

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Berdasarkan penelitian dari Albert Julius Olanta, 2019 “Perbandingan Metode ANP Dan AHP Dalam “Pemilihan Jasa Kurir Logistik Oleh Penjual Gadget Online”. Penelitian ini menghasilkan suatu hasil pengolahan data dan analisis didapatkan bahwa terdapat sedikit perbedaan dalam nilai bobot kriteria antara metode ANP dan AHP. Akan tetapi urutan perolehan skor untuk tiap alternatif tidak mengalami perbedaan antara menggunakan metode ANP dan Metode AHP.

Berdasarkan penelitian dari Moh Ramdhan Arif Kaluku, 2017 “Penerapan perbandingan metode ahp-topsis dan anp-topsis mengukur kinerja sumber daya manusia di gorontalo” ,2017 Penelitian ini menghasilkan sebuah aplikasi web back end sistem. Back end sistem dikelola oleh administrator dalam manajemen kuesioner, manajemen konten, dan pengolahan data dari hasil pengolahan data key performance indicator dan kinerja SDM .

Berdasarkan penelitian dari Muhammad Iqbal Baskor, 2021 “Sistem pendukung keputusan untuk menentukan siswa berprestasi menggunakan metode ANP berbasis web”, Penelitian ini telah berhasil mengimplementasikan sistem pendukung keputusan untuk menentukan siswa berprestasi menggunakan metode analytic network process berbasis web dan sistem pendukung keputusan dengan metode ANP dapat membantu pihak sekolah dalam menentukan siswa berprestasi

Berdasarkan penelitian dari Ida Bagus Kurniawan, 2019 “sistem pendukung keputusan pemilihan mahasiswa berprestasi di universitas dhyana pura menggunakan metode AHP, electre dan topsis” Hasil dari penelitian ini dapat perangkingan alternatif sangat dipengaruhi oleh pembobotan kriteria menggunakan metode AHP.

Berdasarkan penelitian dari Fajar Delli Wihartiko 2018 “sistem pendukung keputusan strategi tindakan atas pelanggaran siswa dengan metode analytical network process” dari hasil penelitian didapatkan pemeringkatan alternatif yang menghasilkan bahwa penanganan pelanggaran siswa berupa peringatan tertulis dan hukuman edukatif merupakan hasil yang mendapatkan nilai tertinggi sesuai dari ketiga pendapat pakar dalam penelitian ini yaitu Wakil Kepala Sekolah Bidang Kesiswaan, Koordinator Bimbingan Konseling dan Pembina OSIS dengan menggunakan metode ANP.

II.2. Landasan Teori

II.2.1. Sistem Pendukung Keputusan

Sistem pendukung keputusan merupakan suatu sistem interaktif yang mendukung keputusan dalam proses pengambilan keputusan melalui alternatif-alternatif yang diperoleh dari hasil pengolahan data, informasi yang menggunakan model-model keputusan, basis data, dan pemikiran manajer sendiri, proses modeling interaktif dengan komputer untuk mencapai pengambilan keputusan oleh manajer tertentu (Rahmad Kurniawan; 2019).

II.2.2. Siswa Berprestasi

Siswa berprestasi merupakan proses mendapatkan dan menggunakan informasi mengenai murid terbaik di sekolah. Sekolah selaku pembuat keputusan harus mengambil keputusan yang tepat mengenai pemilihan siswa atau siswi, bila hal ini dilakukan dengan baik dan benar akan menjamin hasil pemilihan yang berkualitas dan dapat dipertanggung jawabkan. Selain itu pemilihan juga dituntut untuk menghasilkan keputusan yang relevan dan akurat (Alyauma hajjah 2021).

II.2.3. Metode Analytic Network Process (ANP)

Metode Analytic Network Process (ANP) merupakan pengembangan metode Analytical Hierarchy Process (AHP). Metode ANP mampu memperbaiki perbedaan struktur pada AHP berupa kemampuan mengakomodasi keterkaitan antar kriteria atau alternatif. Keterkaitan pada metode ANP ada dua jenis yaitu keterkaitan dalam satu set elemen (inner dependence) dan keterkaitan antar elemen yang berbeda (outerdependence). Adanya keterkaitan tersebut menyebabkan metode ANP lebih kompleks dibanding metode AHP (Albert Julius Olanta, 2019).

II.2.4. Tahapan Metode *Analytic Network Process* (ANP)

Tahapan proses atau langkah-langkah dalam menyelesaikan permasalahan dengan menggunakan metode ANP adalah sebagai berikut :

- a. Defenisikan masalah dan menentukan kriteria solusi yang diinginkan

- b. Tentukan pembobotan komponen dari sudut pandang managerial.
- c. Buat matriks perbandingan berpasangan yang menggambarkan kontribusi atau pengaruh setiap elemen atas setiap kriteria. Perbandingan dilakukan berdasarkan penilaian dari pengambil keputusan dengan menilai tingkat kepentingan suatu elemen.
- d. Hitung eigenvector pembobotan

$$\begin{bmatrix} e_1 = b_1 / \sum_{i=1}^n b_i \\ E = e_2 = b_1 / \sum_{i=1}^n b_i \\ \dots\dots\dots \\ e_1 = b_1 / \sum_{i=1}^n b_i \end{bmatrix} \quad (1)$$

Keterangan : e_i = menyatakan elemen matriks E baris ke- i .

