

BAB II

TINJAUAN PUSTAKA

2.1. Sistem Pendukung Keputusan

2.1.1. Sistem

Sistem adalah kumpulan komponen yang saling berhubungan atau terintegrasi untuk mencapai tujuan tertentu (Fakhriza dkk, 2022). Sistem dapat juga didefinisikan sebagai sekumpulan elemen yang terdiri dari manusia, alat, ide, dan prosedur yang digunakan untuk mencapai tujuan tertentu (Jamalia, 2022).

2.1.2. Karakteristik Sistem

Sistem memiliki beberapa karakteristik atau sifat tertentu (Sutabri, 2012).

1. Komponen Sistem (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berhubungan dan bekerja sama untuk membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa subsistem yang memiliki sifat-sifat yang dapat melakukan fungsi tertentu dan berdampak pada proses secara keseluruhan.

2. Batasan Sistem (*Boundary*)

Batasan atau ruang lingkup sistem adalah area yang membatasi suatu sistem dengan sistem lain atau lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*)

Apapun yang ada di luar area atau batasan sistem yang mempengaruhi operasinya disebut lingkungan luar sistem.

4. Penghubung Sistem (*Interface*)

Penghubung sistem adalah media membantu menghubungkan subsistem.

5. Masukan Sistem (*Input*)

Masukan sistem merupakan energi yang dimasukkan ke dalam sistem untuk mengoperasikan komputer dan data yang akan diolah menjadi informasi

6. Keluaran Sistem (*Output*)

Energi yang telah diolah menjadi keluaran sistem yang berguna, yang dapat digunakan sebagai masukan untuk subsistem yang lain, seperti sistem informasi.

Keluaran ini terdiri dari informasi yang dapat digunakan untuk membuat keputusan atau sebagai input untuk subsistem lainnya.

7. Proses Pengolah Sistem (*Process*)

Pengolah sistem adalah suatu proses yang mengubah masukan menjadi keluaran yang menguntungkan.

8. Sasaran Sistem (*Objective*)

Suatu sistem harus memiliki tujuan dan sasaran yang jelas dan deterministik. Jika tidak, sistem tidak akan berguna.

2.1.3. Sistem Pendukung Keputusan

Pada hakikatnya, proses pembuatan keputusan disebabkan oleh adanya masalah dalam mencapai suatu tujuan tertentu. Tujuan pembuatan keputusan adalah untuk mengatasi atau memecahkan masalah sebagai upaya untuk mencapai tujuan tersebut dapat dilaksanakan dengan baik dan efektif. (Sutabri, 2014).

Sistem pendukung keputusan adalah bagian dari sistem informasi berbasis komputer, termasuk sistem berbasis pengetahuan, yang membantu orang membuat keputusan dalam suatu perusahaan atau lembaga pendidikan. Sistem pendukung keputusan disebut juga sebagai sistem informasi *interaktif* yang memberikan informasi, memodelkan, dan memanipulasi data (Nasution, 2018). Sistem pendukung keputusan biasanya untuk membantu pekerjaan manusia menjadi lebih efisien dan akurat. Tanpa adanya sistem pendukung keputusan, proses penentuan keputusan dilakukan secara manual. Karena kesalahan perhitungan, keakuratan proses tersebut dapat diragukan (Dewi, 2021).

2.1.4. Fungsi Sistem Pendukung Keputusan (SPK)

Menurut Adam dan Lengkong (2019), fungsi dari sistem pendukung keputusan :

1. Membantu memecahkan masalah semi terstruktur
2. Membantu manajer dan pimpinan dalam pengambilan keputusan
3. Meningkatkan efisiensi dan efektivitas proses pengambilan keputusan

2.1.5. Keunggulan Sistem Pendukung Keputusan (SPK)

Menurut Indriani (2019), sistem pendukung keputusan (SPK) memiliki keunggulan:

1. Meningkatkan kemampuan pengambilan keputusan dalam memproses data / informasi
2. Menghemat waktu dalam memecahkan masalah
3. Mempercepat solusi dan hasil yang dapat diandalkan
4. Meningkatkan kepercayaan terhadap keputusan

2.1.6. Kelemahan Sistem Pendukung Keputusan (SPK)

Menurut Indriani (2019), kelemahan dari sistem pendukung keputusan (SPK) :

1. Kemampuan manajemen dan bakat manusia tertentu tidak dapat dimodelkan
2. Sistem pendukung keputusan tidak dapat memberikan alternatif pengetahuan
3. Proses – proses yang dapat dilakukan oleh sistem pendukung keputusan biasanya bergantung pada kemampuan perangkat lunak.

2.1.7. Karakteristik / Sifat Sistem Pendukung Keputusan (SPK)

Secara umum, sistem pendukung keputusan memiliki karakteristik / sifat (SPK) sebagai berikut (Nasution, 2018):

1. Memberikan dukungan kepada pengambil keputusan, terutama dalam situasi yang semi struktur dan tak terstruktur
2. Memberikan dukungan untuk berbagai tingkat manajer, mulai dari eksekutif puncak hingga manajer lini
3. Memberikan dukungan kepada individu dan kelompok
4. Mendukung berbagai keputusan sekuensial dan independen
5. Memberikan dukungan untuk berbagai jenis keputusan
6. Mendukung berbagai proses dan gaya pengambilan keputusan
7. Sistem mampu beradaptasi dengan cepat, yang berarti pengambilan keputusan dapat menghadapi masalah - masalah baru dan pada saat yang sama dapat menyelesaiakannya dengan mengadaptasikan sistem terhadap kondisi perubahan yang terjadi.

