BAB II TINJAUAN PUSTAKA

2.1 Pengertian Sistem Pendukung Keputusan

Sistem pendukung keputusan dimaksudkan menjadi alat bantu bagi para pengambil keputusan untuk memperluas kapabilitas mereka, namun tidak untuk menggantikan penilaian mereka [3].

Sistem Pendukung Keputusan (SPK) merupakan sistem berbasis komputer, yang dapat mendukung pengambilan keputusan untuk menyelesaikan masalah yang semi terstruktur, dengan memanfaatkan data yang ada kemudian diolah menjadi suatu informasi berupa usulan menuju suatu kepuusan tertentu [4]

2.1.1 Komponen Komponen Sistem Pendukung Keputusan

Terdapat, setidaknya tiga komponen utama yang tersusun dalam sebuah sistem spendukung keputusan, antara lain sebagai berikut.

1. Database Management

Manajemen basis data merupakan sub sistem dalam data yang terorganisir pada sebuah database. Untuk kepentingan SPK sendiri, diperlukan data yang relevan dengan permasalahan yang hendak diselesaikan melalui sistem berbasis simulasi.

2. User Interface

Tampilan antarmuka atau pengelolaan dialog adalah proses penggabungan antara dua komponen, yaitu database management dan model base yang nantinya akan bergabung dengan user interface. Nantinya User Interface (UI) akan menampilkan output atau keluaran sistem bagi pengguna perangkat lunak.

3. Model Base

Komponen model merepresentasikan terkait permasalahan ke dalam format data kuantitatif. Yang di dalamnya terdiri dari tujuan permasalahan, komponen, batasan (*constraint*), dan hal terkait lainnya. Mode base sangat memungkinkan untuk menganalisa permasalahan secara utuh dan mengembangkannya menjadi solusi yang terbaik.

2.1.2 Karakteristik Sistem Pendukung Keputusan

Sistem pendukung keputusan memiliki beberapa karakteristik antara lain:

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

2.1.3 Fase Pengambilan Keputusan

Menurut [5] Model pengambilan keputusan menggunakan fase yang terdiri dari 4 fase sebagai bentuk mekanisme dalam proses pengambilan keputusan berdasarkan alternatif dan kriteria. Proses penbambilan keputusan terdiri dari 4 fase yaitu:

1. Fase Penelusuran (*Intelligence*)

Tahap ini merupakan proses penelusuran dan pendeteksian dari lingkup problematika serta proses pengenalan masalah. Data masukan diperoleh, diproses, dan diuji dalam rangka mengidentifikasikan masalah.

2. Fase Perancangan (*Design*)

Pada tahap ini dilakukan dengan melakukan perancangan seperti: perancangan fitur, menu aplikasi, perancangan data, perancangan arsitektur, perancangan interface dan perancangan prosedur.

3. Fase Pemilihan (*Choice*)

Pada tahap ini dilakukan proses pemilihan diantara berbagai alternatif tindakan yang mungkin dijalankan. Hasil pemilihan tersebut kemudian diimplementasikan dalam proses pengambilan keputusan.

4. Fase Implementasi (*Implementation*)

Tahapan ini merupakan tahapan optional dalam pengembangan perangkat lunak. Bagian ini terjadi ketika sistem yang di maksud telah selesai dan mengalami perubahan ataupun permintaan penambahan fitur dikemudian hari.

2.2 Kelas Unggulan

Kelas Unggulan adalah kelas yang diikuti oleh sejumlah siswa yang unggul dalam tiga ranah penilaian dengan kecerdasan di atas rata-rata yang dikelompokkan secara khusus yaitu penilaian seleksi kelas unggulan ini berpedoman pada nilai akademik, prestasi,, absensi, nilai intelektual dan nilai kepribadian siswa [1].

2.3 Metode TOPSIS

TOPSIS (*Technique For Others Reference by Similarity to Ideal Solution*) adalah salah satu metode pengambilan keputusan multikriteria dengan menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan terjauh dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak *Euclidean* untuk menentukan kedekatan relatif dari suatu alternatif dengan solusi optimal [2].

Solusi ideal positif didefinisikan sebagai jumlah dari seluruh nilai terbaik yang dapat dicapai untuk setiap atribut, sedangkan solusi negatif-ideal terdiri dari seluruh nilai terburuk yang dicapai untuk setiap atribut. TOPSIS mempertimbangkan keduanya, jarak terhadap solusi ideal positif dan jarak terhadap solusi ideal negatif dengan mengambil kedekatan relatif terhadap solusi ideal positif. Berdasarkan perbandingan terhadap jarak relatifnya, susunan prioritas alternatif bisa dicapai. 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, melibatkan beberapa orang pengambil keputusan.

