

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Berikut ini akan dijabarkan beberapa penelitian terdahulu yang berkaitan dengan judul yang di angkat sebagai berikut :

Penelitian yang dilakukan oleh Alit Suryo Irawan (2016) dengan judul Sistem Pendukung Keputusan Pemilihan Jurusan Di Sma Islam Sudirman Ambarawa Menggunakan Metode Analytical Hierarchy Process (AHP). Hasil penelitian berupa prototipe produk aplikasi yang telah diuji di lapangan serta dinyatakan memenuhi tujuan yang diharapkan dan hasil penjurusan siswa bisa lebih cepat dan akurat sehingga informasi yang dihasilkan bisa digunakan sebagai pendukung keputusan.

Penelitian yang dilakukan oleh Arman (2019) dengan judul Penerapan Sistem Pendukung Keputusan Penentuan Arah Minat Siswa Dengan Metode Ahp Pada Sma N 14 Padang. Pada metode AHP semua nilai diberikan bobot masing-masing kriteria seperti nilai rapor, nilai UN dan nilai tes minat jurusan yang cocok untuk siswa tersebut dan pekerjaan guru BK juga dapat terbantu. Hasil penelitian dari data siswa nama Givania Pembega dengan nilai rapor 0,1554, nilai rata-rata UN 0,0449 dan nilai minat 0,0729 dapat nilai total 0,2732 dengan minat jurusan digolongkan pada Jurusan IPA Dengan adanya aplikasi sistem pendukung keputusan penentuan arah minat siswa ini dapat dijadikan salah satu alat bantu dalam pengambilan keputusan untuk siswa dalam menentukan arah minat mereka,

dan dapat juga membantu guru BK dalam penentuan arah minat siswanya. Dalam melakukan penelitian ini metode penelitian menggunakan metode SDLC dan alat bantu untuk perancangan sistem menggunakan UML. Bahasa pemrograman menggunakan PHP dan MySQL.

Penelitian yang dilakukan oleh Juarni Siregar (2017) dengan judul Sistem Pendukung Keputusan Penentuan Prioritas Konseling Siswa, Menggunakan Pendekatan AHP-MOORA. Dari hasil percobaan dapat disimpulkan bahwa metode ini mampu memberikan hasil yang baik dan mendekati hasil dari penilaian yang digunakan oleh guru BK. Hasil penelitian ini lebih mendekati prioritas yang harus diberikan oleh siswa karena setiap kriteria penilaiannya memiliki bobot tersendiri dibandingkan dengan prioritas yang tanpa bobot dan hanya dihasilkan dari nilai rata-rata saja.

Penelitian yang dilakukan oleh Samuel Manurung (2018) dengan judul Sistem Pendukung Keputusan Pemilihan Guru Dan Pegawai Terbaik Menggunakan Metode MOORA. Di penulisan ini adalah salah satu merupakan studi kasus yang dapat diselesaikan dengan menggunakan sistem pendukung keputusan, dimana yang menjadi persoalan yang dihadapi di dalam SMP negeri 1 Palipi adalah bagaimana memilih guru dan pegawai yang terbaik di dalam sekolah dan untuk melakukan sebuah seleksi harus menggunakan dengan cara manual dan proses penilaian menjadi lama untuk mendapatkan hasil. Oleh karena itu dibuat sebuah Sistem yang mendukung keputusan yang dapat membantu proses penilaian dan dimana Sistem pendukung keputusan yang dilakukan ini menggunakan metode moora dan dimana metode moora digunakan untuk untuk menguji coba di

dalam correctness yang bertujuan untuk mengetahui akurasi nilai yang diperoleh oleh system, uji coba sensitivitas diberikan pada nilai bobot kriteria dan uji coba modifikasi yang bertujuan untuk mengetahui seberapa banyak kriteria yang dapat ditambahkan.