8. Mudah digunakan, memiliki kemampuan grafis yang kuat, dan mampu menyelesaikan masalah dengan cepat.
9. Keefektifan (akurasi, waktu, dan kualitas) pengambilan keputusan meningkat dibandingkan dengan efisiensi (biaya).
10. Pengambil keputusan memiliki kendali penuh atas setiap langkah proses pengambilan keputusan dalam memecahkan masalah.
11. Pengguna akhir dapat membuat dan mengubah sistem sederhana.
12. Model menganalisis situasi pengambilan keputusan.
13. Akses ke berbagai jenis data, sumber, dan format dari sistem informasi geografi
14. Dapat digunakan secara mandiri oleh seorang pengambil keputusan di satu tempat atau didistribusikan di seluruh organisasi dan di beberapa organisasi sepanjang rantai persediaan

2.1.8. Komponen - Komponen Sistem Pendukung Keputusan

Sistem diperlukan untuk membantu keakuratan data yang dihasilkan lebih nyata dalam proses pengambilan keputusan. Beberapa komponen yang diperlukan untuk sistem pendukung keputusan (Nurjannah, 2021):

1. Manajemen data, yang terdiri dari data yang sesuai dengan fakta dan disimpan ke dalam basis data
2. Manajemen model, yang menunjukkan rancangan software yang akan dibangun
3. Subsistem antarmuka pengguna, yang menjelaskan komponen yang digunakan pengguna untuk berkomunikasi saat memberi kode perintah
4. Manajemen pengetahuan yang berdiri sendiri dan membantu komponen lain berinteraksi

2.1.9. Fase – Fase Pengambilan Keputusan

Ada empat fase dalam pengambilan keputusan (Nasution, 2018).

1. Kecerdasan (*Intelligence*)

Tahap di mana masalah didefinisikan dan informasi yang diperlukan tentang masalah dan keputusan yang akan diambil.

2. Perancangan (*Design*)

Tahap ini mewakili model sistem yang akan dibangun berdasarkan asumsi yang telah ditetapkan, dan model masalah dibuat, diuji, dan divalidasi.

3. Pemilihan (*Choice*)

Tahap ini adalah proses melakukan pengujian dan memilih keputusan terbaik berdasarkan kriteria yang telah ditetapkan.

4. Implementasi (*Implementation*)

Tahap pelaksanaan dari keputusan yang telah dibuat. Pada tahap ini, tindakan yang telah direncanakan harus diambil untuk memastikan bahwa hasil keputusan dapat diubah sesuai kebutuhan.

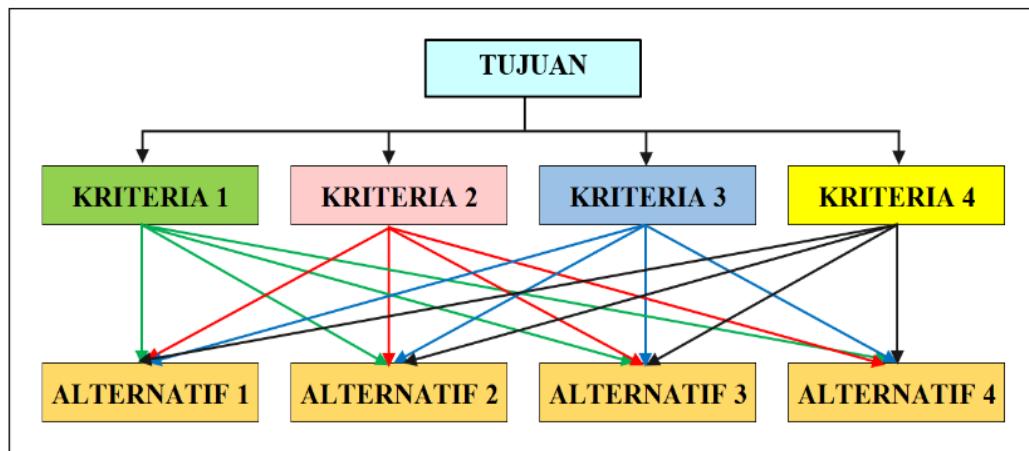
2.2 Metode Algoritma *Analytic Network Process* (ANP)

Metode *Analytic Network Process* (ANP) merupakan pengembangan dari metode *Analytical Hierarchy Process* (AHP) yang dapat memperbaiki kelemahan AHP yaitu kemampuan untuk mengakomodasi keterkaitan antar kriteria dan alternatif. *Analytic Network Process* (ANP) juga merupakan teori matematis yang memiliki kemampuan untuk menganalisis pengaruh dengan menggunakan pendekatan asumsi – asumsi untuk menyelesaikan berbagai jenis masalah (Meliala dkk, 2020). Dalam SPK, ANP digunakan sebagai *algoritma* yang dapat membantu pengambilan keputusan dengan menampilkan urutan prioritas perangkingan untuk memilih alternatif terbaik. Penyelesaian ANP dilakukan dengan mempertimbangkan kompleksitas masalah disertai adanya skala prioritas yang menghasilkan prioritas terbesar (Atsari dkk, 2020).

Proses *Analytic Network Process* (ANP) secara sistematis juga dapat menjelaskan model faktor – faktor termasuk faktor *dependence* dan *feedbacknya*. Dalam jaringan ANP, level – level ANP disebut *cluster* dan masing – masing level dapat memiliki kriteria dan alternatif. Dengan mempertimbangkan semua kriteria, elemen dalam suatu komponen atau cluster dapat mempengaruhi elemen lain dalam komponen atau cluster yang sama dan juga dapat mempengaruhi elemen dalam suatu cluster yang lain. Yang diperlukan dalam ANP adalah mengetahui keseluruhan pengaruh dari semua elemen (Meliala dkk, 2020).

2.2.1. Membangun Model Keputusan

Memisahkan keputusan menjadi hierarki tujuan, standar, dan alternatif (Allen, 2021).



Gambar 2.1. *Hierarchy ANP*

2.2.2. Tahapan - Tahapan *Analytic Network Process (ANP)*

Berikut adalah beberapa langkah yang digunakan dalam metode *Analytic Network Process* (ANP) (Annisa, 2020) yaitu

1. Mendefinisikan masalah dan menentukan kriteria yang sesuai
2. Menentukan pembobotan
3. Membuat matriks perbandingan berpasangan dengan menggunakan skala perbandingan 1–9, yang dapat dilihat pada Tabel 2.1 berikut ini.