- e. Menghitung konsistensi yang diperoleh, apabila nilai kurang dari 10% maka nilai tersebut dapat dikatakan konsisten.

$$CI = \frac{\lambda_{\max} - n}{n - 1} \quad (2)$$

$$CR = CI / RCI \quad (3)$$

keterangan :

CI = *consistency index* (indeks konsistensi)

n = *orde* dari matriks

$\lambda_{\max}$ = nilai *eigen vector* terbesar dari matriks ber-*ordo* n .

RI = *random index* (*index* acak) untuk setiap matriks ber-*ordo* n.

- f. Ulangi langkah 3, 4, dan 5 untuk semua kriteria yang ada.
- g. Kemudian buat *unweighted* supermatrik yaitu dengan cara memasukkan semua *eigen vector* yang telah dihitung pada langkah 5 kedalam sebuah supermatriks.
- h. Langkah selanjutnya adalah membuat *weighted supermatrik* dengan cara melakukan perkalian setiap isi *unweighted supermatrik* terhadap matriks perbandingan kriteria (kluster matrik).

$$W_w = T_z \cdot W \quad (4)$$

Keterangan :

W_w = *Weighted supermatrik*.

T_z = Matrik perbandingan.

W = *Unweighted supermatrik*

- i. Membuat *limiting supermatriks* dengan cara memangkatkan supermatriks secara terus menerus hingga angka disetiap kolom dalam satu baris sama besar, setelah itu lakukan normalisasi terhadap *limiting supermatriks*.

$$\lim_{k \rightarrow \infty} w^k \quad (5)$$

- j. Ambil nilai dari alternatif yang dibandingkan kemudian lakukan normalisasi untuk mengetahui hasil akhir perhitungan.

II.2.5. Metode *Analytical Hierarchy Process* (AHP)

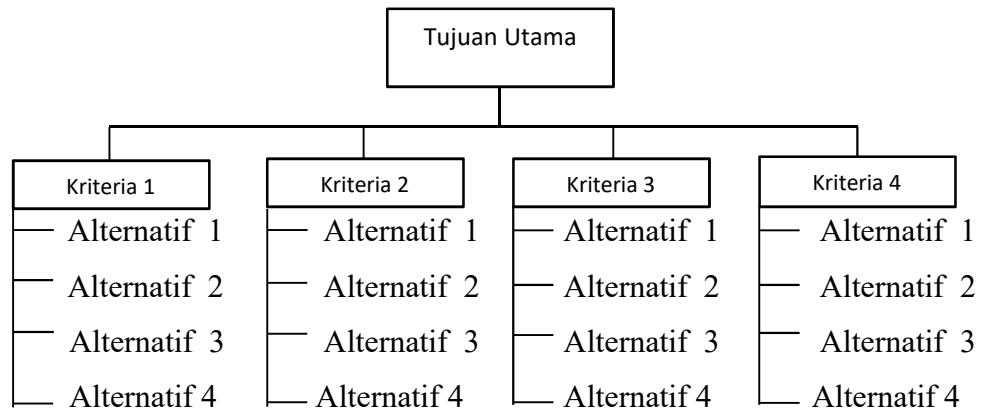
Analytical Hierarchy Process (AHP) merupakan suatu model pendukung keputusan yang dikembangkan oleh Thomas L. Saaty. Metode *Analytical Hierarchy Process* (AHP) telah banyak digunakan untuk membantu pengambilan keputusan. *Analytic Hierarchy Process* (AHP) adalah teori pengukuran melalui perbandingan berpasangan dan tergantung pada penilaian dari para ahli untuk mendapatkan skala prioritas. Perbandingan yang dibuat menggunakan skala penilaian mutlak yang mewakili, berapa banyak lagi, satu elemen mendominasi yang lain sehubungan dengan atribut yang diberikan. Penilaian mungkin tidak konsisten, dan bagaimana mengukur inkonsistensi dan meningkatkan penilaian, bila mungkin untuk mendapatkan konsistensi yang lebih baik adalah kekhawatiran dari *Analytical Hierarchy Process* (AHP). Skala prioritas yang berasal disintesis dengan mengalikan dengan prioritas node induk dan menambahkan untuk semua node tersebut. (Wirhan Fahrozi, 2016; 216).

II.2.6. Tahapan *Analytical Hierarchy Process*

Menurut Kadarsyah dan Ali (1998), langkah-langkah yang dilakukan dalam metode *Analytical Hierarchy Process* (AHP) sebagai berikut:

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan.
2. Membuat struktur hierarki yang diawali dengan tujuan utama

Secara umum, struktur hierarki dapat dilihat pada gambar 1.



Gambar II.1. Struktur Hierarki AHP

Sumber: (Agnia Eva Munthafa dan Husni Mubarak, 2017; 194).

