

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Telah ada beberapa penelitian yang dilakukan terkait dengan Analisa penggabungan metode AHP dan TOPSIS dalam penentuan daerah terdampak virus Covid-19 untuk penetapan PSBB, diantaranya adalah :

Berdasarkan penelitian yang dilakukan oleh Rusydi Umar, Abdul Fadlil, Yuminah dengan judul “Sistem Pendukung Keputusan dengan Metode AHP untuk Penilaian Kompetensi *Soft Skill*” Penelitian ini menghasilkan penilaian prioritas kompetensi *soft skill* yang dibutuhkan perusahaan sebagai berikut: Komunikasi 48%, Kerja sama 27%, Kejujuran 16 %, dan interpersonal 10%. Hasil penelitian ini membuktikan bahwa metode AHP dapat digunakan pada penilaian kompetensi *soft skill* karyawan sampai menentukan nilai prioritas karyawan tertinggi.

Berdasarkan penelitian yang dilakukan oleh Hylenarti Hertyana dengan judul “Sistem Pendukung Keputusan Penentuan Karyawan Terbaik Menggunakan Metode Topsis” Hasil penelitian menggunakan metode Topsis yang penulis lakukan, ada 3 kriteria untuk penentuan karyawan terbaik diantaranya Masa Kerja (A1) diperoleh dari lama karyawan bekerja, Kinerja (A2) diperoleh dari penilaian kinerja karyawan dan Absensi (A3) diperoleh dari kondite kehadiran karyawan. Sistem Penunjang Keputusan atau SPK yang dikembangkan dengan menggunakan Metode Topsis dapat membantu dalam mengambil keputusan

dalam menentukan karyawan terbaik. Berdasarkan Perhitungan menggunakan metode TOPSIS, maka di dapat keputusan bahwa V5 (Karyawan 5) merupakan karyawan terbaik karena memiliki nilai yang terbaik.

Berdasarkan penelitian yang dilakukan oleh Indra Herman Firdaus, Gunawan Abdillah, Faiza Renaldi dengan judul “Sistem Pendukung Keputusan Penentuan Karyawan Terbaik Menggunakan Metode AHP Dan Topsis” Penelitian ini menghasilkan sebuah sistem pendukung keputusan yang dapat merekomendasikan karyawan terbaik pada PT South Pacific Viscose berdasarkan kriteria yang telah ditetapkan yaitu: pengetahuan, kemampuan, sikap, absensi, dan kerjasama dengan menggunakan metode AHP dan TOPSIS. Dilakukan uji coba berupa memasukkan sample data karyawan sebanyak 300 orang kemudian berhasil diolah dalam waktu 0,9531 detik sehingga terbukti sistem ini melakukan perhitungan lebih cepat dibanding sebelumnya.

Perbedaan penelitian yang penulis lakukan dengan penelitian sebelumnya adalah penulis menggunakan penggabungan metode AHP dan TOPSIS dalam penentuan daerah terdampak virus Covid-19 untuk penetapan PSBB. Hasil dari penelitian ini dapat menjadi solusi yang optimal dan akurat bagi pemerintah daerah dalam penentuan daerah terdampak virus Covid-19 untuk penetapan PSBB.

II.2. Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) merupakan sistem yang dapat memberikan pemecahan masalah, melakukan komunikasi untuk pemecahan

masalah tertentu dengan terstruktur maupun tidak terstruktur. SPK didesain untuk dapat digunakan dan dioperasikan dengan mudah oleh orang yang hanya memiliki kemampuan dasar pengoperasian komputer. SPK dibuat dengan menerapkan adaptasi kompetensi yang tinggi sehingga dapat dijadikan sebagai alternatif dalam pengambilan sebuah keputusan. Sistem Pendukung Keputusan terdiri atas dua kata kunci, yaitu sistem informasi dan keputusan. Sistem informasi merupakan serangkaian prosedur formal dengan tahapan di mana data dikelompokkan, diproses sehingga menghasilkan informasi yang selanjutnya diberikan kepada pengguna. Keputusan adalah serangkaian kegiatan untuk memilih suatu tindakan dalam memecahkan masalah. Tindakan memilih dari alternatif yang dihadapi yang didasarkan pada fakta dan dilakukan melalui pendekatan sistematis yang dapat memberikan solusi terbaik yang dilakukan oleh manajer disebut pengambilan keputusan. (Umar, 2018 : 2)

II.3. PSBB

Pembatasan Sosial Berskala Besar (PSBB) merupakan salah satu jenis penyelenggaraan kekarantinaan kesehatan dalam suatu wilayah, seperti karantina rumah, karantina rumah sakit, dan karantina wilayah. Tujuan PSBB ini yaitu untuk mencegah meluasnya penyebaran penyakit yang sedang terjadi antar orang di suatu wilayah tertentu. Untuk pembatasan kegiatan yang dilakukan meliputi peliburan pada sekolah-sekolah dan tempat kerja, pembatasan kegiatan keagamaan, dan/atau pembatasan kegiatan di tempat atau fasilitas umum.