Tabel 2.1. Skala Numerik 1 – 9 Pada ANP

Intensitas / Pembobotan	Keterangan	Penjelasan
1	Sama penting	Kedua elemen mempunyai pengaruh yang sama
3	Sedikit lebih penting	Salah satu elemen memiliki pengaruh yang sedikit lebih penting dibandingkan elemen lainnya
5	Lebih penting	Salah satu elemen memiliki pengaruh lebih penting dibandingkan elemen lainnya
7	Sangat lebih penting	Salah satu elemen memiliki pengaruh yang sangat lebih penting dibandingkan elemen lainnya

9	Mutlak lebih penting	Salah satu elemen mutlak memiliki pengaruh yang lebih penting dibandingkan elemen lainnya
2,4,6,8	Nilai Tengah	Suatu penilaian yang diberikan jika terdapat keraguan terhadap kedua elemen
Kebalikannya		Jika elemen x mempunyai salah satu nilai diatas pada saat dibandingkan dengan elemen y, maka elemen y mempunyai nilai kebalikan bila dibandingkan dengan elemen x

4. Menghitung *Eigenvector* (w)

Eigenvector (w) adalah rumus untuk menghitung bobot pada prioritas matriks

$$e_1 = b_i / \sum_{i=1}^n b_i$$

$$E = e_2 = b_i / \sum_{i=1}^n b_i$$

$$e_i = b_i / \sum_{i=1}^n b_i$$

Keterangan :

e_i = menyatakan elemen matriks e ke baris ke – i

5. Menghitung nilai konsisten.

Apabila nilainya kurang dari 10% , nilai tersebut dianggap konsisten, sehingga nilai tersebut dapat dirumuskan sebagai berikut:

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

$$CR = \frac{CI}{RI}$$

Keterangan :

λ_{\max} : *Eigen value* terbesar dari matriks berordo – n

n : Ordo dari matriks

CI : Indeks konsisten

RI : Indeks acak untuk setiap matriks berordo – n

6. Mengulang kembali langkah 3,4 dan 5 untuk setiap kriteria.

7. Membuat *unweighted supermatix* dengan memasukkan nilai eigenvector dari langkah 5 ke dalam supermatriks.

8. Membuat *weighted supermatrix* dengan mengkalikan setiap isi *unweighted supermatrix* dengan matriks perbandingan kriteria (kluster matriks)

$$Ww = Tz \cdot W$$

Keterangan :

Ww : *weighted supermatrix*

Tz : matriks perbandingan

W : *unweighted supermatrix*

9. Membuat *limiting supermatrix* dengan memangkatkan *supermatrix* berkala hingga pada di setiap kolom menghasilkan nilai yang sama besar dan menormalisasikan *limiting supermatrix* dengan rumus : $\lim_{k \rightarrow \infty} w^k$
10. Menentukan alternatif terbaik

2.2.3. Kelebihan *Analytic Network Process (ANP)*

Menurut Simanjuntak (2022), ANP memiliki beberapa kelebihan :

1. Dapat melakukan perhitungan dengan kriteria data yang dapat dilihat
2. Dapat membuat model dengan membandingkan level dan kriteria
3. Dengan mempertimbangkan semua kriteria yang ada, model aplikasi ANP ini lebih berguna untuk kriteria kuantitatif dan kualitatif.

2.2.4. Kekurangan *Analytic Network Process (ANP)*

Menurut Simanjuntak (2022), *Analytic Network Process (ANP)* memiliki beberapa kekurangan :

1. Membutuhkan banyak ruang dan waktu untuk membuat satu model ANP
2. Untuk menghasilkan hasil perbandingan yang lebih akurat dibutuhkan sub perbandingan minimal 5 sampai 7 konsumen
3. Untuk mendapatkan hasil yang tidak *valid* penilaian *expertise* lebih akurat

2.3. Penilaian Kinerja Guru

2.3.1. Guru

Guru adalah seseorang yang bertanggung jawab untuk mendidik dan memberikan ilmu pengetahuannya kepada anak didiknya sehingga mereka dapat

menjadi orang yang cerdas. Guru juga merupakan seorang pendidik profesional untuk terciptanya generasi penerus bangsa (Putra, 2021).

Peraturan Menteri Pendidikan Nasional Nomor 35 Tahun 2010 menyatakan "Guru adalah pendidik profesional dengan tugas pokoknya mendidik, mengajar, membimbing, mengarahkan, membentuk, mengukur, dan menilai peserta didik pada pendidikan anak usia dini melalui jalur pendidikan formal, pendidikan dasar, dan pendidikan menengah."

2.3.2. Kinerja Guru

Kinerja adalah prestasi yang dicapai oleh seseorang dalam melaksanakan tugas atau pekerjaan sesuai dengan standar dan kriteria yang telah ditetapkan (Muhiban, 2020). Kinerja dapat juga diartikan sebagai tingkat pelaksanaan tugas yang dapat dicapai seseorang dengan menggunakan kemampuan yang ada dan batasan yang telah ditetapkan untuk mencapai tujuan organisasi (Jamalia, 2022).

Kinerja guru adalah tingkat keberhasilan guru dalam menjalankan tugas pendidikan sesuai dengan tanggung jawab dan kewenangannya berdasarkan standar kinerja yang telah ditetapkan selama periode waktu tertentu untuk mencapai tujuan pendidikan (Jamalia, 2022).

2.3.3. Penilaian Kinerja Guru

Penilaian kinerja adalah serangkaian kegiatan yang mengukur hasil kerja terhadap standar kerja yang telah ditetapkan oleh lembaga pendidikan untuk mengetahui tingkat kinerjanya (Irfan, 2020).

Penilaian kinerja guru adalah sebuah proses membandingkan kinerja aktual dengan kinerja ideal. Tujuan dari penilaian ini adalah untuk mengetahui seberapa berhasil guru melakukan tugas pokoknya dalam jangka waktu tertentu (Jamalia, 2022).

Berdasarkan Peraturan Menteri Pendayagunaan Aparatur Negara dan Reformasi Birokrasi Nomor 16 Tahun 2009 bahwa penilaian kinerja guru adalah evaluasi dari semua kegiatan tugas utama yang dilakukan oleh guru dalam rangka pembinaan karir dan jabatannya. Penilaian kinerja guru dimaksudkan untuk

mengevaluasi seberapa baik guru memiliki kemampuan untuk menerapkan semua kompetensi yang diperlukan dalam proses pembelajaran, pembimbingan, atau pelaksanaan tugas tambahan yang berkaitan dengan fungsi sekolah atau madrasah (Kemendikbud, 2016).