3. Membuat matrik perbandingan berpasangan yang menggambarkan kontribusi relatif atau pengaruh setiap elemen terhadap tujuan atau kriteria yang setingkat di atasnya.

Tabel II.1. Matriks Perbandingan Berpasangan

	Kriteria-1	Kriteria-2	Kriteria-3	Kriteria-n
Kriteria-1	K11	K12	K13	K1n
Kriteria-2	K21	K22	K23	K2n
Kriteria-3	K31	K32	K33	K3n
Kriteria-m	Kn1	Kn2	Kn3	Kmn

Sumber: (Agnia Eva Munthafa dan Husni Mubarak, 2017; 194).

4. Mendefinisikan perbandingan berpasangan sehingga diperoleh jumlah penilai seluruhnya sebanyak $n \times [(n-1)/2]$ buah, dengan n adalah banyaknya elemen yang dibandingkan.

Tabel.II. 2. Skala Penilaian Perbandingan Berpasangan

1	Kedua elemen sama penting
3	Elemen yang satu sedikit lebih penting dari elemen lainnya
5	Elemen yang satu lebih penting dari elemen lainnya
7	Elemen yang satu sangat penting dari elemen lainnya
9	Elemen yang satu mutlak sangat penting dari elemen lainnya
2,4,6,8	Nilai-nilai antaradua nilai pertimbangan yang berdekatan
Kebalikan	Jika aktivitas i mendapat satu angka dibandingkan dengan aktivitas j , maka j memiliki nilai kebalikan dibandingkan i

(sumber: Albert Julius Olanta, vol 18, No 2019)

II.2.7. Database

Basis data (*database*) adalah kumpulan dari berbagai data yang saling berhubungan satu dengan yang lainnya. Basis data tersimpan di perangkat keras, serta dimanipulasi dengan menggunakan perangkat lunak. Pendefinisian basis data meliputi spesifikasi dari tipe data, struktur dan batasan dari data atau

informasi yang akan disimpan. *Database* Terdiri dari database, file, record, field dan karakter (Rudi Febrianto dkk, 2017).

II.2.8. MySQL

MySQL adalah salah satu jenis *database server* yang sangat populer, hal ini disebabkan karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. MySQL bersifat *Open Source*, *software* ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat MySQL), bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi (Rudi Febrianto dkk, 2017).

II.2.9. Web Server

Web server adalah *server* yang melayani permintaan klien terhadap halaman web seperti *apache*, IIS (*Internet Information Server*) dan berkomunikasi dengan *Middleware* untuk menterjemahkan kode-kode tertentu, menjalankan kode-kode tersebut dan memungkinkan berinteraksi dengan basis data, PHP atau ASP. Adapun arsitektur aplikasi *server* adalah sebagai berikut :

1. *Browser* atau klien berinteraksi dengan *web server*.
2. Secara *internal web server* berinteraksi dengan *middleware*.
3. *Middleware* yang berhubungan dengan *database* (Rudi Febrianto dkk, 2017).

II.2.10. *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 (Ade Hendini ; 2016).

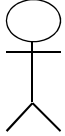
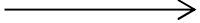
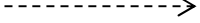
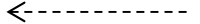
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

1. Use case Diagram

Use case diagram merupakan pemodelan untuk sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Ade Hendini,2016:108). Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada tabel II.3 dibawah ini :

Tabel II.3. 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</i>

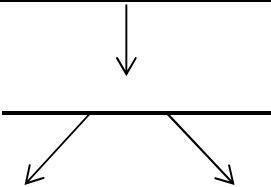
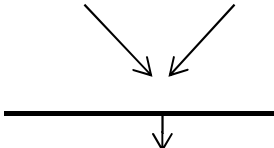
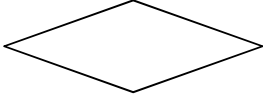
	<i>Case.</i>
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasikan aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam 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 : Ade Hendini ; 2016)

2. Diagram Aktivitas (*Activity Diagram*)

3. *Activity diagram* adalah *diagram* yang menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang terdapat pada *Activity Diagram*, dapat dilihat pada tabel II.4 sebagai berikut :

Tabel II.4. Simbol *Activity Diagram*

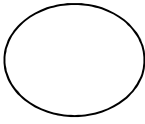
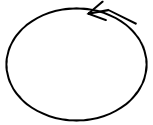
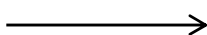
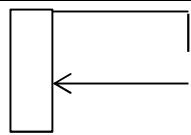
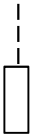
Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau 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, false</i> .
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

(Sumber : Ade Hendini ; 2016 : 109)

4. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* dapat dilihat pada tabel II.5 dibawah ini :

Tabel II.5. 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>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang berbagai melibatkan objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Ade Hendini ; 2016 : 110)

5. *Class Diagram* (Diagram Kelas)

Class diagram 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* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi : Kelas (*Class*), Relasi *Assosiations*, *Generalitation* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* (Ade Hendini, 2016 : 111).

Tabel II.6. 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 : Ade Hendini ; 2016 : 111)