2.3.1 Tahapan Metode TOPSIS

Tahapan topsis digunakan untuk melakukan langkah penerapan keputusan seperti berikut :

1. Menentukan kriteria dan sifat

Kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu Ci dan sifat dari masing-masing kriteria.

Menentukan rating kecocokan
 Rating kecocokan setiap alternatif pada setiap kriteria.

Membuat matriks keputusan yang ternormalisasi
 TOPSIS membutuhkan rating kinerja setiap alternatif Ai pada setiap kriteria
 Ci yang ternormalisasi, yaitu:

$$Rij = \frac{Xij}{\sqrt{\sum_{i=1}^{m} Xij^2}}$$

4. Perkalian antara bobot dengan nilai setiap atribut
Perkalian ini untuk membentuk matrik Y. dapat ditentukan berdasarkan ranking bobot ternormalisasi (yij) sebagai berikut:

$$y_{ij} \!\! = w_i \, r_{ij}$$

dengan i=1,2,...,m dan j=1,2,...,n

 Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif dan negatif

Jarak antara alternatif Ai dengan solusi ideal positif dirumuskan sebagai:

$$Dj^{+} = \sqrt{\sum_{j=1}^{n} (yi^{+} - yij)^{2}}$$

Jarak antara alternatif Ai dengan solusi ideal negatif dirumuskan sebagai:

$$Dj^{-} = \sqrt{\sum_{j=1}^{n} (yij - yi^{-})^2}$$

6. Menentukan nilai preferensi untuk setiap alternatif
Nilai preferensi untuk setiap alternatif (Vi) diberikan sebagai:

$$Vi = \frac{Di^-}{Di^- + Di^+}$$

Nilai Vi yang lebih besar menunjukkan bahwa alternatif Ai lebih dipilih.

2.4 Web Based

Web Based adalah aplikasi yang dibuat berbasis web yang membutuhkan web server dan browser untuk menjalankannya [6].

Sistem berbasis *web based* ada beberapa hal yang penting dan harus kita pikirkan sebelum membangun sistem tersebut [7], diantaranya:

- 1. Tidak membutuhkan *hardware* dengan spesifikasi yang tangguh untuk menjalankan aplikasinya.
- 2. Server yang dibutuhkan cukup diinstallkan *tools* pendukung saja agar klien mudah menjalankan aplikasi
- 3. Infrastruktur jaringan yang dibutuhkan juga cukup besar karena aplikasi yang dibuat dapat diakses dari jaringan luar (internet).
- 4. Aplikasi berbasis *web based* dapat diakses dari berbagai perangkat dengan syarat menggunakan *web browser* saja sudah dapat mengaksesnya.
- 5. Jika aplikasi yang sudah jadi ingin di *update*, sangat mudah untuk melakukannya karena tidak membutuhkan membuka keseluruhan aplikasi.

2.4.1 PHP

PHP adalah bahasa server-side-scripting yang menyatudengan HTML untuk membuat halaman web yang dinamis. PHP Hypertext Preprocessor adalah bahasa pemograman web server-side yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server (server side HTML embedded scripting). PHP adalah script yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru/up to date. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan. Dengan menggunakan program PHP, sebuah website akan lebih interaktif dan dinamis [8].

PHP merupakan bahasa pemprograman yang digunakan oleh pengembang untuk membuat sistem *website* dengan kumpulan bahasa HTML dan *script* lainya [9].

2.4.2 **MySql**

MySQL adalah singkatan dari Structue Query Language yang digunakan untuk mendefinisikan structure data, memodifikasi data pada basis data, menspesifikasi batasan keamanan (security), hingga pemeliharaan data [10].

Mysql adalah RDBMS yang cepat dan mudah digunakan, serta sudah banyak digunakan untuk berbagai kebutuhan

MySQL merupakan bahasa standar yang paling banyak digunakan untuk mengakses database relasional dan merupakan aplikasi yang dapat dipergunakan secara bebas.

2.4.3 XAMP

XAMPP adalah software yang di dalamnya tertdapat server MySQL dan didukung oleh PHP sebagai bahasa pemrograman untuk membuat website dinamis serta terdapat web server apache yang dapat dijalankan di beberapa platform seperti OS X, Windows, Linux, Mac, dan Solaris. XAMPP merupakan software server apache dimana dalam XAMPP yang telah tersedia database server seperti MySQL dan PHP programming. XAMPP memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada Windows dan linux. Keuntungan lain yang didapatkan adalah hanya dengan melakukan instalasi cukup satu kali kemudian didalamnya tersedia MySQL, apacheweb server, Database server PHP support (PHP 4 dan PHP 5) dan beberapa modul lainnya.