2.3.4. Sistem Penilaian Kinerja Guru

Sistem penilaian kinerja guru dilaksanakan menurut standar kompetensi guru yaitu ukuran dari penguasaan pengetahuan, keterampilan dan sikap yang ditunjukkan oleh seorang guru. Standar ini diperlukan agar guru dapat bekerja sesuai dengan tugas, kualifikasi, dan jenjang pendidikan (Jamalia, 2022).

Penguasaan kompetensi guru ini sangat penting untuk mencapai proses pembelajaran berkualitas tinggi, pembimbingan peserta didik dan pelaksanaan tugas tambahan dan/atau tugas lain yang relevan sesuai dengan fungsi sekolah atau madrasah (Kemendikbud, 2016).

Menurut Peraturan Menteri Pendidikan Nasional Nomor 16 Tahun 2007 tentang Standar Kualifikasi Akademik dan Kompetensi Guru, ada 4 (empat) kompetensi untuk guru kelas atau mata pelajaran yang harus dimiliki guru yaitu pedagogik, kepribadian, sosial, dan profesional dengan 14 (empat belas) sub kompetensi yang dapat dilihat pada Tabel 2.2. berikut ini (Kemendikbud, 2016).

Tabel 2.2 Kompetensi dan Sub Kompetensi Guru Kelas / Mata Pelajaran

No .	Kompetensi	Sub Kompetensi
1.	Pedagogik	1.1. Mengenal karakteristik peserta didik
		1.2. Menguasai teori belajar dan prinsip – prinsip pembelajaran
		1.3. Pengembangan kurikulum
		1.4. Kegiatan pembelajaran yang mendidik
		1.5. Pengembangan potensi peserta didik
		1.6. Komunikasi dengan peserta didik
		1.7. Penilaian dan evaluasi
2.	Kepribadian	2.1. Bertindak sesuai dengan norma agama, hukum, sosial dan kebudayaan nasional
		2.2. Menunjukkan pribadi yang dewasa dan teladan

		2.3. Etos kerja, tanggung jawab yang tinggi dan rasa bangga menjadi guru
3.	Sosial	3.1. Bersikap inklusif, bertindak obyektif serta tidak diskriminatif
		3.2. Komunikasi dengan sesama guru, tenaga kependidikan orang tua peserta didik dan masyarakat
4.	Profesional	4.1. Penguasaan materi, struktur, konsep dan pola piker keilmuan yang mendukung mata pelajaran yang diajarnya
		4.2. Mengembangkan keprofesionalan melalui tindakan yang efektif

Setiap tahun, kepala sekolah melakukan penilaian kinerja guru di sekolah. Dalam proses penilaian, penilai menetapkan nilai untuk setiap kompetensi pada skala nilai 1– 4. Sebelum pemberian nilai tersebut, penilai terlebih dahulu mengidentifikasi melalui pengamatan dan pemantauan serta bukti – bukti yang ada. Selain itu, hasil penilaian kinerja guru juga dipengaruhi oleh tingkat kehadiran guru di sekolah. Oleh karena itu, nilai penilaian kinerja guru adalah gabungan dari hasil penilaian kinerja guru dan tingkat kehadiran guru di sekolah (Kemendikbud, 2016).

2.3.5. Syarat - Syarat Sistem Penilaian Kinerja Guru

Untuk memberikan hasil yang akurat dan tepat, sistem penilaian kinerja guru harus memenuhi tiga persyaratan yaitu (Hikam, 2018) :

1. Valid yaitu elemen yang dinilai benar-benar mengukur bagian – bagian tugas guru dalam melaksanakan pembelajaran, pembimbingan, dan tugas lain yang berkaitan dengan fungsi sekolah.
2. Reliabel yaitu memiliki tingkat kepercayaan tinggi jika proses memberikan hasil yang sama untuk seorang guru yang dinilai oleh orang lain kapan pun.
3. Praktis yaitu jika dapat dilakukan oleh siapa saja dengan mudah dan konsisten dalam semua situasi tanpa memerlukan persyaratan tambahan.

2.3.6. Prinsip Penilaian Kinerja Guru

Agar penilaian kinerja guru dapat dipertanggung jawabkan, maka penilaian harus dilakukan sesuai dengan prinsip-prinsip berikut (Kemendikbud, 2016):

1. Objektif

Penilaian kinerja guru disesuaikan dengan situasi guru saat melakukan tugas sehari-hari.

2. Adil

Penilaian kinerja guru dilakukan secara adil dengan menggunakan standar, persyaratan, dan prosedur yang sama untuk semua guru.

3. Akuntabel

Hasil penilaian kinerja guru dapat dipertanggungjawabkan

4. Transparan

Proses penilaian kinerja guru yang transparan memungkinkan penilai, guru yang dinilai, dan pihak lain yang berkepentingan untuk mengetahui apa yang dinilai, bagaimana penilaian dilakukan, dan hasil penilaian

5. Terukur

Penilaian kinerja guru dilakukan secara kualitatif (melalui pengamatan dan pengawasan) dan kuantitatif (melalui indikator dan kriteria kinerja).

6. Komitmen

Penilaian kinerja guru dibuat karena penilai dan penilai memiliki kemampuan untuk menyesuaikan sikap dan tindakan untuk melakukan penilaian kinerja guru sesuai dengan prosedur.

7. Berkelanjutan

Guru harus melakukan penilaian kinerja setiap tahun selama mereka bekerja sebagai guru.