II.4. Metode AHP

Analytical Hierarchical Process (AHP) merupakan hierarki dengan *input* atau masukan utama berupa pandangan manusia. Dikembangkan oleh Prof. Thomas Lorie Saaty dari Wharton Business School awal tahun 1970. Metode ini digunakan untuk mencari urutan atau ranking prioritas dari berbagai alternatif dalam pemecahan masalah. AHP banyak digunakan untuk mengekspresikan pengambilan suatu keputusan yang sangat efektif dari suatu permasalahan yang kompleks.

Penentuan prioritas dengan metode AHP dilakukan melalui beberapa tahapan yaitu:

- a. Menyusun hierarki.
- b. Menilai kriteria dan alternatif.
- c. Memilih prioritas.
- d. Menentukan nilai konsistensi logis.

Di dalam mengambil keputusan, penting untuk diketahui baik tidaknya nilai konsistensi yang digunakan. Adapun langkah-langkah yang harus dilakukan adalah:

- 1) Menghitung nilai perbandingan yang sudah ditetapkan dengan membagi nilai skala yang ada di setiap sel dibagi dengan nilai sel prioritas.
- 2) Hasil perhitungan dari langkah no. 1 di masing-masing sel dibagi dengan jumlah total di masing-masing kolom.
- 3) Mencari Eigen dengan cara menghitung rata-rata per baris.

4) Mencari Lamda (λ) dengan cara mengalikan masing-masing nilai eigen per baris dengan jumlah total per kolom.

5) Mencari lamda (λ maks) dengan cara menjumlahkan hasil lamda

e. Menentukan nilai indeks konsistensi (CI)

Rumus yang digunakan sebagai berikut:

$$CI = \frac{\lambda_{\max} - n}{n-1} \dots\dots\dots (1)$$

Keterangan:

n = banyaknya kriteria.

f. Menentukan rasio konsistensi (CR)

Rumus yang digunakan sebagai berikut:

$$CR = \frac{CI}{RI} \dots\dots\dots (2)$$

Keterangan:

CR = Rasio Konsistensi

CI = Indeks Konsistensi

RI = Indeks Random Konsistensi

g. Memeriksa konsistensi hierarki.

Jika hasil perhitungan nilai rasio konsistensi lebih dari 10%, maka harus diperbaiki atau dihitung ulang. Tapi jika rasio konsistensi kurang atau sama dengan 0,1 maka dapat dinyatakan benar nilai perhitungannya.

II.5. Metode TOPSIS

Metode TOPSIS adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yoon dan Hwang pada tahun

1981. Metode ini merupakan salah satu metode yang banyak digunakan untuk menyelesaikan pengambilan keputusan secara praktis. TOPSIS memiliki konsep dimana alternatif yang terpilih merupakan alternatif terbaik yang memiliki jarak terpendek dari solusi ideal positif dan jarak terjauh dari solusi ideal negatif. Semakin banyaknya faktor yang harus dipertimbangkan dalam proses pengambilan keputusan, maka semakin relatif sulit juga untuk mengambil keputusan terhadap suatu permasalahan. Apalagi jika upaya pengambilan keputusan dari suatu permasalahan tertentu, selain mempertimbangkan berbagai faktor/kriteria yang beragam, juga melibatkan beberapa orang pengambil keputusan. Permasalahan yang demikian dikenal dengan permasalahan *Multiple Criteria Decision Making* (MCDM). Dengan kata lain, MCDM juga dapat disebut sebagai suatu pengambilan keputusan untuk memilih alternative terbaik dari sejumlah alternative berdasarkan beberapa kriteria tertentu. Metode TOPSIS digunakan sebagai suatu upaya untuk menyelesaikan permasalahan *Multiple Criteria Decision Making* (MCDM). Hal ini disebabkan konsepnya sederhana dan mudah dipahami, komputasinya efisien dan memiliki kemampuan untuk mengukur kinerja relative dari alternatif-alternatif keputusan.