Penelitian yang dilakukan oleh Mesran (2018) dengan judul Sistem Pendukung Keputusan Pemilihan Peserta Jaminan Kesehatan Masyarakat (Jamkesmas) Menerapkan Metode MOORA. Sistem pendukung keputusan sebagai sebuah sistem berbasis komputer yang terdiri atas komponen-komponen antara lain komponen sistem bahasa (language), komponen sistem pengetahuan (knowledge) dan komponen sistem pemrosesan masalah (problem processing) yang saling berinteraksi satu dengan yang lainnya, yang membantu pengambilan keputusan melalui penggunaan data dan model-model keputusan untuk memecahkan masalah yang sifatnya semi terstruktur maupun yang tidak terstruktur. Penelitian ini menggunakan Metode MOORA dalam menentukan yang berhak menjadi para peserta Jamkesmas berdasarkan kriteria dengan menggunakan rumus yang hasilnya lebih akurat dan tepat sasaran.

II.2. Landasan Teori

II.2.1. Sistem

Sebuah sistem terdiri dari berbagai unsur yang saling melengkapi dalam mencapai tujuan dan sasaran. Unsur-unsur yang terdapat dalam sistem itulah yang disebut dengan subsistem. Subsistem-subsistem tersebut harus saling

berhubungan dan berinteraksi melalui komunikasi yang relevan sehingga sistem dapat bekerja secara efektif dan efisien (Eka Iswandy ; 2015 : 72).

II.2.2 Informasi

Informasi merupakan data yang telah diorganisir sehingga memberikan arti dan nilai kepada penerimanya. Sedangkan menurut Jogiyanto, Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Dapat dikatakan bahwa data merupakan bahan mentah, sedangkan informasi adalah bahan jadi atau bahan yang telah siap digunakan, Jadi, sumber dari informasi adalah data (Fanny Andalia ; 2015 : 93).

Informasi merupakan data yang telah proses sedemikian rupa sehingga meningkatkan pengetahuan orang yang menggunakan data tersebut. Berdasarkan beberapa pengertian informasi dari para ahli, maka dapat disimpulkan bahwa informasi adalah data yang telah diproses menjadi bentuk yang bernilai bagi penerimanya dan bermanfaat dalam setiap pengambilan keputusan (Rudi Hermawan ; 2016 : 32).

II.2.3. Sistem Pendukung Keputusan

Sistem pendukung keputusan (SPK) atau *Decision Support System* (DSS) adalah sistem informasi interaktif yang menyediakan informasi, pemodelan, dan manipulasi data yang digunakan untuk membantu pengambilan keputusan pada situasi yang semi terstruktur dan situasi yang tidak terstruktur di mana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat. Konsep

DSS dikemukakan pertama kali oleh Scott-Morton pada tahun 1971. Beliau mendefinisikan cikal bakal DSS tersebut sebagai “Sistem berbasis komputer yang interaktif, yang membantu pengambil keputusan menggunakan data dan model untuk memecahkan persoalan-persoalan tidak terstruktur” (Abdul Kadir ; 2014 : 108).

Sistem pendukung keputusan (SPK) biasanya dibangun untuk mendukung solusi atas suatu masalah atau untuk suatu peluang. Aplikasi sistem pendukung keputusan (SPK) digunakan dalam pengambilan keputusan. Aplikasi sistem pendukung keputusan (SPK) menggunakan CBIS (*Computer Based Information System*) yang fleksibel, interaktif, dan dapat diadaptasi, yang dikembangkan untuk mendukung solusi atas masalah manajemen spesifik yang tidak terstruktur. Sistem pendukung keputusan sebagai sistem berbasis komputer yang terdiri dari tiga komponen yang saling berinteraksi, sistem bahasa (mekanisme untuk memberikan komunikasi antara pengguna dan komponen sistem pendukung keputusan lain), sistem pengetahuan (respositori pengetahuan domain masalah yang ada pada sistem pendukung keputusan atau sebagai data atau sebagai prosedur) dan sistem pemrosesan atau lebih kapabilitas manipulasi masalah umum yang diperlukan untuk pengambilan keputusan (Nofriansyah ; 2014 : 1)

Sistem pendukung keputusan terdiri dari 3 komponen utama atau subsistem yaitu :

1. Subsistem data (*database*) Subsistem data merupakan komponen penyedia data bagi sistem. Data yang dimaksud disimpan dalam *database* yang diorganisasikan oleh *Data Base Management System* (DBMS). Pangkalan

data berasal dari dua sumber yaitu sumber internal (dari dalam perusahaan) dan sumber eksternal (dari luar perusahaan).