2.3.7. Tujuan Penilaian / Evaluasi Kinerja Guru

Penilaian kinerja guru bertujuan untuk memperoleh tentang kinerja guru di masa lalu dan memprediksi kinerja di masa depan. Secara teori, penilaian kinerja guru masuk dalam kategori evaluasi dan pengembangan (Jamalia, 2022). Tujuan utama sistem penilaian kinerja guru adalah (Hikam, 2018) :

1. Menentukan tingkat kemampuan seorang guru
2. Meningkatkan efektivitas dan efisien kinerja guru dan sekolah

3. Memberikan dasar pengambilan keputusan dalam menentukan efektif atau tidaknya kinerja seorang guru.
4. Memberikan landasan bagi kelanjutan program pengembangan profesi guru
5. Memastikan guru memenuhi peran dan tanggung jawabnya serta menjaga sikap positifnya dalam mendukung proses pembelajaran
6. Memberikan dasar bagi sistem yang akan meningkatkan peningkatan penembangan profesional dan karir guru serta bentuk kompensasi lainnya

2.4. Perancangan Sistem

Perancangan sistem merupakan gambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa komponen yang terpisah ke dalam suatu kesatuan yang utuh dan berfungsi. Tujuan perancangan sistem adalah sebagai berikut (Samsudin, 2018) :

1. Memenuhi kebutuhan sistem
2. Memberikan gambaran yang jelas dari rancangan bangun lengkap untuk pemrograman komputer dan ahli teknik lainnya

Untuk mencapai tujuan perancangan sistem, ada beberapa tujuan yang harus dipenuhi (Hayati, 2018):

1. Perancangan sistem harus bermanfaat dan mudah dipahami
2. Perancangan sistem dapat mendukung tujuan utama suatu perusahaan atau lembaga pendidikan
3. Perancangan sistem harus efisien dan efektif untuk mendukung pengambilan keputusan
4. Perancangan sistem harus dapat mempersiapkan rancangan bangunan yang terinci untuk masing-masing komponen sistem pendukung

2.5. Database

Database atau basis data adalah sekumpulan data yang disusun dan disimpan dengan baik di dalam komputer yang dapat diolah untuk menghasilkan informasi melalui penggunaan perangkat lunak atau *software* (Khairunnisa, 2019). *Database* atau basis data juga dapat didefinisikan sebagai wadah dari *file*, tabel, atau arsipan

yang disimpan secara bersama dalam media penyimpanan elektronik dan saling terhubung satu sama lain, yang menjadikannya "jiwa" dari sebuah aplikasi (Fakhriza dkk, 2022).

Tujuan sistem *database* termasuk memberikan sarana akses yang *fleksibel*, menjaga *integritas* data, melindunginya dari penggunaan yang tidak sah, dan memberikan sarana untuk penggunaan bersama (*share*) (Sutabri, 2014).

Untuk mengelola *database*, perangkat lunak yang disebut *Database Management System* (DBMS) diperlukan. DBMS adalah perangkat lunak sistem yang memungkinkan pemakai membuat, memelihara, mengontrol, dan mengakses basis data dengan cara yang mudah dan efisien (Suendri, 2019).

2.6. Web

Web adalah *bene* yang menggunakan protokol pengiriman *hiperteks* HTTP, yang berfungsi sebagai gudang informasi yang terdiri dari tulisan, gambar, video, audio, dan animasi. Data yang ditampilkan melalui internet (Nurjannah, 2021). Aplikasi berbasis *web* adalah aplikasi yang menjalankan aplikasi melalui teknologi *browser* dan dapat diakses melalui jaringan komputer. Aplikasi *web* sendiri adalah aplikasi yang disimpan di *server* dan dikirim melalui internet dan dapat diakses melalui antarmuka *browser* (Achmat, 2020).

Software yang disebut *web browser* berfungsi untuk menampilkan dan membuka halaman *web*, yang merupakan halaman internet. *Internet Explorer*, *Mozilla Firefox*, *Opera*, *Google Chrome*, dan lainnya adalah beberapa *web browser* yang dapat digunakan untuk tampilan *web design* (Samsudin, 2018).

2.7. Visual Studio Code

Perangkat lunak *Visual Studio Code* memiliki fitur lengkap yang digunakan untuk mengembangkan aplikasi bisnis dan pribadi dalam format *console*, *windows* dan *web* (Salsabila, 2022).

Microsoft mengembangkan editor kode sumber *Visual Studio Code* untuk *Windows*, *Linux*, dan *macOS*. Ini merupakan pendukung untuk *debugging*, kontrol *Git* tertanam, penyorotan sintaksis, penyelesaian kode cerdas, *snippet*, dan

refactoring kode. *Visual Studio Code* adalah aplikasi editor teks yang dibuat oleh *Microsoft*. Programer sangat menyukai text editor ini, meskipun masih baru. Ini mendukung hampir semua bahasa pemrograman, seperti vb.net, C#, C++, Java, XML, HTML, ASP, PHP dan lain – lain. *Visual code editor* dikembangkan oleh *Microsoft*, tetapi dapat bekerja dengan berbagai *platform* selain *Windows* (Khairunnisa, 2019).



Gambar 2.2. Logo Visual Studio Code

2.8. UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah bahasa yang dibangun berdasarkan grafik atau gambar dan digunakan untuk memvisualisasi, menspesifikasikan, membangun, dan mendokumentasikan sistem pengembangan *software* yang berbasis OO (*Object-Oriented*). UML juga memberikan standar penulisan sistem *blue print*, seperti konsep bisnis proses, penulisan kelas – kelas dalam bahasa program tertentu, skema *database* dan komponen – komponen yang diperlukan untuk sistem *software* (Suendri, 2019). Tujuan UML sendiri adalah sebagai berikut (Fakhriza dkk., 2022):

1. Memberikan bahasa pemodelan visual yang ekspresif dan siap pakai untuk mengembangkan dan menukar model secara mudah dan mudah dipahami.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa.
3. Menyatukan praktek – praktek terbaik yang ada dalam pemodelan.

2.8.1 *Use Case Diagram*

Use case diagram adalah model untuk menunjukkan bagaimana sistem akan berinteraksi dengan beberapa aktor. *Use case diagram* digunakan untuk mengetahui fungsi apa yang ada di dalam sistem dan siapa yang berhak menggunakannya. *Use case* memiliki beberapa simbol seperti pada Tabel 2.3 berikut ini (Fakhriza dkk, 2022) :

Tabel 2.3. Simbol Use Case Diagram

Simbol	Nama	Keterangan
	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, walaupun simbol aktor gambar orang tetapi belum tentu aktor adalah orang, biasanya dinyatakan dengan kata benda diawali frase nama aktor
	<i>Use Case</i>	<i>Fungsionalitas</i> sistem yang terdiri dari unit-unit yang dapat bertukar pesan satu sama lain atau antar aktor biasanya dinyatakan dengan kata kerja di awal frasa nama <i>use case</i> .
	<i>Assosiation</i>	Komunikasi antara aktor dan <i>use case</i> berinteraksi dengan aktor
	<i>Extend</i>	<i>Relasi use case tambahan</i> ke sebuah <i>use case</i> yang dapat berdiri sendiri.
	<i>Include</i>	<i>Relasi use case tambahan</i> dan <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat untuk menjalankan <i>use case</i> ini.
	<i>Generalization</i>	Hubungan <i>generalisasi</i> dan <i>spesialisasi</i> (umum – khusus) antara dua jenis <i>use case</i> dimana fungsi yang satu lebih umum dari pada fungsi lainnya

2.8.2. Activity Diagram

Aktivitas diagram menunjukkan aliran kerja (*workflow*) atau aktivitas dari sistem atau menu yang ada pada perangkat lunak dan bukan menunjukkan aktivitas yang dilakukan oleh aktor (Fakhriza dkk, 2022). *Aktivitas diagram* memiliki beberapa simbol seperti pada Tabel 2.4 berikut ini (Samsudin, 2018).