Langkah-langkah yang dilakukan dalam menyelesaikan suatu permasalahan menggunakan metode TOPSIS adalah sebagai berikut :

1. Menggambarkan alternatif (m) dan kriteria (n) ke dalam sebuah matriks, dimana X_{ij} adalah pengukuran pilihan dari alternatif ke-i dan kriteria ke-j. Matriks ini dapat dilihat pada persamaan tiga.

$$D = \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \\ X_{31} & X_{32} & X_{33} \end{bmatrix} \dots\dots\dots (3)$$

2. Membuat matriks R yaitu matriks keputusan ternormalisasi. Setiap normalisasi dari nilai rij dapat dilakukan dengan perhitungan menggunakan persamaan 4.

$$r_{ij} = \frac{X_{ij}}{\sqrt{\sum_{i=1}^m X_{ij}}} \dots\dots\dots (4)$$

3. Membuat pembobotan pada matriks yang telah dinormalisasi Setelah dinormalisasi, setiap kolom pada matriks R dikalikan dengan bobot-bobot (w_j) untuk menghasilkan matriks pada persamaan lima.

$$D = \begin{bmatrix} W_1 r_{11} & W_1 r_{12} & W_n r_n \\ W_2 r_{21} & \dots & \dots \\ W_j r_{m1} & W_1 r_{m2} & W_j r_{mm} \end{bmatrix} \dots\dots\dots (5)$$

4. Menentukan nilai solusi ideal positif dan solusi ideal negatif. Solusi ideal dinotasikan A^+ , sedangkan solusi ideal negatif dinotasikan A^- . Persamaan untuk menentukan solusi ideal dapat dilihat pada persamaan enam.

$$A^+ = \{ \max V_{ij} | j \in J, (\min V_{ij} | j \in j'), i = 1,2,3,\dots,m \} = V_1^+, V_2^+, \dots, V_n^+$$

$$A^- = \{ \max V_{ij} | j \in J, (\min V_{ij} | j \in j'), i = 1,2,3,\dots,m \} = V_1^-, V_2^-, \dots, V_n^-$$

$$J = \{j=1,2,3,\dots,n \text{ dan } j \text{ merupakan } \textit{benefit criteria}\}$$

$$J' = \{j=1,2,3,\dots,n \text{ dan } j \text{ merupakan } \textit{cost criteria}\} \dots\dots\dots (6)$$

5. Menghitung *separation measure*. Separation measure ini merupakan pengukuran jarak dari suatu alternatif ke solusi ideal positif dan solusi ideal negatif.

- a. Perhitungan solusi ideal positif dapat dilihat pada persamaan tujuh:

$$S_i^+ = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^+)^2} \dots\dots\dots (7)$$

Dengan $i = 1,2,3,\dots,m$

b. Perhitungan solusi ideal negatif dapat dilihat pada persamaan delapan:

$$S_i^- = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^-)^2} \dots\dots\dots (8)$$

Dengan $i = 1,2,3,\dots,m$

6. Menghitung nilai preferensi untuk setiap alternatif. Untuk menentukan ranking tiap-tiap alternatif yang ada maka perlu dihitung terlebih dahulu nilai preferensi dari tiap alternatif. Perhitungan nilai preferensi dapat dilihat melalui persamaan sembilan.

$$C_i^+ = \frac{S_i^-}{S_i^+ + S_i^-} \dots\dots\dots (9)$$

Dimana $0 < C_i^+ < 1$ dan $i=1,2,3,\dots,m$

Setelah didapat nilai C_i^+ , maka alternatif dapat diranking berdasarkan urutan C_i^+ . Dari hasil perankingan ini dapat dilihat alternatif terbaik yaitu alternatif yang memiliki jarak terpendek dari solusi ideal dan berjarak terjauh dari solusi ideal negatif.

II.6. *Unified Modelling Language* (UML)


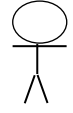

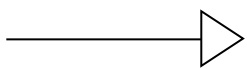
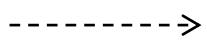
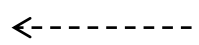
Unified Modelling Language (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan tek-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metologi berorientasi objek. (Simatupang, 2018 : 5)

Dalam membangun perancangan sistem dengan alat bantu perancangan *Unified Modeling Language (UML)* ada beberapa tahapan yang akan dilakukan, yaitu sebagai berikut :

1. Use Case Diagram

Use Case Diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibangun. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibangun. 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. Adapun simbol *Use Case Diagram*.

Gambar II.1. Simbol Use Case diagram


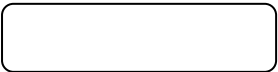
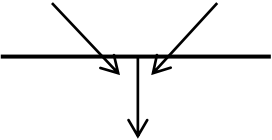
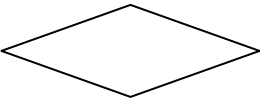

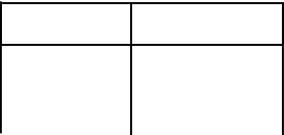
Simbol	Deskripsi
<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
<p>Actor</p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
<p>Asosiasi / <i>association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor.
<p>Generalisasi</p> 	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
	<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 : Simatupang ; 2019)

2. Diagram Aktivitas (*Activity Diagram*)

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Penekanan pada diagram aktivitas adalah menggambarkan aktivitas sistem atau aktivitas yang dapat dilakukan oleh sistem, bukan apa yang dilakukan aktor. Adapun simbol-simbol *activity diagram*.