- a. Data Internal Merupakan data yang disimpan dalam satu atau beberapa tempat dalam suatu organisasi. Data ini mengenai orang, produk, servis dan proses. Contoh : data mengenai data pribadi klien.
 - b. Data eksternal Data eksternal berupa data dari luar sistem yang masih berkaitan dengan sistem tersebut dan berpengaruh pada sistem. Contoh: Data standar dari WHO.
 - c. Ekstraksi Data Ekstraksi data merupakan teknik menghubungkan (*interfacing*) data internal dengan data eksternal dan akan menghasilkan basis data yang akan digunakan bersama model base SPK dan Subsistem manajemen dialog yang menghasilkan SPK.
2. Subsistem model (model base) Model yang disimpan ditambahkan rincian keterangan dan penjelasan yang kompeherensif mengenai model yang dibuat, sehingga mampu : mengakses dan mengintegrasikan sub rutin model, menghubungkan model dengan model yang lain melalui pangkalan data dan mengelola model base dengan fungsi manajemen yang analog dengan manajemen *database*.
 3. Subsistem Dialog (*user system interface*) Sistem diartikulasikan dan diimplementasikan sehingga pemakai dapat berkomunikasi dengan SPK. Fasilitas yang dimiliki oleh subsistem ini adalah : Bahasa aksi (*action language*), Bahasa tampilan dan Bahasa pengetahuan (Sariyun Naja Anwar ; 2012 : 2)

II.2.4. Metode AHP

Metode AHP dikembangkan oleh Thomas L. Saaty, seorang ahli matematika. Metode ini adalah sebuah kerangka untuk mengambil keputusan dengan efektif atas persoalan yang kompleks dengan menyederhanakan dan mempercepat proses pengambilan keputusan dengan memecahkan persoalan tersebut kedalam bagian-bagiannya, menata bagian atau variabel ini dalam suatu susunan hirarki, member nilai numerik pada pertimbangan subjektif tentang pentingnya tiap variabel dan mensintesis berbagai pertimbangan ini untuk menetapkan variabel yang mana yang memiliki prioritas paling tinggi dan bertindak untuk mempengaruhi hasil pada situasi tersebut. Metode AHP ini membantu memecahkan persoalan yang kompleks dengan menstruktur suatu hirarki kriteria, pihak yang berkepentingan, hasil dan dengan menarik berbagai pertimbangan guna mengembangkan bobot atau prioritas. Metode ini juga menggabungkan kekuatan dari perasaan dan logika yang bersangkutan pada berbagai persoalan, lalu mensintesis berbagai pertimbangan yang beragam menjadi hasil yang cocok dengan perkiraan kita secara intuitif sebagaimana yang dipresentasikan pada pertimbangan yang telah dibuat (Alit Suryo Irawan ; 2018:

4)

II.2.5. Metode MOORA

Metode Moora menggunakan perkalian sebagai untuk menghubungkan rating atribut, dimana rating atribut harus dipangkatkan dulu dengan bobot yang bersangkutan, Preferensi untuk alternatif Si. Secara umum, prosedur MOORA meliputi langkah-langkah sebagai berikut: 1. Penentuan nilai matrik Menentukan Tujuan untuk mengidentifikasi atribut evaluasi yang bersangkutan 2. Normalisasi matriks Mewakikan semua informasi yang tersedia untuk setiap atribut dalam bentuk matriks keputusan. $X =$ 3. Normalisasi matriks Breures (2008) menyimpulkan bahwa untuk penyebut, pilihan terbaik adalah akar kuadrat dari jumlah kuadrat dan setiap alternatif peratribut.

$$X^*_{ij} = X_{ij} / \sum = \dots\dots\dots (1)$$

Untuk $j = 1 \dots m$ 4. Mengoptimalkan Atribut Untuk optimasi Multiobjektif, ukuran yang dinormalisasi ditambahkan dalam kasus maksimasi (untuk atribut yang menguntungkan) dan dikurangi dalam kasus minimasi (untuk atribut yang tidak menguntungkan).

$$Y_i = \sum - \sum = + = \dots\dots\dots (2)$$

Dimana G adalah jumlah atribut yang akan dimaksimalkan, (n-g) adalah jumlah atribut yang akan diminimalkan, dan y_i adalah nilai penilaian yang telah dinormalisasikan dari alternatif 1 terhadap semua atribut.