Tabel 2.4. Simbol Activity Diagram

Simbol	Nama	Keterangan
	<i>State</i>	Menambahkan state untuk suatu objek
	<i>Activity</i>	Menambahkan kegiatan baru pada diagram

	<i>Start Point</i>	Menunjukkan dimana aliran kerja dimulai
	<i>End Point</i>	Menunjukkan dimana aliran kerja berakhir
	<i>State Transition</i>	Menambah transisi dari suatu kegiatan ke kegiatan yang lainnya.
	<i>Fork</i> (Percabangan)	Menunjukkan kegiatan yang dilakukan secara parallel.
	<i>Join</i> (Penggabungan)	Menunjukkan kegiatan yang digabungkan.
	<i>Decision</i>	Menambahkan titik keputusan pada aliran kerja

2.8.3. Sequence Diagram

Sequence diagram atau diagram urutan adalah diagram yang menunjukkan interaksi antar objek - objek di dalam sistem yang disusun dalam urutan atau rangkaian waktu tertentu. Interaksi antar objek - objek tersebut seperti pengguna, *display* dan lainnya berupa pesan (*message*). *Sequence diagram* digunakan untuk menggambarkan suatu rangkaian langkah – langkah yang dilakukan sebagai tanggapan terhadap suatu kejadian (*event*) untuk menghasilkan *output* tertentu. *Sequence diagram* memiliki beberapa simbol seperti pada Tabel 2.5 berikut ini (Samsudin, 2018).

Tabel 2.5. Simbol Sequence Diagram

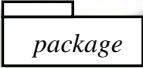
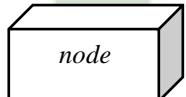
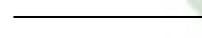
Simbol	Nama	Keterangan
	<i>Actors</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem.
	<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang dilakukan.
	<i>Boundary Class</i>	Menggambarkan sebuah form.
	<i>Control Class</i>	Menggambarkan penghubung antara <i>boundary</i> dengan tabel.

	<i>A Focus Of Control and A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message.
<u>A message()</u> 	<i>A Message</i>	Menggambarkan pengiriman pesan.

2.8.4. Deployment Diagram

Deployment diagram merupakan penunjuk pengkonfigurasian komponen pada saat aplikasi dijalankan. *Deployment diagram* berfungsi untuk memodelkan sistem penambahan (*embedded system*) yang mewakili rancangan (*device, node* dan *hardware*), bentuk *client / server*, metode terdistribusi murni dan rekayasa ulang aplikasi. *Deployment diagram* memiliki beberapa simbol seperti pada Tabel 2.6 dibawah ini (Rajagukguk, 2021) :

Tabel 2.6. Simbol Deployment Diagram

Simbol	Nama	Keterangan
	<i>Package</i>	Sampul untuk membungkus <i>node</i>
	<i>Node</i>	Gambar perangkat yang digunakan; itu tidak dibuat sendiri, dan jika ada komponen pendukung, itu harus sesuai dengan rancangan awal
	<i>Dependency / Ketergantungan</i>	Hubungan antar node yang menunjukkan langkah berikutnya
	<i>Link</i>	Sebagai penghubung di node yang digunakan

2.8.5 Class Diagram

Class Diagram atau diagram kelas menggambarkan struktur sistem yang digunakan untuk mendefinisikan kelas – kelas yang akan digunakan untuk membangun sistem. Kelas memiliki *atribut* yaitu variabel – variabel yang dimiliki oleh suatu kelas dan metode (operasi) yaitu fungsi – fungsi yang dimiliki oleh suatu kelas. *Class Diagram* memiliki beberapa symbol seperti pada Tabel 2.7 berikut ini (Fakhriza dkk, 2022).

Tabel 2.7. Simbol Class Diagram

Simbol	Nama	Keterangan
_____	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi pelaku dan struktur data dari objek yang ada diatasnya objek induk (<i>ancestor</i>)
◇	<i>Nary Association</i>	Usaha untuk menghindari asosiasi dengan lebih dari 2 objek
_____	<i>Class</i>	Himpunan objek – objek yang berbagi atribut dan operasi yang sama.
○	<i>Collaboration</i>	Urutan aksi - aksi yang ditampilkan oleh sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
◀-----	<i>Realization</i>	Objek yang benar – benar dilakukan oleh suatu objek
----->	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
_____	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.8.6 Component Diagram

Component diagram menunjukkan hubungan antara komponen yang terdapat pada suatu sistem yang akan dibangun. Komponen utama yang terdapat dalam suatu sistem adalah (Rajagukguk, 2021) :

1. *User interface* yaitu bagian yang mengatur tampilan sistem yang dibangun
2. *Business processing* yaitu komponen yang mengatur fungsi bisnis
3. Data yaitu komponen yang akan memanipulasi data
4. *Security* yaitu komponen yang bertanggung jawab atas keamanan sistem

Component diagram memiliki beberapa simbol seperti pada Tabel 2.8 dibawah ini.