Dari pengertian diatas disimpulkan bahwa XAMPP merupakan software server apache di mana memiliki banyak keuntungan seperti mudah untuk digunakan, tidak memerlukan biaya serta mendukung pada instalasi Windows dan Linux. Hal ini juga didukung karena dengan instalasi yang di lakukan satu kali tersedia MySQL, apache web server, Database server PHP support.

2.4.4 Kamus Data

Kamus data adalah sistem yang terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi tersedia saat dibutuhkan. Kamus data berfungsi sebagai penyedia informasi bagi user dalam pengintergrasian data data agar saling terhubungantara satu dengan lainnya [11]. Beberapa istilah dalam basis data antara lain:

1. Field.

Merupakan bagian terkecil dari *record* yang memiliki nilai untuk dijadikan kunci sebagai pewakil *record* lainya.

2. *Cadidat Key*

Merupakan atribut yang mengidentifikasikan secara spesifik *field* kunci dari suatu *entity*.

3. Primary Key

Merupakan atribut yang mengidentifikasikan dan mewakili transaksi secara spesifik.

4. Alternate Key

Merupakan kunci kadidat dari primary key yang belum terpakai.

5. Record

Merupakan kumpulan dari elemen atau field yang saling terkait.

6. Entity

Merupakan tempat kejadian informasi direkam.

7. Table

Merupakan menu untuk memasukan data data pada database.

8. Foreign Key

Pelengkap hubungan antara suatu atribut dengan atribut induk.

Operasi yang dapat dilakukan oleh basis data adalah :

- 1. Pembuatan basis data baru (*create database*)
- 2. Pembuatan *file/table* (*create table*)
- 3. Penghapusan *file/table* (*drop table*)
- 4. Penambahan dan pengisian *table* (*insert*)
- 5. Pengubahan data (*update*)
- 6. Penghapusan data (*delete*)

Berikut merupakan tabel keterangan kamus data :

Tabel 2.1 Keterangan Kamus Data

No.	Nama Field	Туре	Size	Keterangan

2.5 UML (Unified Modelling Language)

UML (unified Modelling Language) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teksteks pendukung. Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada UML (Unified Modelling Language) [11].

2.5.1 Use Case Diagram

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut [11]. Berikut simbol-simbol yang akan digunakan dalam menggambarkan Use Case Diagram dapat dilihat pada Tabel 2.2.

Tabel 2.2 Simbol *Use Case Diagram*

Simbol	Deskripsi
	Use case: Fungsionalitas yang disediakan sistem
	sebagai unit-unit yang saling bertukar pesan antar
	unit atau aktor, biasanya dinyatakan dengan
	menggunakan kata kerja di awal frase nama use
	case.
	Aktor: seseorang/sesuatu yang berinteraksi dengan
+	yang akan dibuat. diluar sistem informasi.
\downarrow	Biasanya dinyatakan menggunakan kata benda
	Asosiasi (association): merupakan komunikasi
	antara aktor dan use case yang berpartisipasi pada
	use case atau use case memiliki interaksi dengan
	aktor.
	Simbol

Simbol No **Deskripsi** Generalisasi (generalization): merupakan 4. hubungan (umum – khusus) antara dua buah *use* case dimana fungsi yang satu adalah fungsi yang lebih umum << Include >> Include berarti use case yang ditambahkan akan ·····> 5. dipanggil saat use case tambahan dijalankan. <<Extend>> Ekstensi (extend) merupakan use case tambahan ke sebuah *use case* yang ditambahkan dapat 6.

Tabel 2.2 Simbol *Use Case Diagram* (Lanjutan)

2.5.2 Activity Diagram

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem [11]. Berikut simbol-simbol yang akan digunakan dalam menggambarkan activity diagram dapat dilihat pada Tabel 2.3.

 Tabel 2.3 Simbol Activity Diagram

berdiri sendiri walau tanpa use case tambahan itu.

No.	Simbol	Keterangan		
		Status awal aktivitas sistem, sebuah diagram		
1.		aktivitas memiliki sebuah status awal.		
		Aktivitas yang dilakukan sistem, aktivitas biasanya		
2.		diawali dengan kata kerja.		
		Percabangan (Decision) merupakan asosiasi		
3.		percabangan dimana jika ada pilihan aktivitas lebih		
3.		dari satu.		
		Penggabungan (Join) merupakan asosiasi		
4		penggabungan dimana lebih dari satu aktivitas		
4.		digabungkan menjadi satu.		
1	i	1		

Tabel 2.3 Simbol *Activity Diagram* (Lanjutan)

No.	Simbol	Keterangan
5.	Nama swimlane	Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

2.5.3 Class Diagram

Class diagram mengembangkan struktur sistem dari segi pendefinisian kelaskelas yang akan dibuat untuk membangun sistem [11]. Berikut simbol-simbol yang akan digunakan dalam menggambarkan Class Diagram dapat dilihat pada Tabel 2.4.

