yang ditawarkan (M. Setiadi, dkk; 2016).

II.2.3. Karyawan

Karyawan adalah mereka yang bekerja pada suatu badan usaha atau perusahaan, baik swasta ataupun pemerintah, dan diberi imbalan kerja sesuai dengan peraturan perundang-undangan yang berlaku (Silvia Rahmawati & Shalahuddin; 2017).

II.2.4. SAW

Metode *Simple Additive Weighting* (SAW) sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode *Simple Additive Weighting* (SAW) adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Metode *Simple Additive Weighting* (SAW) membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat dibandingkan dengan semua ranting alternatif yang ada. 3 Langkah penyelesaian *Simple Additive Weighting* (SAW) adalah sebagai berikut:

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 proses perankingan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik (A_i) sebagai solusi.

$$r_{ij} = \left\{ \frac{X_{ij}}{\text{Max}_{X_{ij}}} \right. \quad (\text{II.1})$$

$$r_{ij} = \left\{ \frac{\text{Min}_{X_{ij}}}{X_{ij}} \right. \quad (\text{II.2})$$

Keterangan:

r_{ij} = nilai rating kinerja ternormalisasi

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

Max X_{ij} = nilai terbesar dari setiap kriteria

Min X_{ij} = nilai terkecil dari setiap kriteria

Benefit = jika nilai terbesar adalah nilai terbaik

Cost = jika nilai terkecil adalah nilai terbaik

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

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

diberikan sebagai:

$$V_i = \sum_{j=1}^n w_j r_{ij} \quad (\text{II.3})$$

Keterangan:

V_i = ranking untuk setiap alternatif

W_j = nilai bobot dari setiap kriteria

r_{ij} = nilai rating kinerja ternormalisasi

Nilai V_i yang lebih besar mengidentifikasi bahwa alternatif A_i lebih dipilih (Satria Abadi, dkk; 2017).

II.2.5. PHP

PHP adalah bahasa yang dirancang secara khusus untuk penggunaan pada *Web*. *PHP* adalah *tool* untuk pembuatan halaman web dinamis. Pada awalnya *PHP* merupakan kependekan dari *Personal Home Page* (Situs Personal). *PHP* pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu *PHP* masih bernama *FI* (*Form Interpreted*), yang wujudnya

berupa sekumpulan *script* yang digunakan untuk mengolah data form dari *web*. Saat ini *PHP* adalah singkatan dari *PHP: Hypertext Preprocessor*, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: *PHP: Hypertext Preprocessor* (Ahmad Lutfi, 2017 ; 105).

II.2.6. MySQL

MySQL adalah salah satu aplikasi DBMS (*Database Management System*) yang sudah sangat banyak digunakan oleh para pemrogram aplikasi web. Dalam sistem database tak relasional, semua informasi disimpan pada satu bidang luas, yang kadangkala data di dalamnya sangat sulit dan melelahkan untuk diakses. Tetapi *MySQL* merupakan sebuah sistem database relasional, sehingga dapat mengelompokkan informasi ke dalam tabel-tabel atau grup-grup informasi yang berkaitan. Setiap tabel memuat bidang-bidang yang terpisah, yang mempresentasikan setiap bit informasi. *MySQL* menggunakan indeks untuk mempercepat proses pencarian terhadap baris informasi tertentu. *MySQL* memerlukan sedikitnya satu indeks pada tiap tabel. Biasanya akan menggunakan suatu *primary key* atau pengenal unik untuk membantu penjejakan data (Ahmad Lutfi, 2017; 106).

II.2.7. Database

Database ialah suatu wadah untuk menampung sebuah data yang ada pada sebuah sistem. Database juga bias diartikan sebagai kumpulan data. Database juga

biasa dikenal formal dan tegas. Database juga bisa diartikan dengan kumpulan data yang terintegrasi yang dapat dimanipulasi, diambil dan dicari secara cepat. (Hesananda, et al. 2017)

II.2.8. *Unified Modeling Language (UML)*

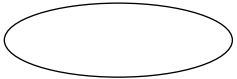
Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language (UML)*. UML adalah satu alat bantu yang sangat handal didunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam membentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain (Munawar ; 2018 : 49).

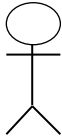



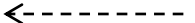
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasiskan UML adalah sebagai berikut :

1. *Use case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan tipikal interaksi antara (pengguna) sebuah *system* dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem yang dipakai (Munawar ; 2018 : 89).

Tabel II.1. Simbol *Use Case Diagram*

Gambar	Keterangan
	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan

	dengan menggunakan kata kerja di awal nama <i>Use Case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i> , tetapi tidak memiliki <i>control</i> terhadap <i>Use Case</i> .
	Asosiasi antara aktor dan <i>Use Case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>Use Case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>Use Case</i> lain (<i>required</i>) atau pemanggilan <i>Use Case</i> oleh <i>Use Case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>Use Case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber : Munawar ; 2018 : 93)

2. *Class Diagram* (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Class diagram merupakan diagram statis dari suatu aplikasi. *Class Diagram* tidak hanya digunakan untuk memvisualisasikan, menggambarkan, dan mendokumentasikan berbagai aspek sistem tetapi juga untuk membangun kode eksekusi (*executable code*) dari aplikasi perangkat lunak (Munawar ; 2018 : 101).

Tabel II.2. Simbol Class Diagram




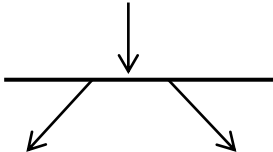
<i>Multiplicity</i>	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

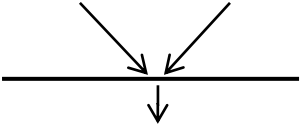
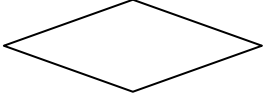
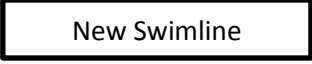
(Sumber : Munawar ; 2018 : 101)

3. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram bagian penting dari UML yang menggambarkan aspek dinamis dari sistem. Logika prosedural, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah dideskripsikan dalam *activity diagram* (Munawar ; 2018 : 137).

Tabel II.3. Simbol Diagram Aktivitas

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

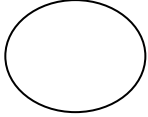
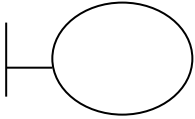
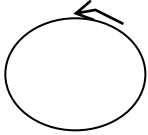

	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

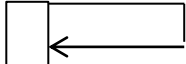
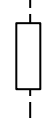

(Sumber : Munawar ; 2018 : 137)

4. Diagram Urutan (*Sequence Diagram*)

Sequence diagram adalah salah satu *interaction diagram*. Karena *sequence diagram* mengacu kepada obyek, maka sebelum membuat diagram ini class diagram sudah harus teridentifikasi (Munawar ; 2018 : 186).

Tabel II.4. Simbol Diagram Urutan

Gambar	Keterangan
	<i>EntityClass</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .

	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Munawar ; 2018 : 186)