Tabel 2.8. Simbol *Component Diagram*

Simbol	Nama	Keterangan
	<i>Package</i>	Sampul yang mengandung berbagai komponen
	<i>Komponen</i>	Komponen sistem
	<i>Dependency / Ketergantungan</i>	Menunjukkan bagaimana masing-masing komponen berfungsi dalam satu sistem
	<i>Interface / Antarmuka</i>	Mewakili komponen lain sehingga tidak dapat mengaksesnya secara langsung
	<i>Link</i>	Perhubungan antar komponen

2.8.7. *Object Diagram*

Object diagram merupakan gambaran dari suatu sistem yang akan dibangun, baik dalam memberikan nama objek dan proses yang dilalui objek tertentu. *Object diagram* akan memberikan penjelasan dari isi setiap atribut yang ada dalam tiap kelas. Hubungan link dalam *object diagram* merupakan sistem yang akan menghubungkan apabila ditemukan objek yang akan digunakan oleh objek lain. *Object diagram* memiliki beberapa simbol seperti pada Tabel 2.9 dibawah ini (Rajagukguk, 2021) :

Tabel 2.9. Simbol *Objek Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Objek yang berjalan dalam kelas ketika sistem sedang berjalan
	<i>Link</i>	Hubungan antara dua objek

2.8.8. *Statechart Diagram*

Statechart diagram menampilkan perubahan status atau transisi status suatu proses dan ditampilkan dalam bentuk graf berarah. *Statechart diagram*

menggambarkan interaksi yang ada pada objek, yang biasanya menunjukkan tahapan proses pengguna dengan sistem yang dijalankan. *Statechart diagram* memiliki beberapa simbol seperti pada Tabel 2.10 dibawah ini (Rajagukguk, 2021).

Tabel 2.10. Simbol *Statechart Diagram*

Simbol	Nama	Keterangan
●	<i>Initial State / Status Awal</i>	Kondisi awal yang memungkinkan sistem dimulai
○	<i>Final State / Status Akhir</i>	Kondisi akhir dari setiap tindakan
→	<i>Event</i>	Proses yang mengubah status sistem
state	<i>State</i>	Kondisi yang menggambarkan tindakan yang terjadi

2.8.9. *Collaboration Diagram*

Collaboration diagram adalah perluasan dari objek dan diagram. Objek diagram menunjukkan objek – objek dan hubungannya satu dengan yang lain. *Collaboration diagram* menunjukkan *message – message* objek yang dikirim satu sama lain (Samsudin, 2018).

2.8.10. *Package Diagram*

Package diagram adalah suatu bentuk pengelompokan untuk mengambil sebuah bentuk di UML dan mengelompokkan elemen – elemen pada tingkatan unit yang lebih tinggi. Kegunaan *package diagram* adalah untuk mengelompokkan *class* (Samsudin, 2018)

2.9. Bahasa Pemrograman Web

2.9.1. HTML

Hypertext Markup Language (HTM) adalah sekumpulan simbol – simbol atau tag – tag dalam sebuah file yang digunakan untuk menampilkan halaman pada

web browser (Muhiban, 2020). Beberapa manfaat dari HTML antara lain adalah (Rahmad, 2022) :

1. Dapat membuat halaman *web* lebih mudah dibaca dan dipahami oleh pengguna
2. Tag HTML digunakan untuk menandai teks pada halaman dengan membuatnya seperti teks tebal miring, bergaris bawah dan pilihan format yang lebih tersedia
3. Mengelola perilaku *web* dengan *JavaScript*, menerapkan server PHP dan merancang halaman *web* dengan CSS
4. Dapat memasang gambar, video, tabel dan lain – lain.
5. Membuat formulir online dan tandai elemen

2.9.2. XAMPP

XAMPP merupakan sebuah *software web server apache* yang didalamnya sudah tersedia *database server MySQL* dan dapat mendukung pemrograman PHP. XAMPP merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di Linux dan Windows. Keuntungan lainnya adalah hanya dengan menginstal satu kali sudah tersedia *Apache Web Server, MySQL, Database server, PHP Support* (Ningsih dkk, 2022)



Gambar 2.3. Logo XAMPP

2.9.3. PHP

PHP adalah singkatan dari *Hypertext Preprocessor* yaitu sebuah bahasa pemrograman yang perintahnya dilaksanakan oleh server dan hasilnya ditampilkan pada komputer pengguna. PHP juga menggambarkan HTML embedded yaitu sintaks PHP yang dituliskan bersamaan dengan sintaks HTML. Jadi PHP dan HTML adalah penggabungan bahasa pemrograman yang saling bekerja sama. PHP juga merupakan bahasa pemrograman bersifat *open source* yang dapat digunakan secara gratis. Sifat *open source* pada PHP memberikan kemampuan berkembang cepat (Amin, 2022).

Bahasa pemrograman PHP selain dapat membuat dokumen HTML, secara dinamis, dapat membuat gamar, PDF dan animasi *flash* dengan *script* yang sederhana. Bahasa pemrograman PHP dapat bekerja sama dengan baik pada sebagian besar DBMS diantaranya oracle, MSSQL, SQL server, MySQL, database dan PostgreSQL (Amin, 2022).



Gambar 2.4. Logo PHP

2.9.4. MySQL

MySQL merupakan salah satu jenis *database server* untuk membangun aplikasi *web* dengan menggunakan *database* sebagai sumber dan pengelola datanya. MySQL dan PHP merupakan pasangan perangkat lunak yang ideal dalam pengembangan aplikasi *web* (Muhiban, 2020). MySQL umumnya digunakan untuk mengolah data dimana pengolahan data yang umum digunakan adalah *CREATE*, *UPDATE*, *DELETE*, *UPDATE* dan *SELECT* (Nujannah, 2022).