Tabel 2.4 Simbol Class Diagram

No.	Simbol	Deskripsi
1.	Nama_kelas +Atribute +Operasi	Kelas pada struktur sistem.
2.	Antar Muka/Interface Nama_Interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	Asosiasi / Asociation	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan symbol
4.	Asosiasi Berarah / Digunakan Association	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>symbol</i> .
5.	Generalisasi	Relasi antar kelas dengan makna generalisasi- spesialisasi (umum khusus)

No. **Simbol** Deskripsi Ketergantungan / Relasi kelas makna antar dengan 6. dependency ketergantungan antar kelas. Agregasi / aggregation Relasi antar kelas dengan makna semua 7. bagian (whole-part)

Tabel 2.4 Simbol *Class Diagram* (Lanjutan)

2.6 Pengujian Black Box Testing

Black box testing yaitu pengujian perangkat lunak dari segi pendefinisian fungsional tanpa menguji desain dan kode program. Pengjian dimaksudkan untuk mengetahui apakan fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan [11].

Pengujian yang dilakukan dengan membuat kasus yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji dilakukan harus dibuat dengan benar dan salah, seperti proses *login* "Jika user memasukan *username* dan *password* yang benar maka dapat *login*?"

2.7 Penelitian Terdahulu

Penelitian terdahulu digunakan sebagai acuan refrensi pada penelitian yang dilakukan:

Nama Judul **Hasil Penelitian** Peneliti Sahadi, Maulana Pendukung Perencanaan sistem pemilihan siswa/i sistem Ardhiansyah, Keputusan untuk masuk kelas unggulan Husain Pemilihan SMPN 13 Tangerang Τ. Siswa/i Selatan dapat menjadi solusi untuk (2020)Kelas sekolah. Selain dapat mempersingkat Unggulan waktu perhitungan, Menggunakan sistem yang dibangun juga dapat Metode mengurangi tingkat resiko kesalahan **AHP** dan **TOPSIS** perhitungan yang pada umumnya terjadi akibat human error

Tabel 2.5 Penelitian Terdahulu

Tabel 2.5 Penelitian Terdahulu (Lanjutan)

Nama Peneliti	Judul	Hasil Penelitian
Sushanty Saleh (2016)	Model Sistem Penunjang Keputusan Penentuan Jurusan Bagi Siswa SMA Menggunakan Metode TOPSIS (Studi Kasus Yadika Natar)	Sistem ini diharapkan dapat membantu dalam menentukan dan memilih siswa/i yang direkomendasikan untuk masuk ke jurusan IPA serta mempermudah proses penilaian dalam melakukan perangkingan.
Sushanty Saleh, Indera, Asrul Zakaria (2021)	Sistem Pendukung Keputusan dalam Memprediksi Kelulusan Mahasiswa IIB Darmajaya dengan Metode TOPSIS	Hasil aplikasi prediksi kelulusan yang dibangun dapat membantu para dosen pembimbing akademik dalam mengawasi serta memberi bimbingan dan nasehat kepada mahasiswa aktif semester 5 dan 6 agar bisa lulus tepat pada waktunya.
Danang Ramawanto (2018)	Sistem Pendukung Keputusan Penentuan Kelas Unggulan Di SMP NEGERI 1 BAKUNG Menggunakan Metode TOPSIS	Penerapan metode Technique Order Preference by Similarity To Ideal Solution(TOPSIS) dapat mempercepat proses penyeleasaian perhitungan pemilihan calon penerima beasiswa di SMP Negeri 1 Bakung dengan menghitung nilai atau skor untuk setiap alternatif, dan menentukan rangkingan alternatif yang dibangun dengan bahasa pemrograman PHP
Bobby Bachry, Dona Yuliawati (2021)	Sistem Penunjang Keputusan Penilaian Kelayakan Huni Perumahan dengan Metode Simple Additive Weighting (Studi Kasus Rija Husaein)	Sistem yang dibangun mampu membantu dan mempercepat kerja tim penilai dalam melakukan penilaian akhir kelayakan rumah dan meminimalisir kesalahan dalam menentukan rumah yang akan
Sutedi, Hendri Purnomo, Nirmala Handayani (2019)	The Application of Profile Matching Method in Decision Support System for Selection of Training Instructors (Case Study at IIB Darmajaya's Training Center)	Metode dengan sistem pendukung keputusan untuk membangun pilihan yang layak untuk membuat keputusan yang tepat berdasarkan seperangkat kriteria yang telah ditentukan.