Sebuah urutan peringkat dan Y_i menunjukan pilihan terakhir. Alternatif terbaik memiliki nilai Y_i tertinggi, sedangkan alternatif terburuk memiliki nilai yang rendah (Mesran ; 2018 : 17).

II.2.6. Visual Basic

Microsoft Visual Basic 2010 adalah salah satu komponen Microsoft Visual Studio 2010. Software ini diluncurkan Microsoft pada tanggal 12 April 2010 dengan nama kode Dev10 dan menggunakan .Net Framework 4.0. *Integrated Development Environment* (IDE) pada Visual studio 2010 telah didesain ulang sehingga lebih enak dipandang dan digunakan programmer.

Untuk kode editor-nya, Visual Basic 2010 telah menambah fitur *highlights reference*. Ketika satu simbol/kode dalam bahasa pemrogramannya dipilih, maka simbol/kode yang sama, meskipun penggunaannya berbeda akan terlihat berwarna sama. Misal jika kode math dipilih, seluruh kode math akan terlihat berwarna sama (Fadillah ; 2014 : 10).

II.2.7. Database

Database atau basis data adalah sekumpulan data yang memiliki hubungan secara logika dan diatur dengan susunan tertentu serta disimpan dalam media penyimpanan komputer. Data itu sendiri adalah representasi dari semua fakta yang ada pada dunia nyata. Database sering digunakan untuk melakukan proses terhadap data-data tersebut untuk menghasilkan informasi tertentu. Misalnya dari data nama siswa dan tanggal lahir siswa anda bisa mendapatkan informasi nama siswa yang berulang tahun pada hari ini. Tentu saja informasi tersebut akan anda dapatkan dari software pemrosesan database dengan cara anda memberikan perintah dalam bahasa tertentu yang SQL (*Structured Query Language*) (Wahana Komputer ; 2013 : 56)

II.2.8. SQL Server

SQL Server 2008 adalah sebuah terobosan baru dari Microsoft dalam bidang database. SQL Server adalah sebuah DBMS (*Database Management System*) yang dibuat oleh Microsoft untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti IBM dan Oracle. SQL Server 2008 dibuat pada saat kemajuan dalam bidang hardware sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa SQL Server 2008 membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data (Komputer ; 2013 : 2)

II.2.9. Unified Modelling Language (UML)

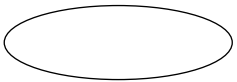
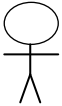

Menurut Gata (2013 : 4) Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak


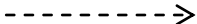

UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

1. *Use case* Diagram

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. 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. Simbol-simbol yang digunakan dalam *use case* diagram, yaitu :

Tabel II.4. Simbol Use Case

Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i>.</p>
	<p>Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i>, tetapi tidak memiliki control terhadap <i>use case</i>.</p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan</p>


	siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan 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.



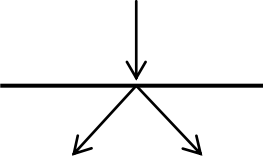
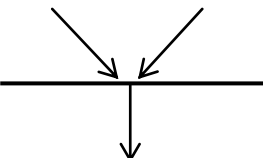
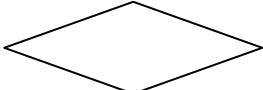

(Sumber : Gata, 2013 : 4)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu :

Tabel II.5. Simbol *Activity Diagram*

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan

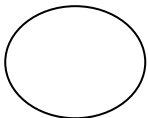
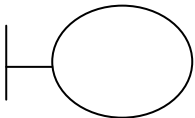
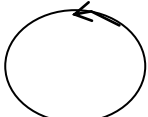

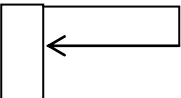


	merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	<i>Join</i> (penggabungan) 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</i> diagram untuk menunjukkan siapa melakukan apa.

(Sumber : Gata, 2013 : 6)

3. 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*, yaitu :

Tabel II.6. 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 formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <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 : Gata, 2013 : 7)

4. *Class Diagram* (Diagram Kelas)

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

Tabel II.7. *Multiplicity Class Diagram*

Multiplicity	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 : Gata, 2013 : 9)