MySQL adalah server *database* yang merupakan pilihan untuk pengembangan aplikasi karena (Rahmad, 2022) :

1. Fleksibel

MySQL dapat dimanfaatkan untuk pengembangan aplikasi area kerja dan aplikasi *web* dengan memanfaatkan inovasi yang berbeda. Dengan menyediakan *plug-in* dan *driver* yang unik untuk masing – masing teknologi tersebut. MySQL mampu beradaptasi dengan teknologi yang akan digunakan sebagai pengembang aplikasi yaitu PHP, JSP, Java, Delphi, C++ atau lainnya

2. Performa Tinggi

MySQL memiliki performa yang tinggi karena memiliki mesin kueri yang cepat sehingga pemrosesan transaksi dapat dilaksanakan dengan cepat

3. Lintas Platform

MySQL dapat digunakan di berbagai lingkungan atau *platform* (dalam hal ini sistem operasi) termasuk Microsoft Windows, Linux dan Unix. Karena itu *migrasi* dan antar sistem operasi jika perlu menjadi lebih sederhana

4. Gratis

MySQL tersedia gratis, namun ada juga perangkat lunak MySQL yang komersial karena layanan telah ditambahkan dengan keterampilan tertentu dan menerima layanan dukungan teknis

5. Proteksi Data Handal

MySQL di industri *database* memprioritaskan perlindungan keamanan data sebagai prioritas utama. Fitur manajemen pengguna MySQL, *enkripsi* data dan fitur lainnya menjadikannya alat yang efektif untuk menangani hal ini

6. Komunitas Luas

MySQL memiliki komunitas yang besar karena banyaknya aplikasi. Ini berguna jika terjadi masalah pemrosesan data MySQL



Gambar 2.5. Logo MySQL

2.9.5. CSS

CSS merupakan singkatan dari *Cascading Style Sheets* merupakan salah satu bahasa pemrograman *web* untuk pengaturan *layout*, kerangka, gambar, warna dan lain sebagainya (Muhiban, 2020). Dalam kebanyakan kasus, halaman *web* yang ditulis dalam HTML atau XHTML diformat dengan CSS. Fungsi utama CSS adalah untuk membedaan antara konten dokumen dan konten situs *web* dalam kaitannya dengan tampilan dokumen di halaman situs *web* (Rahmad, 2022)



Gambar 2.6. Logo CSS

2.10. Penelitian Yang Relevan

Penelitian yang relevan adalah penelitian – penelitian yang sebelumnya pernah dilakukan yang memiliki kaitan dengan judul penelitian yang diteliti. Tujuannya adalah untuk memperkuat penelitian yang akan dilakukan karena sudah

ada referensi ilmiah yang memiliki relevansi. Adapun penelitian yang relevan yang menjadi acuan penulis sebagai referensi dalam melakukan penelitian ini dapat dilihat pada Tabel 2.11 berikut ini.

Tabel 2.11. Penelitian Yang Relevan

No	Nama	Judul	Tahun	Hasil Penelitian
1	Dadang Adib Fangsuri dan Sri Lestanti	Penerapan Metode <i>Additive Ratio Assesment</i> (ARAS) Untuk Mendukung Penilaian Kinerja Guru Pada SDN Sentul 2	2020	Hasil dari penelitian ini menunjukkan bahwa perancangan dalam membangun aplikasi penilaian kinerja guru berbasis <i>web</i> dengan menggunakan metode ARAS dapat mengolah nilai dan data guru secara sistematis dan mampu memberikan rekomendasi guru terbaik
2.	Arif Kusdiantoro	Rancang Bangun Aplikasi Penilaian Kinerja Guru Berbasis <i>Website</i> Menggunakan Metode <i>Graphic Rating Scale</i> (Studi Kasus : SMA	2020	Hasil dari penelitian ini menunjukkan bahwa aplikasi yang digunakan dapat memberikan perhitungan yang sesuai dengan metode <i>Graphic rating scale</i> serta dapat melakukan penilaian sesuai dengan aturan yang telah ditetapkan oleh Peraturan Menteri Pendidikan Nasional Nomor 16 Tahun 2007 Tentang standar kualifikasi akademik dan

		Barunawati Surabaya)		kompetensi guru. Selain itu aplikasi juga dapat memberikan histori dari penilaian kinerja
3.	Manik Ayu Titisari	Pemilihan Mitra Konsultan Menggunakan <i>Analytical Network Process</i>	2021	Hasil dari penelitian ini menunjukkan bahwa hasil dari pembobotan dapat digunakan sebagai input dalam menyusun strategi rantai pasok khususnya pemilihan mitra kerja yang akan digunakan dalam jangka waktu yang panjang
4.	Khoirun Nisa, Febrian Murti Dewanto dan Rahmat Robi Waliyansyah	Sistem Pendukung Keputusan Penilaian Kinerja Guru (PKG) Menggunakan Metode <i>Topsis</i> (Studi Kasus : SDN 01 Gabus)	2021	Hasil dari penelitian ini menunjukkan bahwa penilaian kinerja guru sudah berjalan dengan baik dan dapat menghasilkan pembobotan kriteria penilaian dan informasi yang jelas dan cepat serta dapat menampilkan hasil perangkingan yang lebih akurat
5.	Firdaus Tafolano	Implementasi Metode	2022	Hasil dari penelitian menunjukkan bahwa dengan

		<p><i>Analytical Network Process (ANP)</i></p> <p>Pada Sistem Pendukung Keputusan Dalam Seleksi Atlet Penyandang Disabilitas Sumatera Utara Untuk Bertanding di Ajang Pekan Paralimpiade (Peparnas)</p>		<p>sistem yang telah dirancang dapat mempermudah pihak Dinas Pemuda dan Olahraga Kota Medan dalam melakukan proses seleksi atlet penyandang disabilitas yang baik dan berkomponen untuk di daftarkan pada kompetisi Pekan Paralimpiade Nasional (Perpanas)</p>
--	--	---	--	--

Berdasarkan uraian penelitian yang relevan, maka dapat diambil kesimpulan bahwa penelitian yang relevan ini memiliki kesamaan dengan penelitian yang penulis lakukan yaitu sama – sama bertujuan membuat suatu sistem pendukung keputusan yang lebih mudah dan akurat. Sedangkan perbedaannya dapat dilihat pada :

1. Metode sistem pendukung keputusan yang digunakan dimana penulis menggunakan metode *Algoritma Analytical Network Process (ANP)* sedangkan pada penelitian yang relevan menggunakan metode *Additive Ratio Assesment (ARAS)*, *Graphic Rating Scale* dan *Topsis*
2. Penerapan metode *Analytical Network Process (ANP)* dimana penulis menggunakan metode tersebut untuk “*Penilaian Kinerja Guru*” sedangkan pada penelitian relevan digunakan untuk “*Pemilihan Mitra Konsultasi*” dan “*Seleksi Atlet Penyandang Disabilitas Sumatera Utara Untuk Bertanding di Ajang Pekan Paralimpiade (Peparnas)*”.



UNIVERSITAS ISLAM NEGERI
SUMATERA UTARA MEDAN