Gambar II.2. Simbol *Activity Diagram*

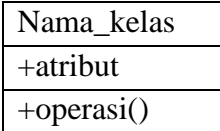
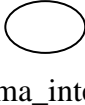

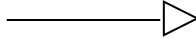

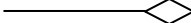
Simbol	Deskripsi
Status Awal 	<i>Start point</i> , status awal aktivitas sistem diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
Aktivitas 	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis, biasanya diawali dengan kata kerja
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	Status akhir yang dilakukan sistem
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas lain

(Sumber : Simatupang ; 2019)

3. Class Diagram (Diagram Kelas)

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Berikut adalah simbol-simbol *class diagram*.

Gambar II.3. Simbol Class Diagram

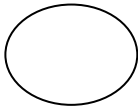
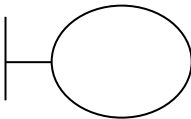
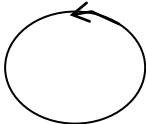
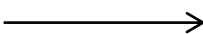
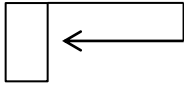
Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem.
Antar muka / <i>interface</i> 	Sama dengan konsep interface dalam pemrograman berorientasi objek.
Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan kelas yang lain.
Generalisasi 	Relasi antar kelas dengan makna generalisasi-generalisasi spesialisasi (umum khusus)
Kebergantungan / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua bagian (whole-part)



(Sumber : Simatupang ; 2019)

4. Diagram Urutan (*Sequence Diagram*)

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu”. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada use case. Adapun simbol *Sequence Diagram*.

Gambar II.4. Simbol *Sequence Diagram*

Gambar	Keterangan
	<i>Entity Class</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>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> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri

	<p><i>Activation, 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 : Simatupang ; 2019)

II.7. Basis Data (Database)

Database merupakan sekumpulan dari data yang terintegrasi. Sistem *Database* merupakan sistem komputerisasi yang bertujuan untuk menyimpan sejumlah data sehingga memudahkan pengguna untuk mendapatkan dan mengupdate informasi sesuai dengan kebutuhan informasi yang disimpan dalam *Database* dapat berupa text maupun angka. (Mardiono, 2019 : 3)

II.7.1. MySQL (Database)

MySQL merupakan suatu jenis database server yang sangat terkenal. *MySQL* termasuk jenis RDBMS (Relational Database Management System). *MySQL* berfungsi untuk mengolah *Database* menggunakan bahasa *SQL*. *MySQL* bersifat *open source* sehingga bisa menggunakannya secara gratis. Pemrograman *PHP* juga sangat mendukung dengan *database MySQL*. (Fahrozi, 2018 : 3)

II.7.2. Normalisasi

Normalisasi adalah salah satu cara untuk meminimalisir pengulangan data (*data redundancy*), normalisasi akan diperlukan jika ada indikasi bahwa tabel

yang kita buat tidak baik (terjadi pengulangan informasi, potensi inkonsistensi data pada operasi perubahan, tersembunyinya informasi tertentu dan lain sebagainya) dan diperlukan supaya jika tabel-tabel yang didekomposisi kita gabung kembali dapat menghasilkan tabel awal sebelum didekomposisi, sehingga diperoleh tabel yang baik. Hasil dari normalisasi adalah himpunan-himpunan data (*tabel-tabel*) dalam bentuk normal (*normal form*). (Efendy, 2018 : 1)

II.8. PHP

PHP merupakan Bahasa *scripting* seperti HTML. Dalam pengembangan web pada HTML yang memungkinkan dibuatnya aplikasi dinamis yang memungkinkan adanya pengolahan data dan pemrosesan data. Semua *sintax* yang diberikan akan sepenuhnya dijalankan pada server sedangkan yang dikirimkan ke *browser* hanya hasilnya saja. Kemudian merupakan bahasa berbentuk *script* yang ditempatkan dalam *server* dan diproses di *server*.

PHP dikenal sebagai sebuah bahasa *scripting*, yang menyatu dengan tag-tag HTML, dieksekusi di server, dan digunakan untuk membuat halaman web yang dinamis seperti halnya *Active Server Pages* (ASP) atau *Java Server Pages* (JSP). PHP merupakan sebuah software *Open Source*. (Fahrozi, 2018 : 2)

